взлом и защита WebMoney

крис касперски ака мыщъх no-email

вопреки всем заверениям разработчиков система WebMoney катастрофически ненадежна и вскрывается буквально ногтем. существует множество червей, троянов и хакерских групп, специализирующихся на похищении электронных кошельков, кражи которых приняли массовый характер. хотите узнать как это делается и как обезопасить себя?

введение

Начнем с того, чего не может быть. Никаких "генераторов WebMoney" не существует и не может существовать в принципе. Вся наличность храниться на центральном сервере оператора, а электронные кошельки представляют лишь средство доступа к ней. Грубо говоря, от того, что вы сгенерируете комбинацию цифр для кодового замка, деньги и драгоценности в сейфе еще не появятся. И хотя существует возможность подобрать шифр к чужому сейфу, вероятность открыть его без помощи владельца (гусары! про паяльник мы помним, но молчим) настолько мала, что об этом даже не стоит и говорить!

А вот украсть чужую комбинацию вполне реально! Именно этим "генераторы WebMoney" и занимаются. Они либо делают дубликат с электронного кошелька и передают их злоумышленнику, либо скрыто вызывают Keeper'a и осуществляют на свой счет. Аналогичным образом действуют вирусы и троянские программы. Так же отмечены и целенаправленные атаки на конкретную жертву. Можно ли от них защититься? Система WebMoney, разработанная неспециалистами, изначально проектировалась без оглядки на безопасность и хотя в последнее время появился целый комплекс "противопожарных" мер, приляпанных задним числом, положение остается критическим. Пользователи путаются в системах защиты, служба поддержки дает довольно туманные и расплывчатые рекомендации (обновить Windows, настроить брандмауэр и т. д.), а тем временем кражи электронных кошельков продолжаются.

Мы не ставим перед собой задачу научить кого бы то ни было воровать, мы просто хотим показать и доказать (!), что система WebMoney действительно очень ненадежна и проектировалась даже не задницей (к ней все-таки примыкает спинной мозг), а вообще неизвестно чем. Здесь не будет расплывчатых слов (чтобы нас не обвинили в клевете), но не будет и конкретных рекомендаций. Мы не дает готовых атакующих программ и не говорим какие именно байтики нужно хакнуть, но поверьте, весь необходимый хакерский инструментарий может быть создан с нуля за одну ночь — святое для хакеров время!

Но обо всем по порядку. Не будем спешить вперед и совать лазерный диск в дисковерт, тем более что последний нам еще понадобиться.

Рисунок 1 они появляются из мрака, снимают все электронные деньги и уходят в никуда

>>> врезка ЧТО МОЖНО И ЧТО НЕЛЬЗЯ (ОТРЕЧЕНИЕ)

Экспериментировать (в образовательных целях) можно только со своим собственным электронным кошельком или с кошельками лиц, давших письменное разрешение. Несанкционированное вмешательство в чужие системы и кошельки ***категорически недопустимо***!

Рисунок 2 они ловят тех, кто появляется из мрака, и уводят в тот мир, откуда уже нет возврата

начало начал или классическая миссионерская

Система WebMoney является своеобразным аналогом обыкновенных банковских чеков, а это значит, что для совершения платежей нам в обязательном порядке необходимо предварительно зарегистрироваться на центральном сервере оператора и открыть счет, что уже является огромным недостатком, ну да ладно.

Идем на www.webmoney.ru, скачиваем программу Keeper Classic, запускам ее (кстати, через Proxy-сервер заставить это чудо научно-инженерной мысли работать мне так и не удалось, пришлось поднимать NAT и маппить 2802 порт), заполняем регистрационные данные (от фонаря или честно), придумываем себе любой пароль по вкусу, после чего программа приступает к генерации секретного ключа и просит нас подергать мышь и понажимать клавиши. Это необходимо для того, чтобы получить действительно случайные данные, как будто пседослучаный генератор на основе таймера здесь не годится. На фоне общей незащищенности системы бравировать словами RSA, RC5, MD4, MD5, SSL просто глупо. Впрочем, психологический расчет разработчиков мне вполне понятен. Если секретный ключ будет генерироваться за доли секунды — какой пользователь в него поверит?

Рисунок 3 она ни от куда не появляется, никого не ловит, а просто сидит и паяет

Как бы там ни было по завершению регистрации нам присваивается уникальный 12-значиный идентификатор WMID (Web Money ID), и генерируется пара ключей. Открытый ключ передается на центральный сервер оператора WebMoney, а секретный сохраняется в файле с расширением *.kwm (Key of Web Money), который может быть расположен на жестком диске, сменном носителе или смарт-карте. Короче, обыкновенная несимметричная криптография типа PGP.

Еще создается файл *.pwm, хранящий сведения о наших кошельках (текущий баланс, история операций и т. д.). В принципе он необязателен, ведь вся информация расположена на центральном сервере оператора. Keeper может работать и без *.pwm файла, автоматически подгружая данные из сети, правда только за последние три дня. Собственно говоря, *.kwm файл тоже необязателен и его можно восстановить. Для этого необходимо знать пароль, иметь доступ к почтовому ящику, указанному при регистрации, а так же нотариально заверенное заявление, что ты не лось (подробнее об этом можно прочитать здесь: http://www.owebmoney.ru/returnkey.shtml). Чисто теоретически хакер может хакнуть наши денежки ***только*** на основе пароля, но практически это слишком хлопотно и небезопасно.

Секретной информацией, регламентирующий доступ к кошельку, является один лишь kwm-ключ. WMID везде публикуется открыто и это нормально. Зная WMID, можно узнать регистрационные данные пользователя, которые он пометил "открытыми", но нельзя определить номер его кошелька (кошельков).

Номер кошелька — это условно-секретная информация. Зная номер кошелька, мы не можем вытащить с него деньги, но можем выставить счет, заполнив поле "описание покупки" как можно более правдоподобно. Способ конечно, дурацкий, но есть некоторый шанс, что он пройдет. Пользователи, регулярно оплачивающие большое количество мелких счетов, постепенно привыкают не обращать на них внимания и проверяют графу "от кого" только при возникновении сомнений. Разумеется, никакого кайфа в таком способе взлома нет, к тому же злоумышленник может очень нехило погореть и отправится в компанию дядей, которые разорвут ему задницу, так что заметной популярности от так и не сыскал.

