чемоданчик хакера

крис касперски, ака мыщъх, no-email

чем вообще ломают программы? правильно, головой, задницей и руками. а только потом программами. но первичны все-таки программы, поскольку именно они формируют сознание, позволяя начинающему сделать свои первые шаги в дремучем лесу машинных кодов. вот только этих программ _настолько_ много, что новичок, попавший на хакерский сайт оказывается в полной растерянности — что качать, а что не надо. мыщъх дает предельно сжатый обзор хакерского софта, покрывающий потребности практически любого взломщика.

Рисунок 1 чемоданчик хакера в минимальной комплектации

отладчики

Лучший отладчик всех времен и народов это, конечно же, **soft-ice**, на котором выросло не одно поколение хакеров. Это интерактивная программа с развитым командным интерфейсом,

представляющим собой компромисс между легкостью освоения и удобством использования. Другими словами, руководство читать обязательно. Никаких интуитивно-понятных менюшек в стиле Turbo-Debugger здесь не будет.

Изначально созданный фирмой Nu-Mega, soft-ice был продан компании Compuware, долгое время распространяющей его в составе уродливо framework'a DriverStudio, 3 апреля 2006 по малопонятным причинам компания объявила о прекращении работы над продуктом, похоронив тем самым уникальнейший проект. Последняя версия DriverStudio 3.2 поддерживает всю линейку Windows вплоть до Server 2003, а так же архитектуру AMD x86-64. То есть лет на пять запаса прочности у soft-ice еще должно хватить, а там… мы что-нибудь придумаем.

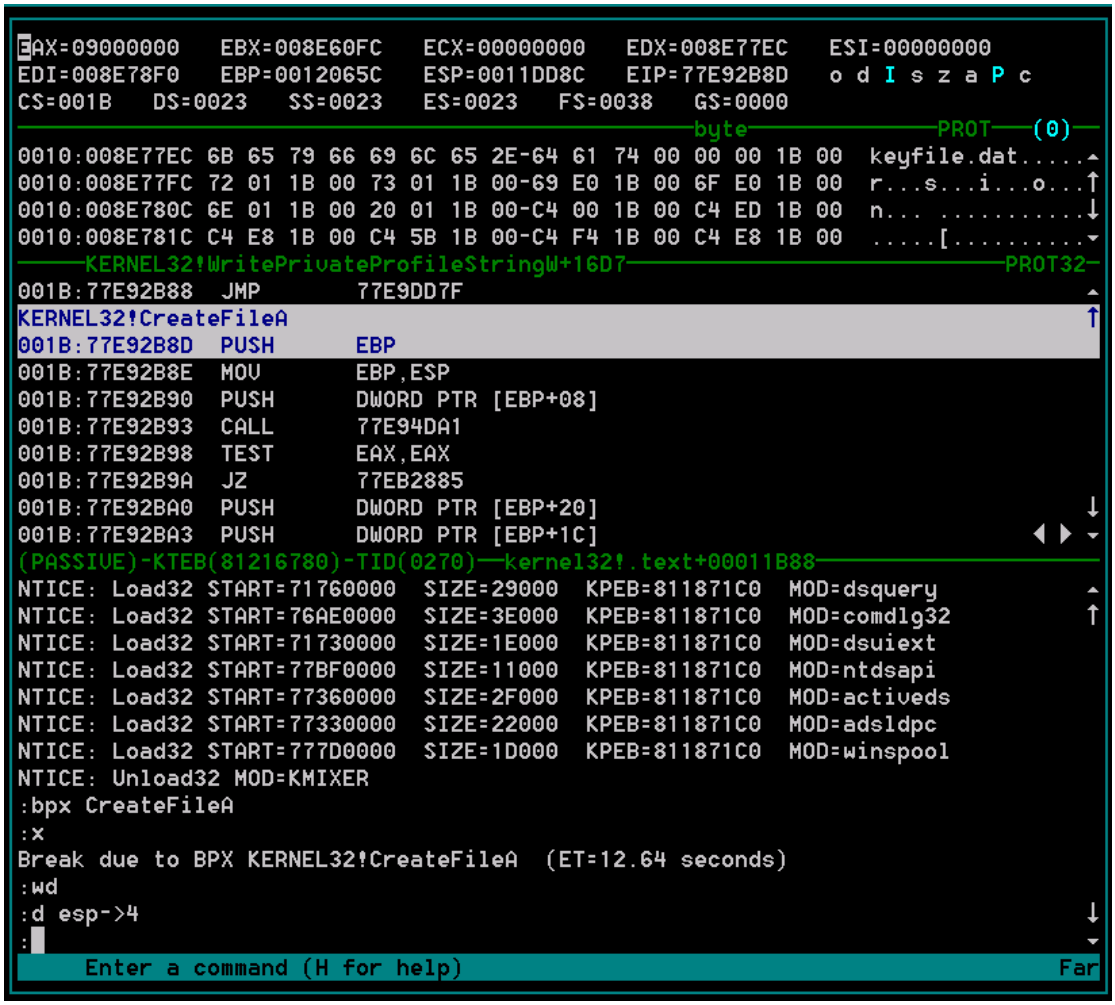

Рисунок 2 внешний вид отладчика soft-ice

Найти soft-ice можно на любом хакерском сайте или в Осле. Чтобы не качать всю судию целиком (это же без малого 200 метров), можно воспользоваться пакетом DeMoNiX'a (reversing.kulichki.net) — содержащим в себе один лишь soft-ice, выдернутый из Driver Studio v2.7 build 562, и занимающий всего 2,27 Мбайт. Однако, инсталлятор содержит ошибки, а старая версия не поддерживает новых веяний Microsoft (хотя замечательно идет под W2K, мыщъх вообще работает с build'ом 334 и полностью им удовлетворен).

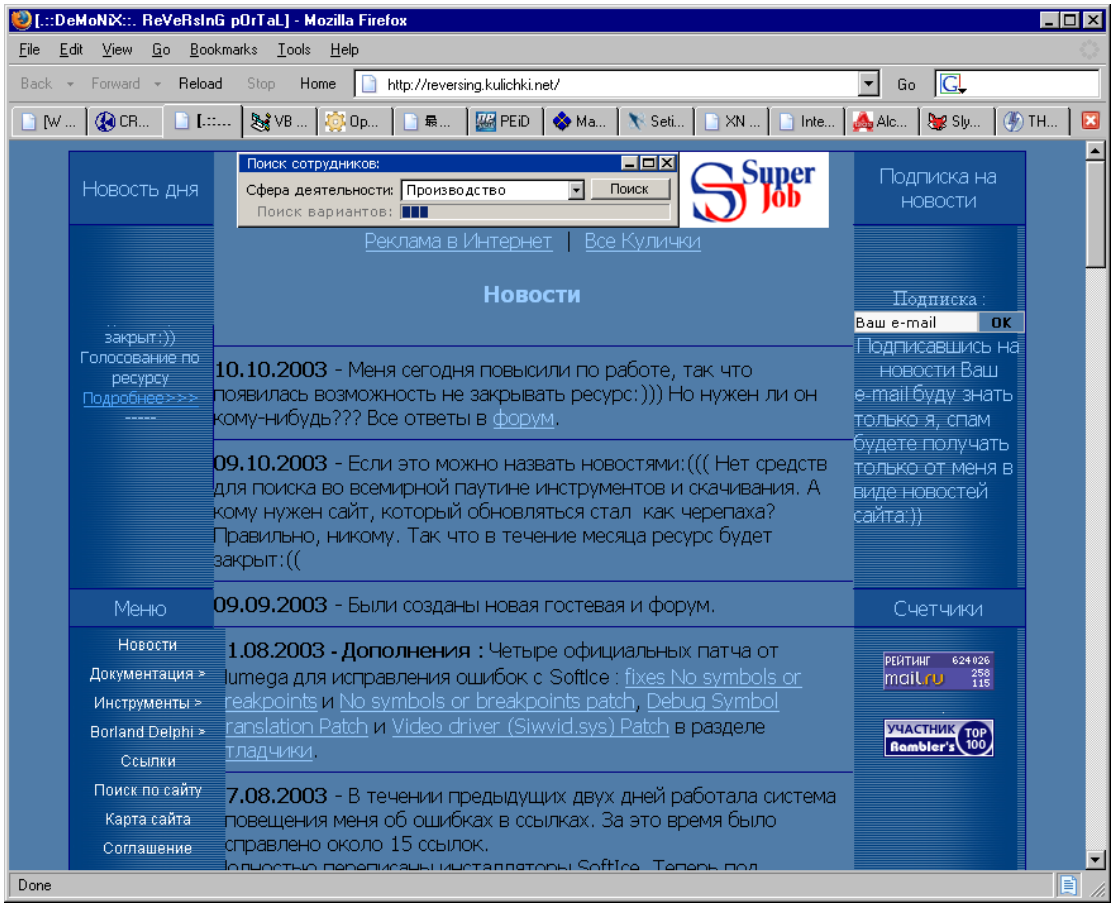

Рисунок 3 сайт DeMoNiX'a

Вместе с soft-ice желательно сразу же установить **IceExt** (sourceforge.net/projects/iceext) — неофициальное расширение, позволяющее скрывать отладчик от взора большинства защит, дампить память, задействовать кириллические кодировки 866/1251, приостанавливать потоки и делать множество других вещей (например, играть в тетрис).