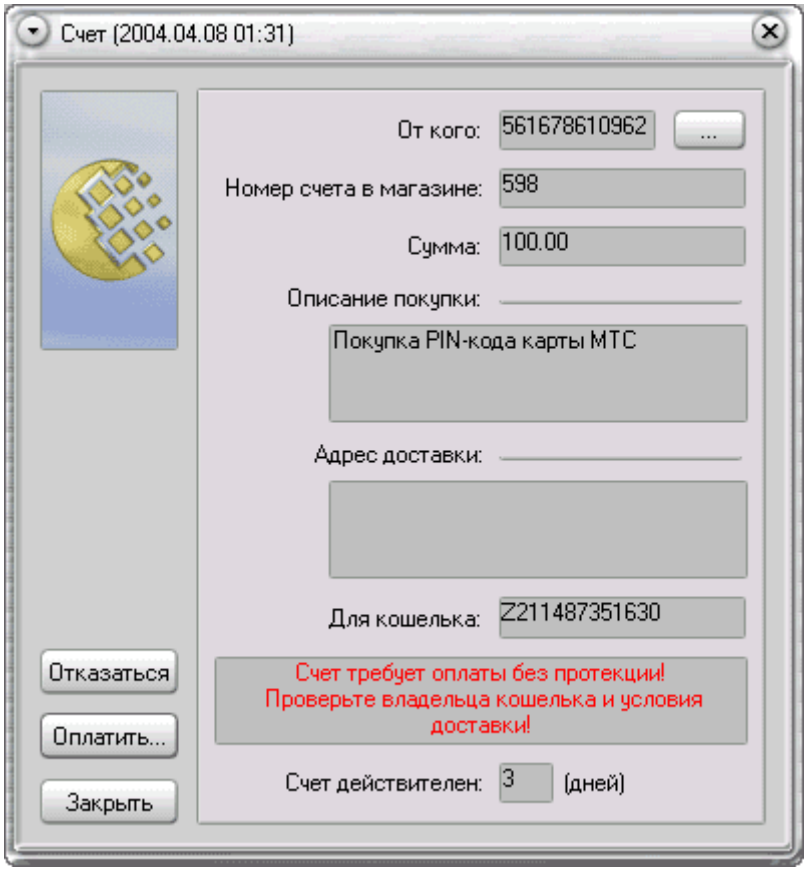

Рисунок 4 выставляем жертве левый, но правдоподобный счет — вдруг да оплатит?

А вот кража kwm-файлов процветает. По умолчанию ключи сохраняются в keys.kwm, но в-приципе имя файла может быть любым, как впрочем и расширение. Большинство хакеров и троянских программ производит тупой поиск по маске *.kwm, поэтому переименование файла ключей в dontreadme.txt до некоторой степени увеличивает нашу защищенность, однако, продвинутые хакеры могут залезть в реестр, где Keeper хранит свои настройки и подсмотреть путь к файлу. Еще можно искать по его содержимому, сканируя все файлы (правда, это займет много времени и вызовет подозрительную дисковую активность). Гурманы наверняка перехватят вызов API-функции CreateFile, показывающий какие файлы открывает Keeper. И даже если формат настроек реестра в последующих версиях будет изменен, вариант с CreateFile продолжит работать (hint: если бы разработчики не были идиотами, они бы создали несколько файлов с ключами, один подлинный, все остальные — сторожевые датчики, при обращении к которым раздается сигнал тревоги).

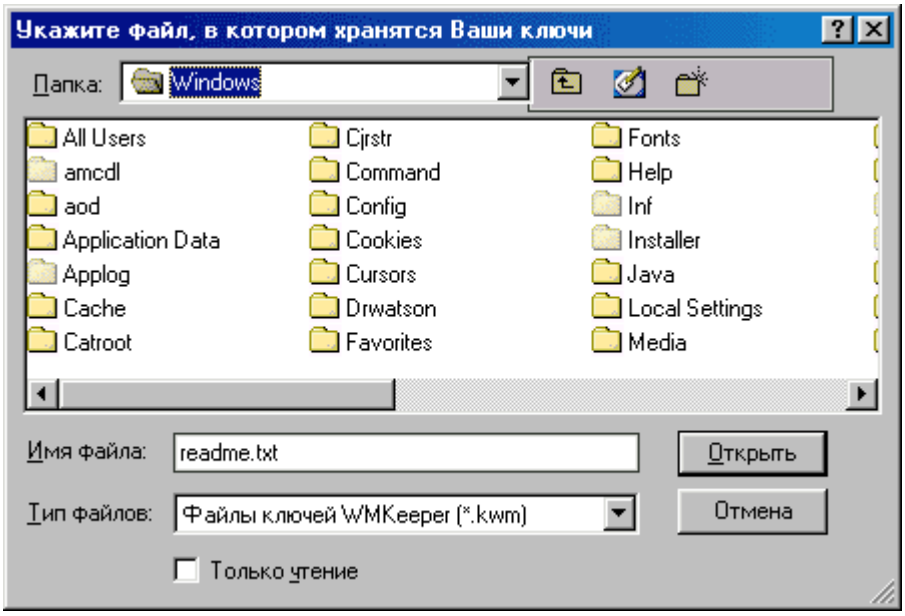

Рисунок 5 прячем kwm-файл подальше от хакеров

По умолчанию размер файла ключей составляет 1.2 Мбайт (в аккурат на дискету), но при желании его можно увеличить вплоть до 100 Мбайт. Это затрудняет кражу ключа с передачей по Интернет, и в общем-то не создает никаких непреодолимых неудобств. 100 Мбайт это половина mini CD-R, один Zip-100M или два CD-R в формате бизнес-карты. Конечно, быстродействие системы до некоторой степени упадет (огромный файл так сразу и не прочтешь), однако, безопасность стоит того. Или не стоит? По локальной сети утащить 100 Мбайт не проблема, по DSL модему или кабельному интернету тоже. И даже позорный по нынешним меркам модем на 33600 педераст этот файл за ~70 часов. Не так уж и много, если вспоминать, что практически никто из пользователей не перегенерирует ключи каждый день. Разрезав файл на мелкие кусочки, передаваемые в фоновом режиме, утащить его за две-три недели вполне реально, хотя это будет самый тупой и неперспективный путь.

Если хакер внедрился в чужую систему (а внедриться в нее можно разными путями), ему ничего не стоит загрузить файл в память, открыть кошелек, перевести деньги на свой счет и грохнуть жесткий диск, чтобы жертва не смогла войти в Интернет и пожаловаться кому следует. Кстати, на счет "пожаловаться". Вариантов не так уж и много и помощи ждать не от куда. Ну разве что от господа бога (if you're real god, return my money, you sic fuck) да на братков. Если доступ к WMID у нас еще есть (что за тупой хакер попался!), можно определить WMID на который были переведены деньги, зайти на сайт Арбитражного Сервиса (http://arbitrage.webmoney.ru/), оплатить арбитражный сбор (а для этого необходимо иметь WebMoney, которые у нас подчистую умыкнул злоумышленник), и заблокировать хакерский кошелек. Только если хакер не лось, деньги за считанные минуты будут переброшены на e-gold или любым другим путем выведены из системы, так что на его кошельке их не окажется и блокировать будет особо и нечего. Кстати говоря, кошельки с начальным или персональным аттестатом блокируется только по решению арбитражной комиссии, то есть достаточно взять аттестат и… Вот только не надо говорить, что владельцы аттестатов воровством не занимаются, поскольку сообщают свои паспортные данные. Агащазблин! Так и свои. Выдачей аттестатов сейчас занимаются все кому не лень и надеяться что все они люди честные, добросовестные и неподкупные просто наивно, тем более когда речь идет о деньгах, пусть даже электронных. Человек, который вознамерился похитить $100.000 (а почему бы и нет), получит без проблем не только фиговый аттестат, но еще и фальшивый паспорт в придачу. Ну и кого по этому аттестату потом искать?! Если даже сотрудники МВД подделывают паспорта на потоке, о чем не раз говорило TV (а это уже криминал), то что говорить за "аттестаты", у которых вообще нет никакого юридического статуса?!