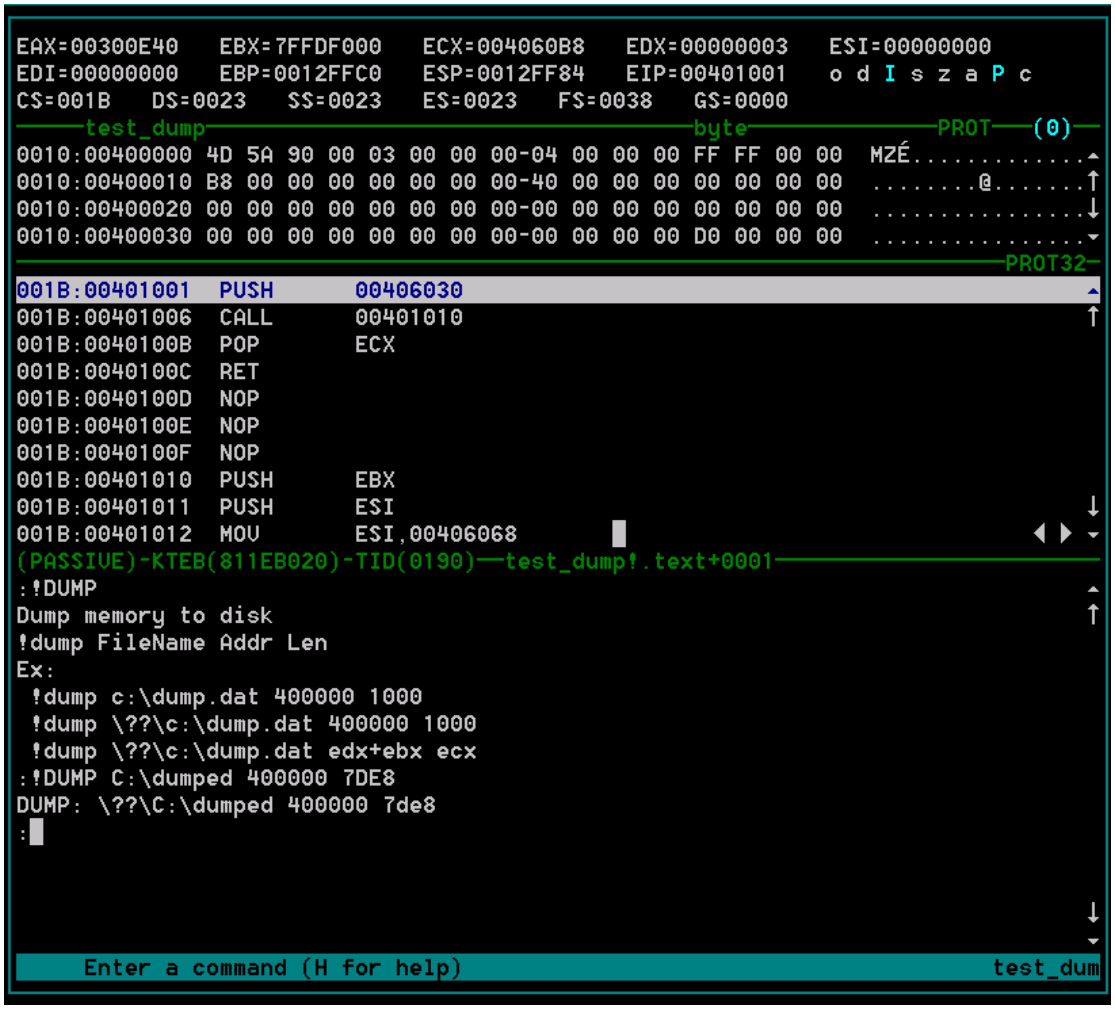

Рисунок 4 снятие дампа с помощью IceExt

Если IceExt откажется запускаться, скорректируйте следующие ключи в данной ветви системного реестра: HKLM\SYSTEM\CurrentControlSet\Services\NTice: KDHeapSize (DWORD): 0x8000; KDStackSize (DWORD): 0x8000.

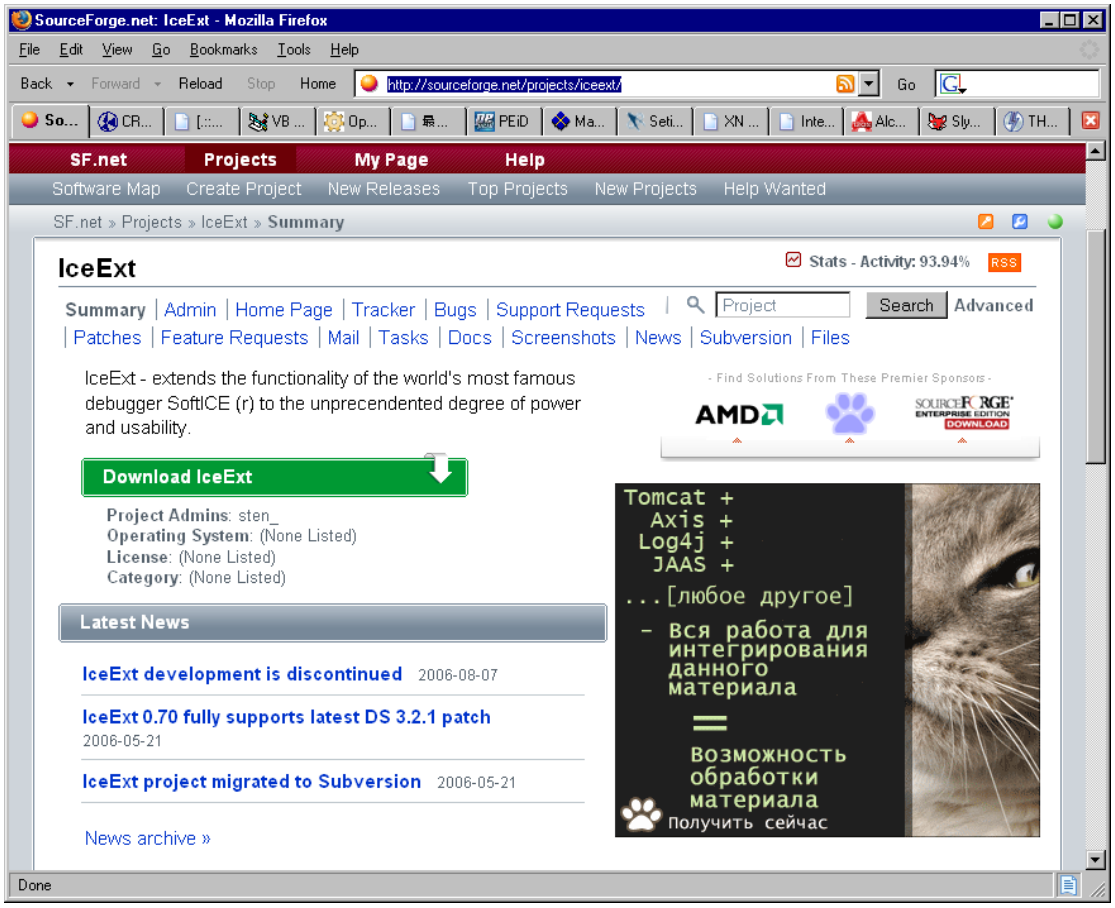

Рисунок 5 отсюда раздают IceExt

Другое неофициальное расширение для soft-ice **IceDump** (programmerstools.org/system/files?file=icedump6.026.zip) так же умеет делать много полезных вещей, и удачно дополняет IceExt.

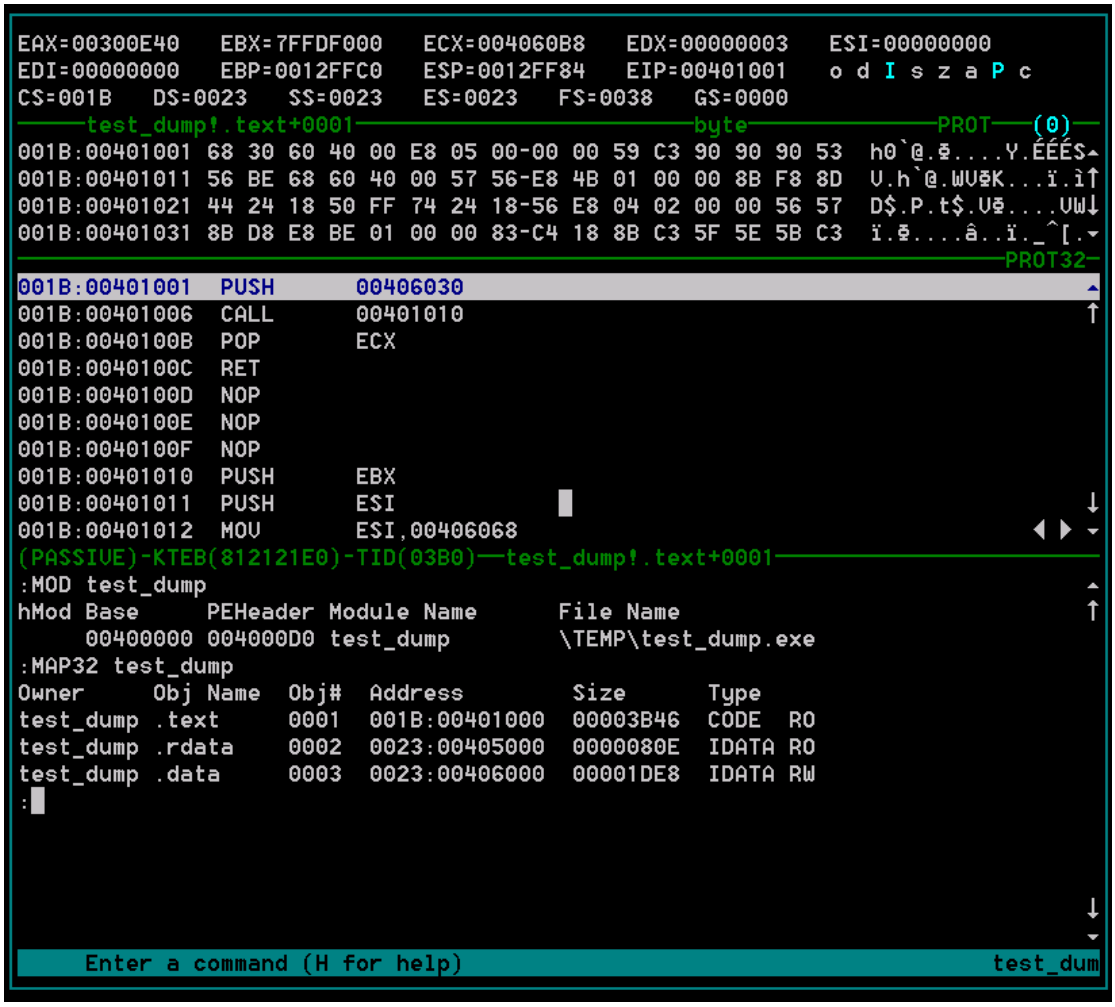

Рисунок 6 снятие дампа с помощью IceDump

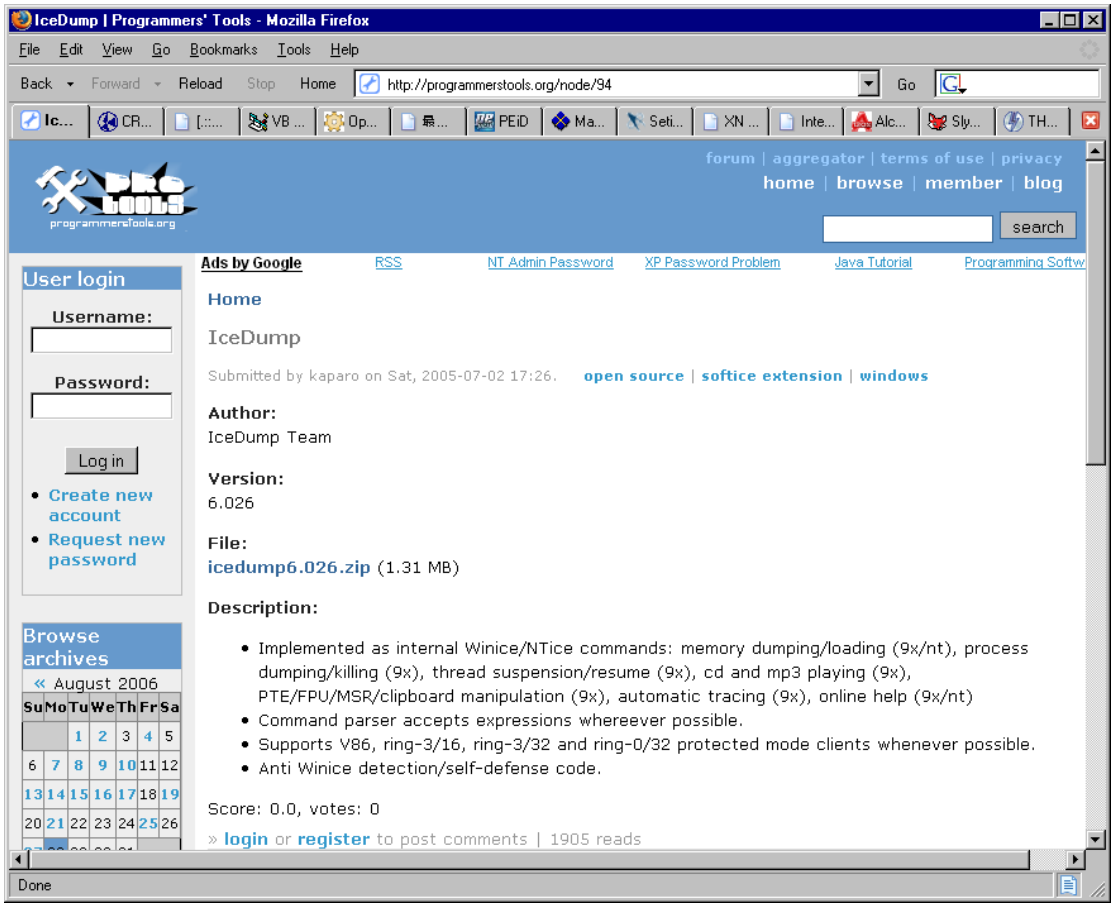

Рисунок 7 IceDump на programmerstools

Кстати, сам soft-ice замечательно работает под виртуальной машиной VM Ware, для этого достаточно добавить в vmx-файл пару строк: paevm = TRUE и processor1.use = FALSE. Так же отмечены проблемы с многоядерными и HT-процессами (хотя и не у всех). Лечится путем отрубания всего этого хозяйства через добавление ключа /ONECPU в файл boot.ini.

Помимо soft-ice существуют и другие отладчики, из которых в первую очередь хотелось бы отметить бесплатный **Olly-Debugger** (www.ollydbg.de). Это удобный инструмент прикладного уровня, ориентированный