Впрочем, ситуация с переброской ворованных денег через несколько кошельков все-таки рассматривалась разработчиками, и они тщательно поработились о… злоумышленниках! Судите сами. Жертве после подачи уже упомянутого иска следует обратиться к Администратору Арбитражного Сервиса (WMID 937717494180, arbitrage@webmoney.ru), и попросить его проследить всю цепочку. Вся "прелесть" в том, что Администратор работает

только с понедельника по пятницу с 10 до 18 часов по Москве. Мы, мол, не служба спасения и тоже спать хотим. Очень хорошая платежная система скажу я вам!!! При том что вывод денег из системы осуществляется практически мгновенно и счет идет на минуты, администратор видите ли хочет баиньки. Я не понял, это студенческая общага или платежная система?! Что стоило при миллионных оборотах (о которых реклама не перестает упоминать) нанять несколько человек для круглосуточной поддержки?! Ведь речь в данном случае идет о деньгах! Естественно, для хакеров безопаснее всего совершать кражи либо в полночь, либо на выходных. Но это ладно, оставим пустые слова и познакомимся с Keeper'ом поближе.

keeper снаружи и изнутри

Вот тут некоторые восхищаются как разработчиком удалось так много втиснуть в объем Keeper'а ("*не знаю как вы, а я искренне преклоняюсь перед теми, кто в 2 мегабайта дистрибутива Keeper Classic умудрился вложить такую "вкусную" начинку, да еще и красиво упаковать это дело снаружи" http://www.owebmoney.ru/clashistory.shtml*). А что они, собственно говоря, в него вместили? Конечно, в наш век, когда Hello, World с трудом вмешается на лазерный диск, программы занимающие "всего" несколько мегабайт уже вызывают уважение…

Основной объем (~2,2 Мбайта) занимает WMClient.dll который, собственно, сам Keeper и есть. Это DCOM-объект, написанный на Microsoft Visual .NET с компиляцией в машинный код, ничем не упакованный и никак, я повторяю, ***никак*** не препятствующий своему анализу. Здесь нет ни шифрованного, ни р-кода, ни антиотладочных приемов, ни противодействия дизассемблеру, дамперу, API-шипону. Ничего! Бери-и-анализируй! Во всяком случае версия 2.4.0.3 (самая последняя на момент написания этой статьи) ведет себя именно так. Будь разработчики хоть малость поумнее они либо использовали Microsoft Visual C++ 6 (знаменитую "шестерку") плюс любой качественный протектор (например, ExeCryptor), лобо откомпилировали NET-приложение в р-код, который намного сложнее дизассемблировать.

WebMoney.exe (~180 Кбайт) это только "пускалка" и в ней нет ничего интересного, тем не менее дизассемблировать его все-таки стоит. Хотя бы затем, чтобы посмеяться над разработчиками и оценить их квалификацию.