на хакерские нужны, поддерживающий механизм планинов и собравший вокруг себя целое сообщество, написавшее множество замечательных расширений и дополнений, прячущих OllyDbg от глаз защит, автоматически определяющих оригинальную точку входа в упакованной программе, облегчающих снятие протекторов и т. д. и т. п.

Неплохую коллекцию плагинов можно найти на wasm'еи на www.openrce.org.

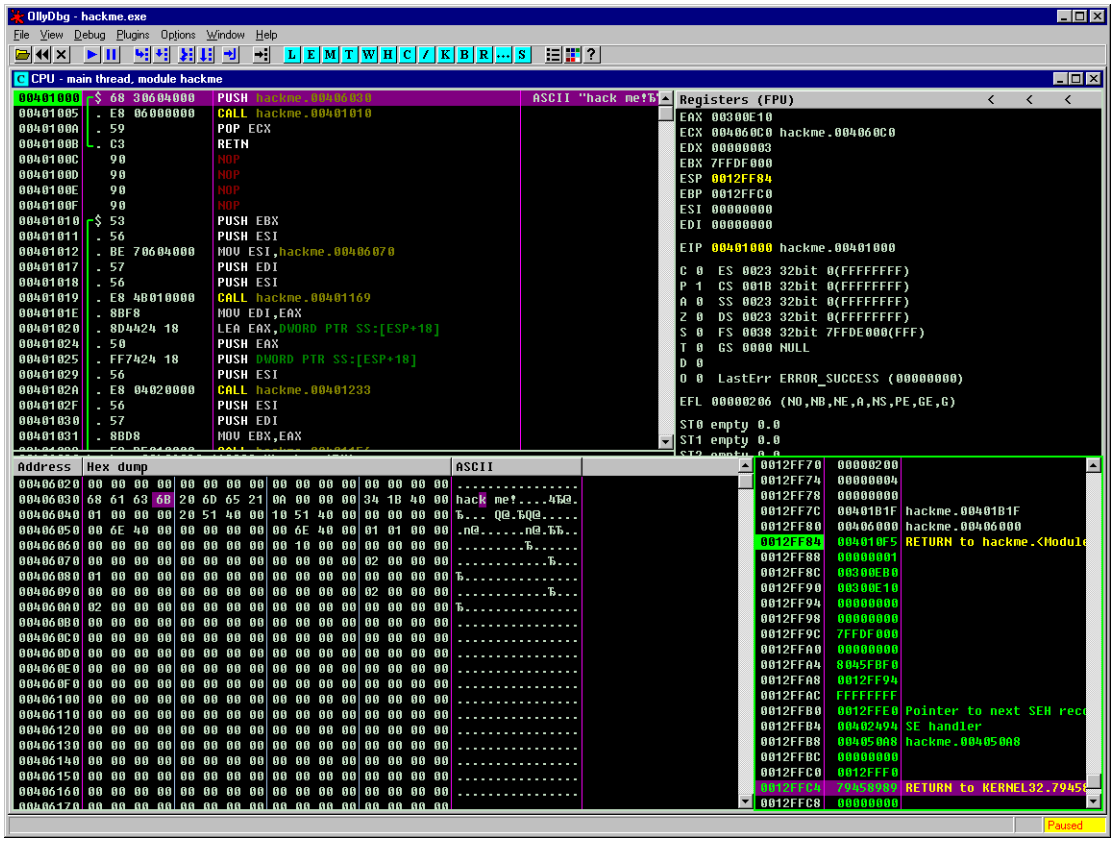

Рисунок 8 внешний вид отладчика OllyDbg

Самый свежий (и пока еще во многом экспериментальный) ядерный отладчик это, бесспорно, **SYSER** (www.sysersoft.com), выпущенный нашими китайскими братьями и в настоящее время переживающий стадию активного развития и становления.

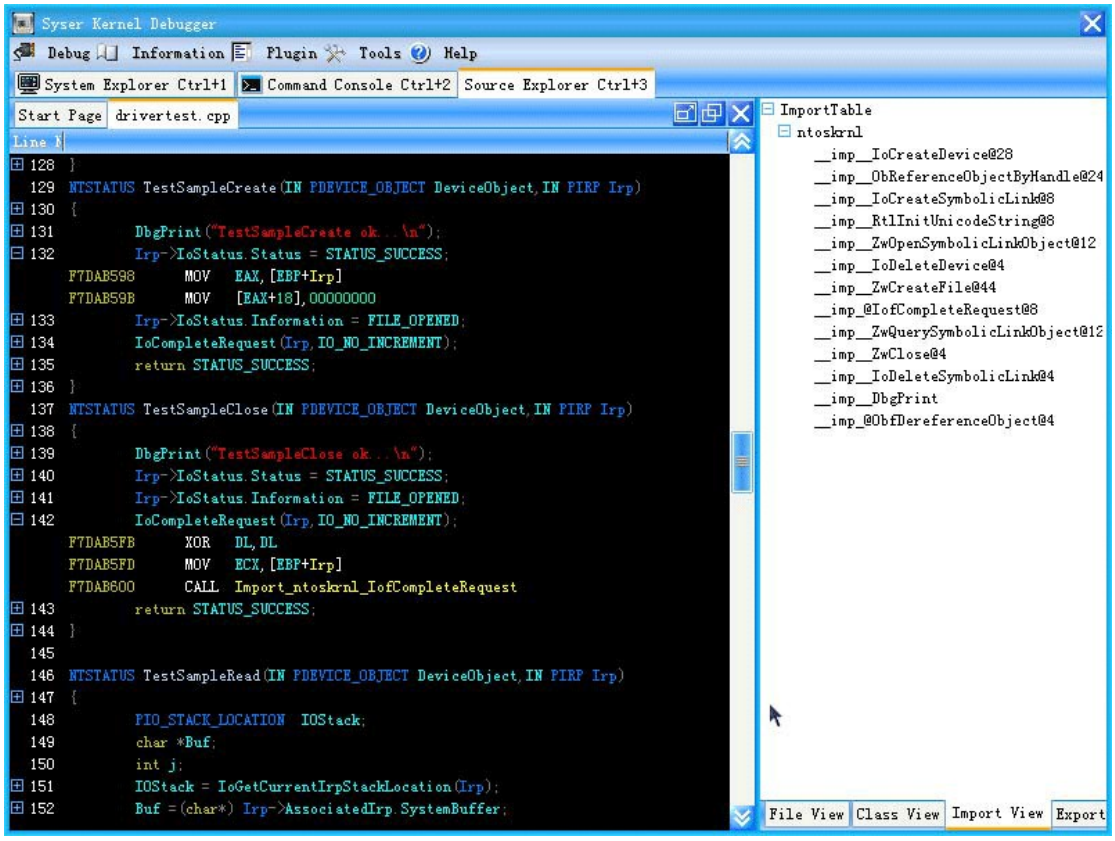

Рисунок 9 отладчик SYSER за отладкой термоядерного драйвера

Довольно много народа пользуется **Microsoft WinDeb**, входящим в состав бесплатного набора Debugging Tools. Он вполне пригоден для взлома только… уж очень неудобен для тех, кто привык к черному экрану soft-ice.