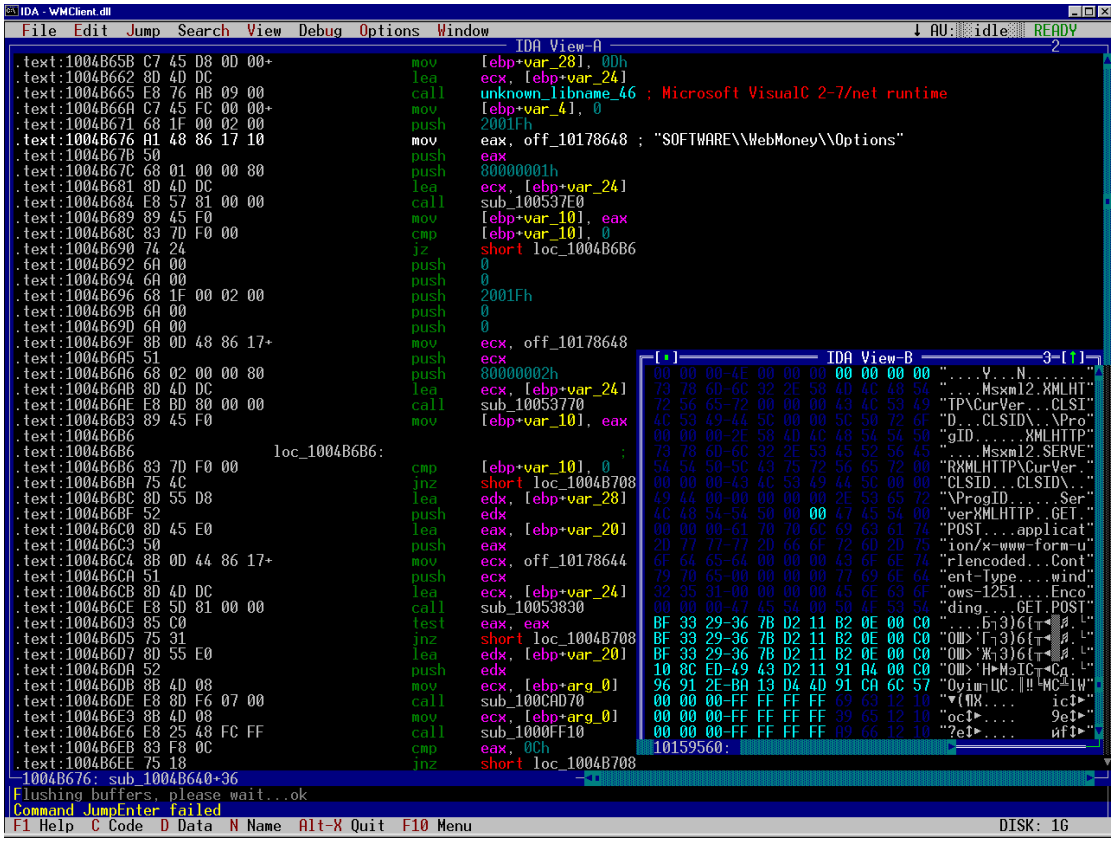

Рисунок 6 Keeper Classic в дизассемблере

Итак, будем считать, что на компьютер с установленным Keeper'ом внедрен хакерский код, исполняющийся с пользовательскими привилегиями (условимся, что администраторских прав нам не дали, и хотя повысить свои привилегии с пользователя до system в W2K/XP в общем-то не проблема, не говоря уже о 9x, где никакого разделения привилегий отродясь не бывало, будем действовать в спартанских условиях приближенным к боевым). Что мы можем сделать? У нас два пути. Предварительно дизассемблировать Keeper'a, восстановить протокол обмена с сервером, дождаться когда будет вставлен носитель на котором лежит секретный ключ и… дальше фантазируйте сами. Лично мне, ковыряться в Keeper'e — лень. Дизассемблирование это кропотливое дело и на восстановление протокола обмена может уйти не одна неделя. Использование снифферов существенно сокращает этот срок, однако, все равно ломы. Гораздо проще и эффективнее воровать деньги руками самого Keeper'a. Устанавливаем шпиона, перехватывающего клавиатурный ввод, дожидаемся ввода WMID или определяем его другими путями, ведь WMID ни для кого секретом не является (первый способ в основном используется вирусами, второй — хорош при целенаправленной атаке), затем в одним "прекрасный" момент (после 18 часов или в выходной день) отключаем вывод на экран, запускам WebMoney.exe и путем эмуляции клавиатурно-мышиного ввода делам все, что мы хотели. Например, пополняем кошелек жертвы. А почему бы и нет?! Мы же ведь ломаем свой собственный кошелек, верно? Вот его и пополним! Мы же не бандиты какие, а честные хакеры!

Техника эмуляции ввода подробно описана в "Записках мыщъх'а", электронную версию которой можно бесплатно сжевать с моего мыщъх'иного ftp сервера nezumi.org.ru (только напоминаю, что он доступен не все время), к тому же в 67 номере Хакера была опубликована статья "Ломка WebMoney" в которой все это описано. Так что не будем разводить демагогию и жевать резину по сто раз. Отметим лишь общий механизм. Сначала мы находим окно Keeper'a вызовом функции FindWindow или EnumWindows и определяем его дескриптор. Затем, используя EnumWindows перечисляем дочерние окна, принадлежащие элементам управления (кнопкам, строкам редактирования и т. д.). Посылая элементам управления разнообразные сообщения (это можно сделать с помощью функции SendMessage) мы легко возьмем их под своей контроль. Отключение вывода на экран осуществляется либо перехватом служб GDI (реализуется сложно, но действует на ура), либо расположением поверх Keeper'a отвлекающего окна, например, окна браузера с порнографической картинкой. Да много всякого тут можно придумать!

Проблема в том, что начиная с некоторого времени тупая эмуляция перестала действовать. Keeper обзавелся так называемыми "летающими цифрами". Вроде тех, что используется для предотвращения автоматической регистрации на многих сайтах. Прежде чем совершить какой-то платеж, необходимо ввести три графических цифры, которые случайным образом появляются на экране. Идея, конечно, интересная, да вот позаимствована она явно невпопад. Тяжелое детство, хреновое образование, глубокое похмелье. А голова-то бо-бо. Впрочем, голова тут не причем. Все равно ей думать некому. Приемам безопасности разработчиков явно не учили. Отрывочные знания в стиле "тут зубил, а тут девушку танцевал, а тут меня двинули кирпичом" так и прут изо всех сторон.

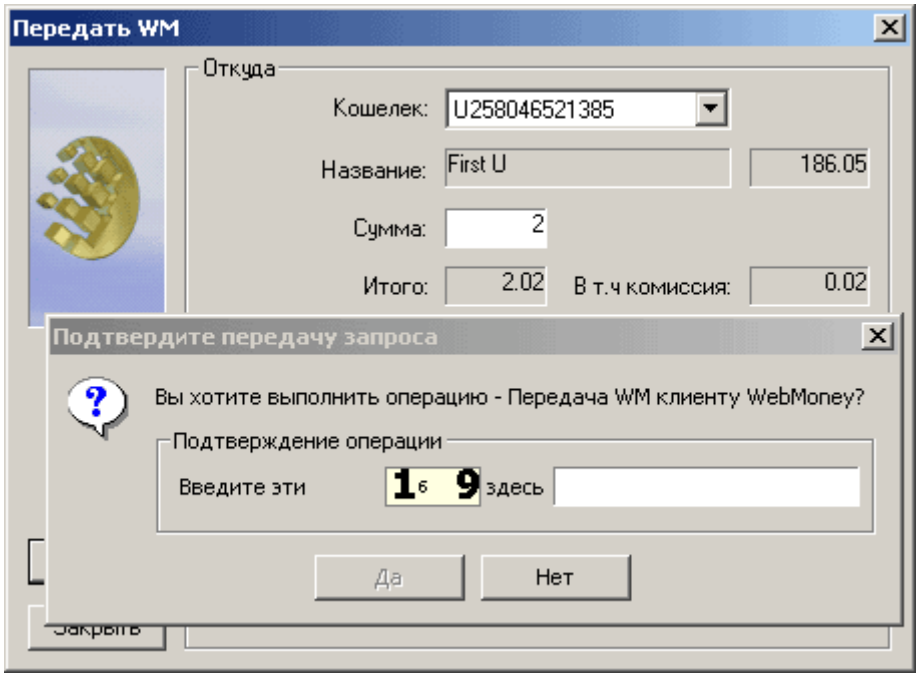

Рисунок 7 защита Keeper'а "летающими цифрами"

Почему "летающие цифры" действуют на web-серверах (там, где они впервые и появились)? Да потому и *только потому*, что во-первых, защитный код находится вне пределов досягаемости хакера, а, во-вторых, потому, что защита нацелена исключительно на роботов, но не людей. Для охраны mail.ru от спамеров и вандалов такой меры более чем достаточно, но только не для Keeper'а! Во-первых, в текущих версиях Keeper'а летающие цифры элементарно распознаются простеющим OCR, свободно умещающимся в сотню килобайт (при использовании готовых библиотек), во-вторых, хакерскому коду ничего не стоит захватить кусочек экрана и отправить его дежурящему у монитора хакеру, чтобы тот распознал их самостоятельно, в-третьих, эта защита отключается бит-хаком, т. е. правкой машинного кода Keeper'а, в-четвертых летающие цифры можно вырубить через реестр (если попытаться их отключить средствами самого Keeper'а, он запросит подтверждение на легитимность этой операции), в-пятых, даже если защита будет ужесточена, в запасе у хакеров останется расшифровка протокола обмена и создание своих собственных клиентов без всяких там цифр, в шестых… Короче, способов взлома очень и очень много и никакой пользы от этой защиты нет, не говоря уже о том, что многие пользователи до сих пор сидят на старых версиях без летающих цифр или отключают их за ненадобностью.