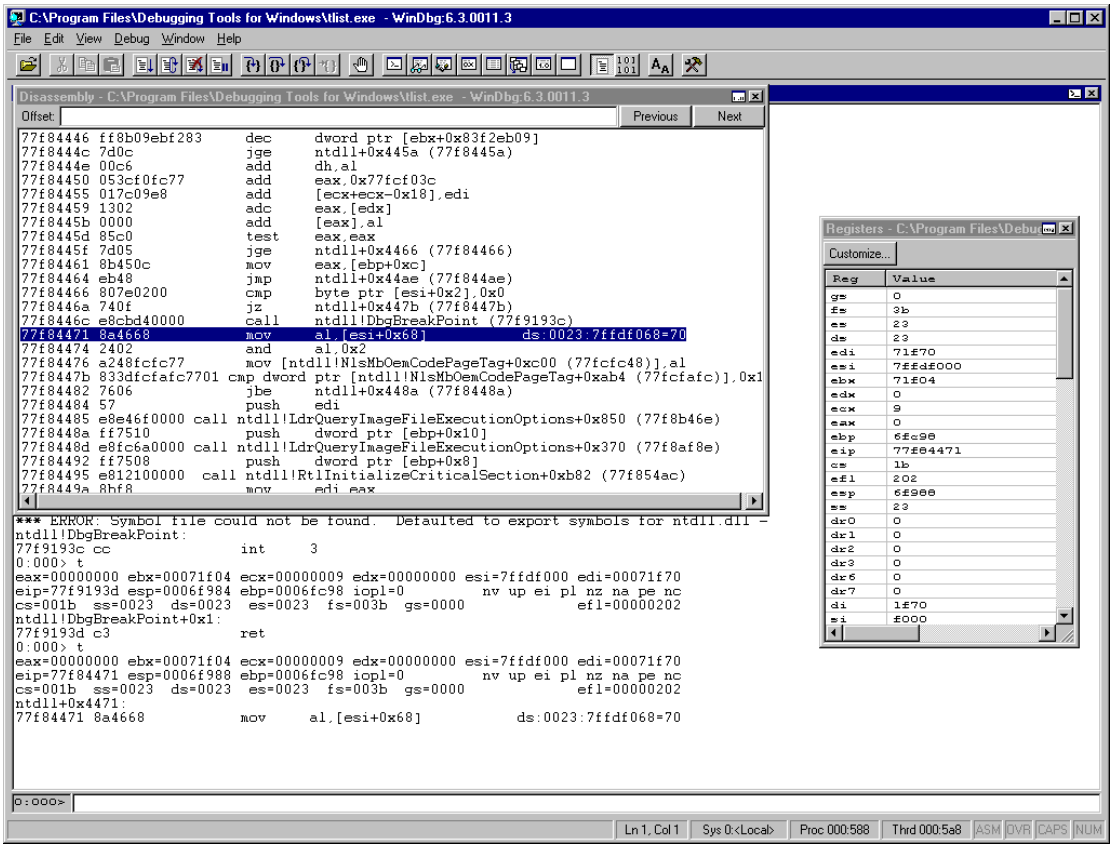

Рисунок 10 отладчик WinDeb

дизассемблеры

Существует всего лишь один дизассемблер, пригодный для _профессиональной_ работы — **IDA Pro** (www.idapro.com), стоящий немереных денег, но как и всякое другое добро, свободно валяющийся в Осле.

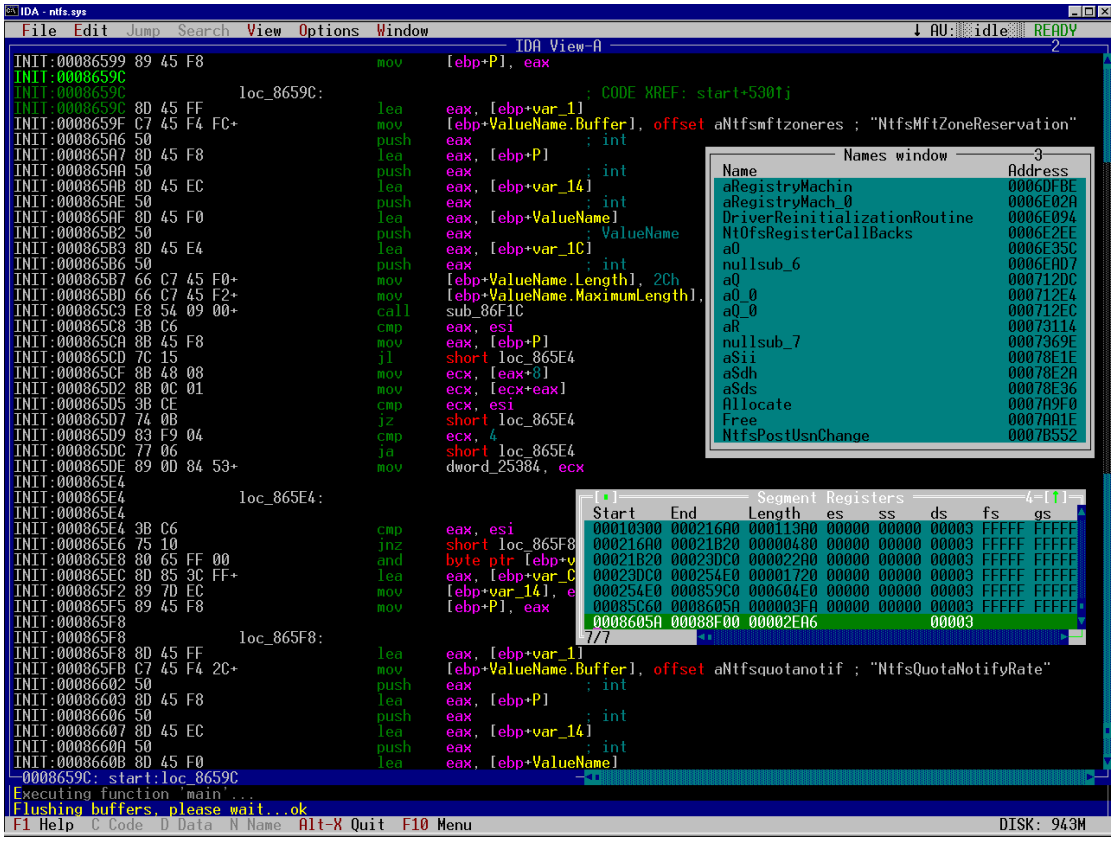

Рисунок 11 консольная версия IDA Pro

IDA Pro переваривает огромное количество форматов файлов и множество типов процессоров, легко справляясь с байт-кодом виртуальных машин Java и .NET, поддерживает макросы, плагины и скрипты, содержит интегрированный отладчик, работает под MS-DOS, Windows, LINUX и обладает уникальной способностью распознавать имена стандартных библиотечных функций по их сигнатурам.