А вот еще одна широко разрекламированная фишка — подтверждение авторизации по e-mail. На неискушенный взгляд все выглядит железно — прежде, чем с нашим счетом удастся что бы то ни было сделать, необходимо ввести код, который придет по e-mail. Если хакер упрет *.kwm файл, он останется с носом, а мы — с деньгами. Ведь доступа к нашему почтовому ящику он не получит. Логика железная, но неправильная. Почтовые ящики ломаются не так уж и сложно (конкретные приемы взлома приведены во множестве книг и статей, так что не будет повторяться), к тому же, коль скоро хакер утащил *.kwm файл, он утащит и пароль на e-mail. Исключение составляет, пожалуй, лишь кража смарт-карт и сменных носителей с ключами, но… такая кража как правило осуществляется либо близкими людьми, которые могут поиметь и e-mail, либо грабителями, получившими физический доступ к сменному носителю, хранимому, как правило, в непосредственной близости от компьютера. Ну и что им стоит украсть еще и пароль на ящик?

Ладно, а как на счет блокировки всех IP адресов, кроме своего? Начнем с того, что в локальных сетях захват чужого адреса не является непреодолимой проблемой. Тот же, кто сидит на Dial-Up'е как правило получает динамические IP адреса, выделяемые из общего пула. Прописывать их — задолбешься, да и любой клиент того же провайдера будет авторизован без проблем. Но это неважно. Никакому хакеру хранить у себя чужой кошелек на хрен не нужно. Он просто снимет деньги руками Keeper'а, запущенного на компьютере жертвы, который наверняка имеет правильный IP и никакая "блокировка" его не останавливает!

Защитные меры, предлагаемые разработчиками, можно перечислять очень долго. Практически все они ориентированы на воровство *.kwm файла с последующей передачей его по сети. Почему-то разработчики думают, что это единственный способ взлома, хотя это далеко не так. Еще они советуют "правильно" настроить брандмауэр, чтобы предотвратить утечку информации и регулярно латать систему, чтобы не проникли ни хакеры, ни черви. Ну на счет брандмауэров они явно погорячились. Достаточно сходить на популярный сайт http://www.firewallleaktester.com/, чтобы убедиться, что существуют атаки, пробивающие ***все*** персональные брандмауэры. Я так же писал об этом в "записках исследователя компьютерных вирусов", фрагменты которой можно скачать с ftp://nezumi.org.ru, там же лежит готовый демонстрационный код.

Теперь разберемся с обновлениями. Многие сайты, принимающие оплату через WebMoney работают только с IE, потому что используют ActiveX. И хотя для альтернативных бразузеров типа Оперы и Лиса выпущены плагины, работают они кое-как и в реальности приходится использовать именно IE, количество дыр в котором достойно книги рекордов Гиннеса. То есть, создатели WebMoney сами подсаживают нас на дырявый браузер, и при этом еще заботливо рекомендуют, не забудь вовремя обновиться мол. А может, мне еще и пол сменить?! Так что проблема не в пользователях. Проблема в мозгах разработчиком (точнее, в их полном отсутствии). Проблема в концепции всей системы. Проблема в принципиальной уязвимости протокола передачи денег и незащищенности Keepr'a. Черт возьми, сколько лет уже существуют алгоритмы генерации "одноразовых" ключей, при котором воровать просто нечего и нечем. Но почему о них знаю я, — совсем далекий от криптографии и финансовых махинаций мыщъх, — но не знают разработчики платежной системы?! Понапринимали непонятно кого…

keeper light или борьба с сертификатами

Небезопасность классического Keeper'a — общепринятый факт, но Light все еще считается достаточно защищенным: "*В Keeper Classic файл с ключами можно по частям перетаскать, email можно взломать и т.д. Ключи, хранящиеся на сменном носителе, троян может переписать на винчестер в момент, когда дискета или CD вставлены. То есть теоретически возможно добраться до денег, хотя при соблюдении всех мер предосторожности — крайне сложно.* ***Но Light с не экспортируемым сертификатом дает 100%-ную гарантию безопасности***" (http://owebmoney.ru/cafe/index.php?showtopic=108).

Звучит заманчиво, но как с этим обстоят дела на практике? Попробуем разобраться. Начнем с вопроса — как все-таки работает Keeper Light? Очень просто. Секретный ключ теперь хранится не в *.kwm файле, а в специальном сертификате, а все управление идет через WEB-интерфейс по специальным криптографическим протоколам.

Где браузер хранит сертификаты? Зависит от самого браузера. Например, Mozilla – в каталоге "./mozilla/defaul/<blahblahblah>/cert8.db", а вот IE, запущенный под управлением Windows XP Professional, использует довольно навороченную систему. Сертификаты с открытыми ключами хранятся в персональном (personal) хранилище, расположенном в каталоге Documents-n-Settings\<username>\Application-Data\Microsoft\SystemCertificates\My\Certificates, которая свободна доступа всем желающим (ведь это открытая информация!). Сертификаты пользователя расположены в его профиле. Закрытые ключи хранятся в каталоге Documents-n-Settings\<username>\Application Data\Microsoft\Crypto\RSA. Все файлы, расположенные здесь, автоматически шифруются случайным симметричным ключом — основным ключом пользователя (user's master key), длинною в 64 символа. Основной ключ генерируется по алгоритму Triple DES на основе пользовательского пароля с которым он входит в систему.

Что значит вся эта теоретическая бодяга в практическом плане? А то, что стащить сертификат с закрытым ключом из-под Windows XP не удастся! То есть, стащить-то удастся, но толку от этого будет ноль, поскольку на чужом компьютере он просто не будет работать! (На то он и закрытый сертификат!). Правда, его можно экспортировать, даже не обладая никакими особенными привилегиями. Распотрошите программу Менеджера Сертификатов, если не знаете как. Собственно говоря, для переноса сертификатов с компьютера на компьютер Keeper Light использует экспортируемый сертификат, который хранится в файлах с расширением .pfx. Их можно встретить как на внешних носителях, так и на жестких дисках. Вот только здесь есть одно "но". Экспортируемый сертификат закрыт паролем, назначаемым пользователем, и чтобы его импортировать в свою систему необходимо либо закинуть клавиатурного шпиона, либо попробовать вскрыть пароль методом перебора. Но первое слишком заметно, второе — долго, поэтому кража сертификатов не получила большого распространения.

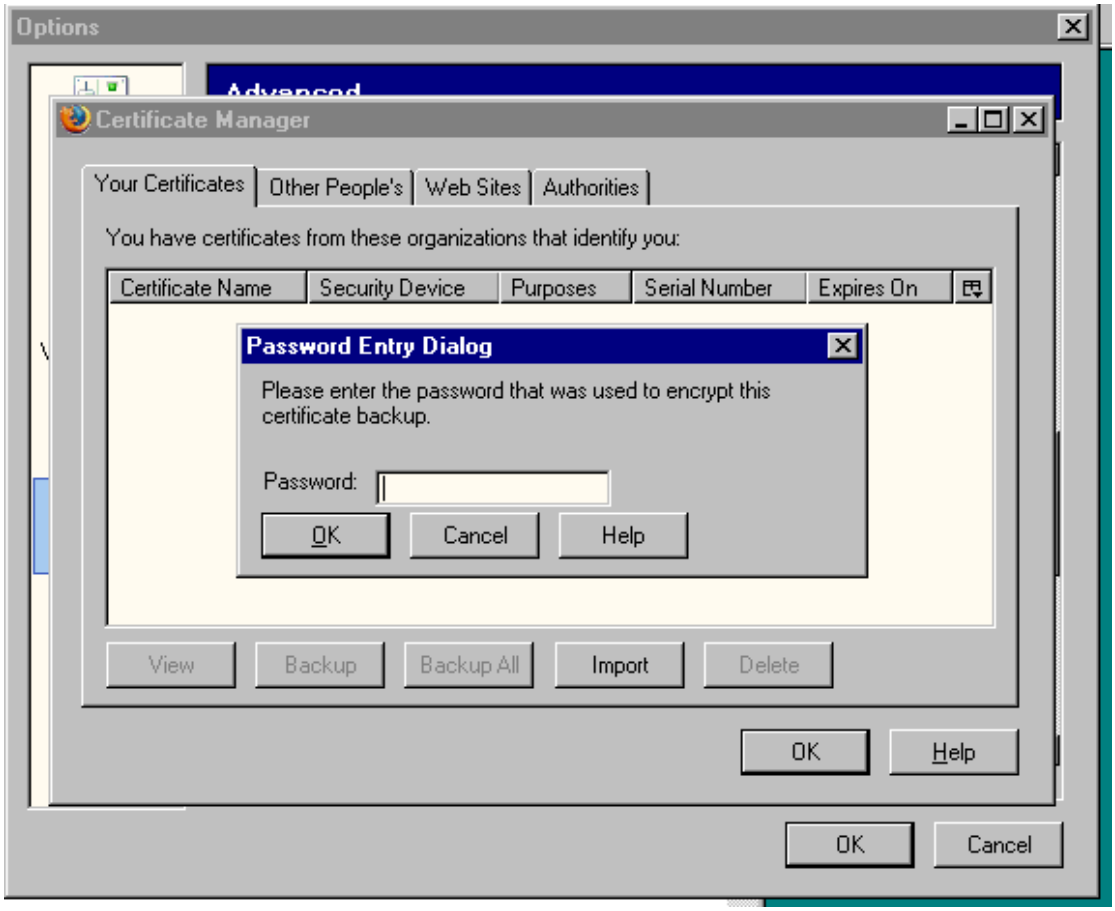

Рисунок 8 запрос пароля при импорте сертификата

Означает ли это, что Keeper Light защищен? Нет и еще раз нет!!! Если Keeper Classic можно защитить хотя бы теоретически (установить драйвер, обеспечивающий прямой клавиатурный ввод, отсекающий эмуляторы и следящий за целостностью Keeper'a и самого себя), то Keeper Light работает через браузер, "целостность" которого невозможно контролировать в принципе!

Первое, что приходит на ум — это уже упомянутая эмуляция. Говорим "start https://light.webmoney.ru", тем или иным способом прячем окно браузера (достаточно просто получить его дескриптор и можно рисовать поверх него, что попало) и эмулируем последовательность нажатия клавиш для пополнения электронного кошелька. Действует железно и неотвратимо. Единственный минус — каждый тип (и, возможно, версия) браузера требует своего подхода, но можно остановиться только на IE 5/6 как на наиболее популярном.

С остальными браузерами еще проще. Берем исходники Лиса и создаем хакерский мини-браузер на их основе, который ничего не выводит на экран, но с кошельками работает только так. Правда, среди пользователей WebMoney поклонников Лиса не так уж много, но это все же лучше, чем совсем ничего. Кстати, пусть приверженцы IE не чувствуют себя в безопасности. Исходные тексты W2K были украдены уже давно и создать свой клон IE на их основе вполне реально, не говоря уже о том, что IE это просто набор DCOM-объектов и собрать свой браузер на их основе сможет даже начинающий.

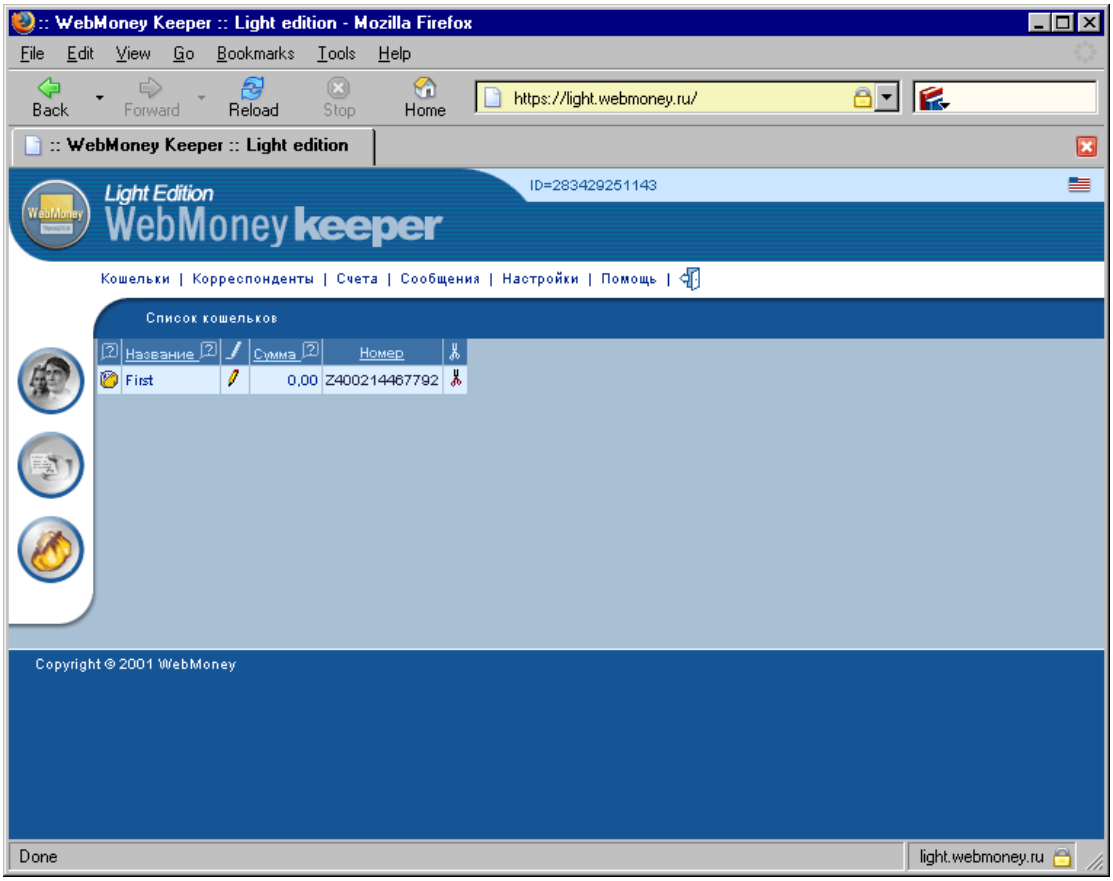

Рисунок 9 Keeper Light это просто WEB-интерфейс, позволяющий работать с кошельком через любой браузер

А что если импортировать сертификат перед каждым открытием кошелька, а затем удалять его из хранилища? Действительно, это до некоторой степени увеличит защищенность, однако, хакерская программа может либо дожидаться появления окна "WebMoney Keeper :: Light Edition", сигнализирующего о том, что пользователь вошел в систему, либо шпионить за клавишами, передавая секретный пароль вместе с сертификатом по сети. Так что, электронные деньги все равно остаются в щекотливой ситуации!

авторизация по сотовому телефону — надежна?

Последним писком моды стала система авторизация с помощью сотового телефона. При регистрации в службе ENUM (http://enum.ru/) нам на мобильник уставляется специальное Java-приложение (так же называемое мидлетом), называющее себя Enum Client. Он принимает пятизначные числа (например, 09652) и генерирует на их основе ответ, причем алгоритм генерации уникален для каждого пользователя. Если нет сотового телефона — подойдет Pocket PC или любое другое устройство с поддержкой Java (например, настолько PC, только смысла в нем будет немного).

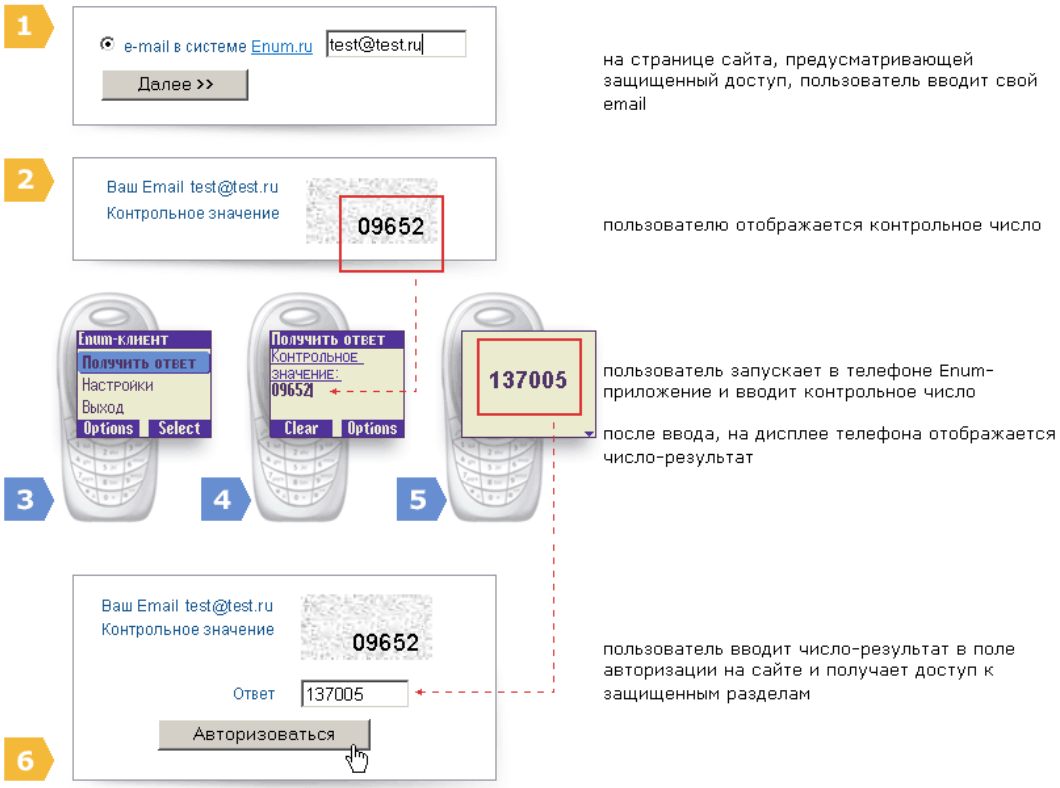

Рисунок 10 последовательно операций при активации платежа через сотовый телефон или КПК

Служба ENUM позволяет совершать покупки через сервис Merchant (https://merchant.webmoney.ru/) вообще не прибегая к Keeper'y — ни к классическому, ни к облеченному. Считается, что взломать электронный кошелек и похитить наличность в этом случае уже не удастся: "*Мошенники и вирусописатели используют Интернет для кражи с наших компьютеров ценной информации. Но какую бы защиту мы не изобрели — файрволлы, антивирусы, антикейлогеры, антитрояны, сертификаты — всегда есть теоретическая вероятность ее обхода и кражи паролей (или ключей Кипера, например) с компьютера, потому что и хакеры, и защитные инструменты используют ОДИН И ТОТ ЖЕ канал — Интернет. И проблема Интернета состоит в том, что нет другого, альтернативного канала хранения-передачи информации. Так вот, ENUM эту проблему решает. Он предоставляет нам тот самый другой канал. Хакер может влезть на ваш компьютер, "подсадить" троянский вирус, но он не сможет влезть в ваш мобильный телефон. Угадать же, по какому уникальному для каждого пользователя алгоритму Enum Client из одного числа получает другое, тоже нельзя*" (http://owebmoney.ru/enum.shtml).

Рисунок 11 логотип системы ENUM

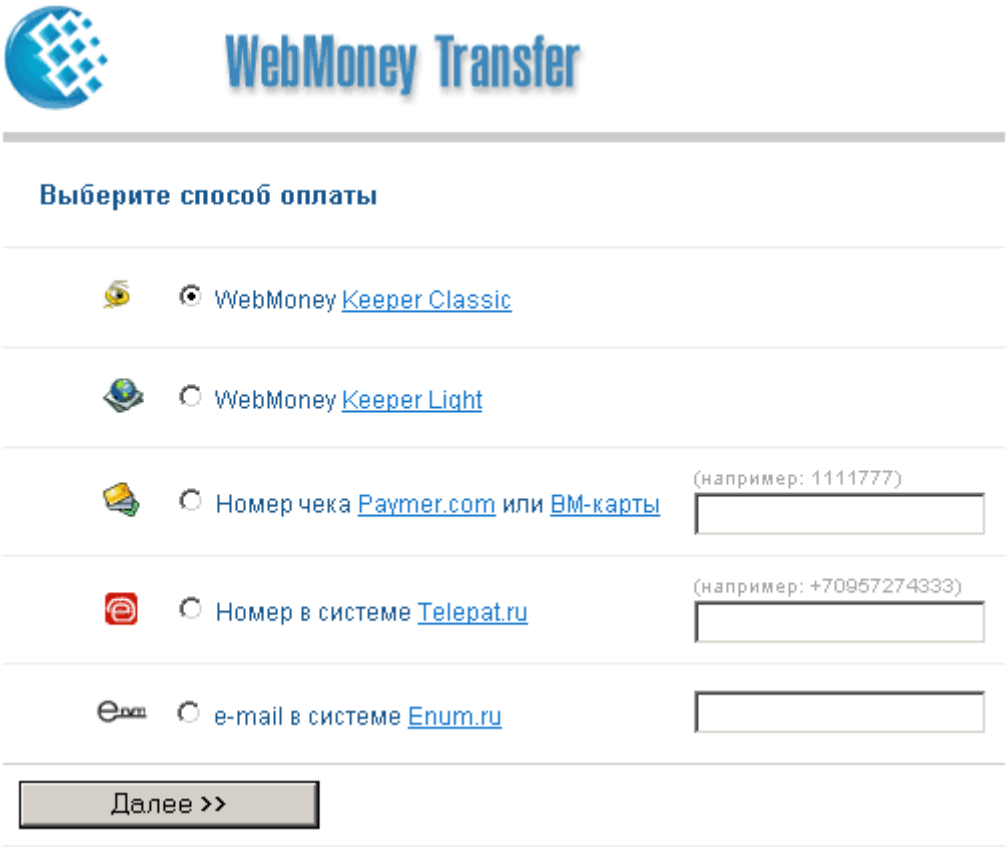

Рисунок 12 служба Merchant

Действительно ли это так? Как говориться, "если нельзя, но очень хочется, то все-таки можно". Дополнительный "канал связи" и в самом деле многократно усиливает безопасность, но говорить о принципиальной невозможности взлома — преждевременно. Начнем с того, что ***алгоритм генерации номеров для всех пользователей все-таки един*** (дизассемблируйте мидлет, если не верите), только ключ генерации разный и подобрать его вполне возможно. Достаточно перехватить один-единственный отклик для данной комбинации цифр. Восстановление ключа не займет много времени и троянской программе это вполне по силам. Надеюсь, не нужно объяснять как считать комбинацию цифр из окна редактирования.

К тому же, сотовые телефоны содержат кучу дыр. ИК-протоколы и Голубой Зуб буквально кишат ими. Журнал Хакер неоднократно писал об этом. Если жертва имеет сотовый телефон или КПК, то возможно она имеет и адаптер Голубого Зуба или ИК, который держит постоянно включенным. Злоумышленник может передавать телефону любые AT-команды, выполнять мидлеты или считывать их содержимое. А что?! Можно приколоться и написать вирус, похищающий электронные кошельки и передающий их через сотовый телефон! В обход всех брандмауэров! Вот вам и дополнительный канал связи!

Впрочем, все это придирки старого мыщъх'астного хакера. Настроение просто плохое. Идет дождь, и от депрессии спасает только Sirenia (очень мощная готическая группа из далекой Норвегии — рекомендую). Если трезво смотреть правде в глаза (в такие красные мыщъх'иные глаза, маленькие словно бусинки), необходимо признать, что хакнуть ENUM очень сложно, так что определенный смысл в нем все-таки есть. Но это не значит, что можно заводить электронный кошелек и смело класть на него $100.000. Тогда точно взломают!

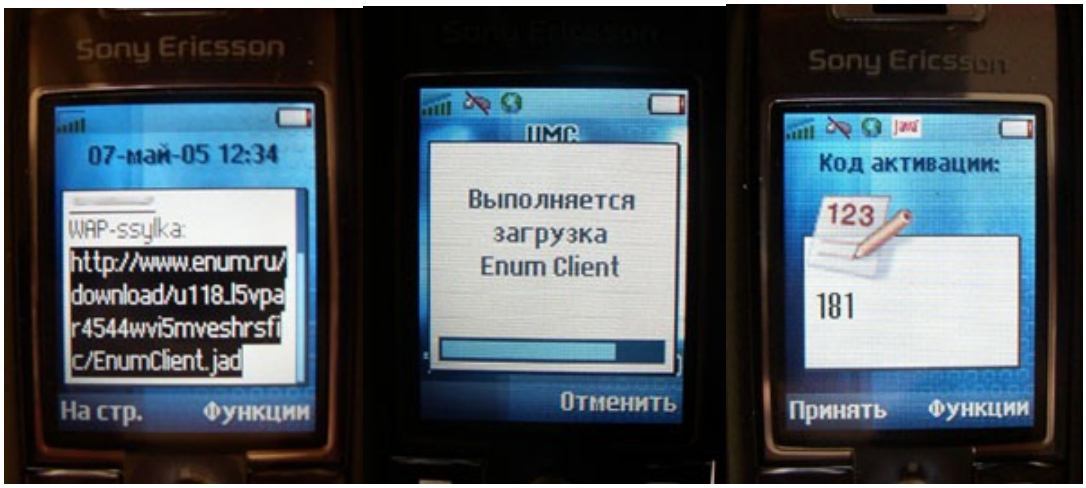

Рисунок 13 активация платежа по сотовому телефону через систему ENUM

бои в памяти — кровавая война продолжается

>>> врезка как ломают обменники

Взлом обменников не входит в наши планы (свой собственный обменник имеет далеко не каждый, а ломать чужие — незаконно), поэтому отметим лишь основные пункты. С хакерской точки зрения обменник представляет сайт, как правильно управляемый PHP и работающий под LINUX/BSD/NT.

Вот через ошибки в PHP-скриптах их чаще всего и ломают. Так же, некоторые web-программисты оставляют "черный ход" на тот случай если им вдруг захочется кушать, а кушать будет нечего. Реже ломают ось. Наибольшее количество дыр, естественно, имеет NT и все производные от нее системы (в том числе и хваленный Windows 2003 Server). LINUX и BSD подломать чуть-чуть сложнее, но… если взять сканер безопасности (например, X-Spider), то можно обнаружить, что на многих из них стоит корявый SendMail или проржавевший Apache. Переполнение буфера, засылка shell-кода и сервер в наших руках!

Рисунок 14 рабочее место хакера

заключение

Взлом WebMoney это не миф, а суровая реальность и обезопасить себя на 100% нельзя, даже если вы эксперт по безопасности. Всегда существует риск подхватить вируса через еще неизвестную дыру в операционной системе или браузере, причем, если от потери оперативных данных на винчестере спасает резервирования, от раскрытия конфиденциальных данных — физическое отключение Интранета от Сети, то от кражи электронных денег не спасает ничто!

Рисунок 15 Sireina – готическая хакерская музыка, что колбасит мыщъх'а