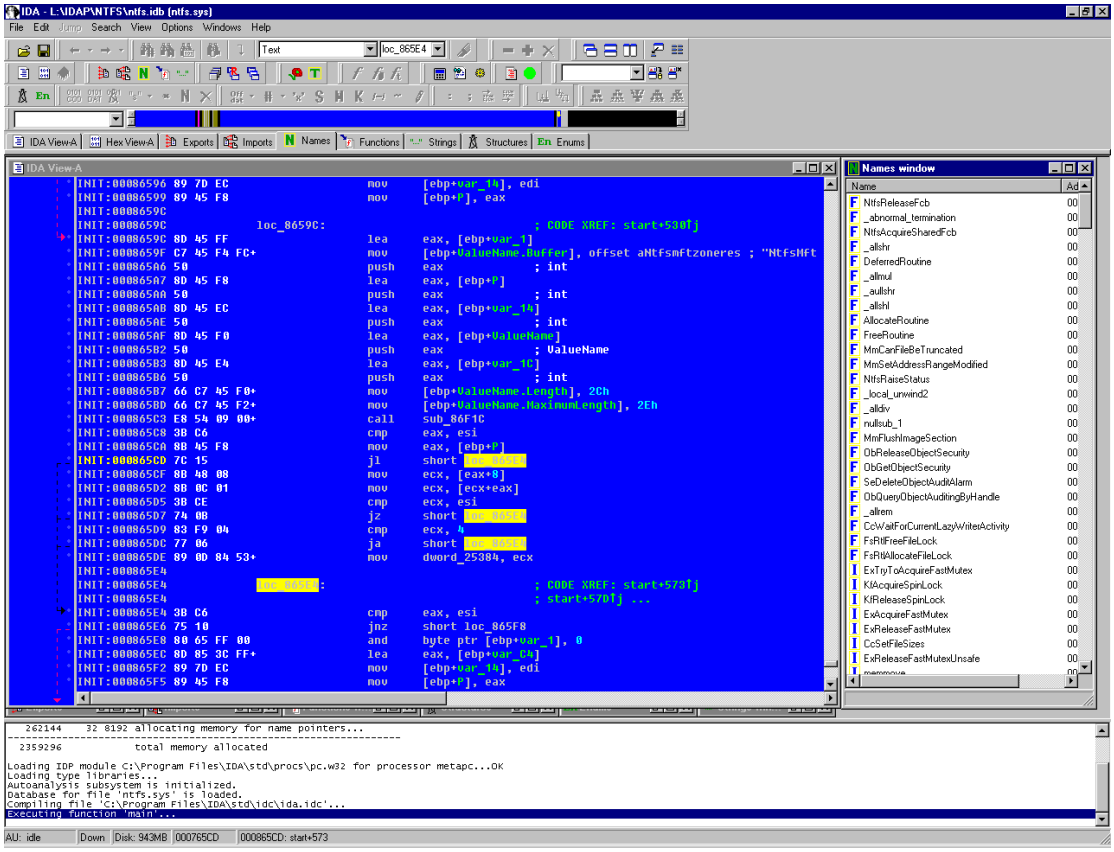

Рисунок 12 графическая версия IDA Pro

Самое главное — IDA Pro – это _интерактивный_ дизассемблер, то есть ***инструмент***, позволяющий работать с двоичным файлом, мыслить и творить, а не тупой автомат, заглатывающий хакаемую программу и выплевывающий "готовый" дизассемблированный листинг, в котором все дизассемблированного неправильно.

В последних версиях IDA PRO сделаны определенные подвижки в сторону автоматической распаковки файлов и снятия обфускаторов. Внушительную коллекцию плагинов и скриптов можно найти как на официальном сайте, так и на www.openrce.org.

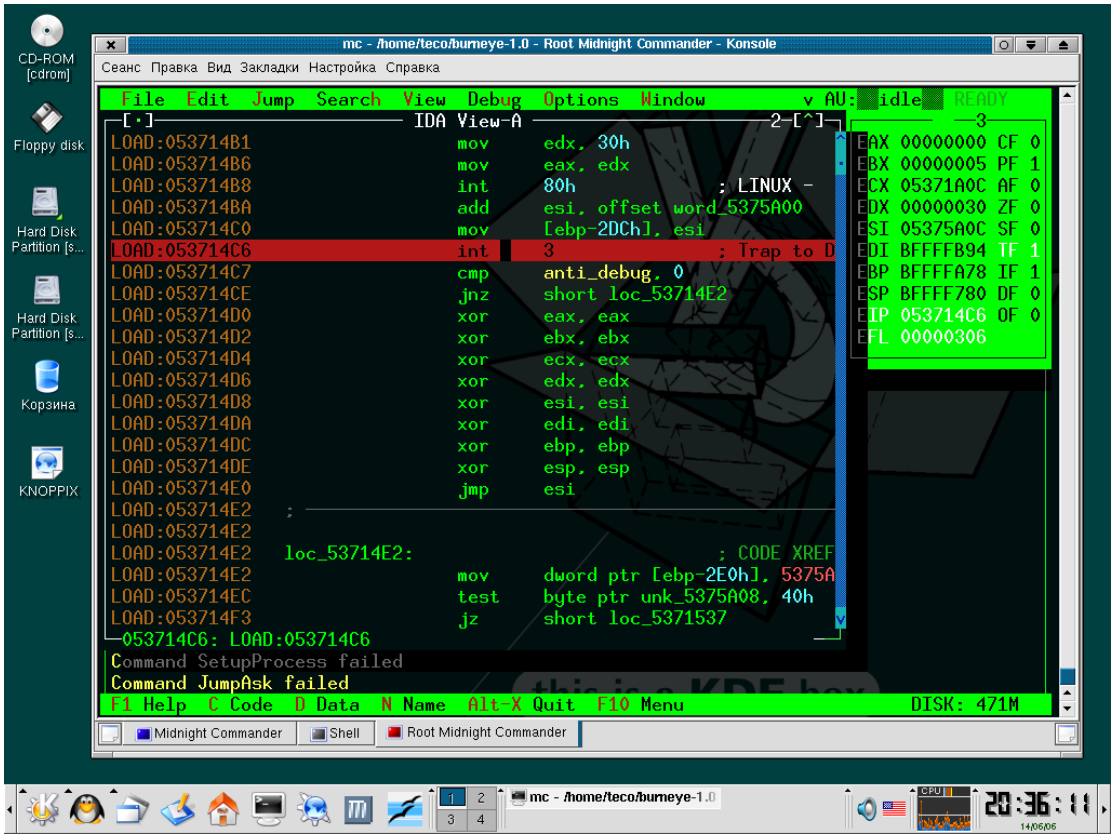

Рисунок 13 IDA Pro под LINUX

Конкурентам до IDA Pro еще расти и расти как до Луны. Тем не менее, народ активно качает бесплатный (ныне заброшенный) дизассемблер и отладчик в одном "флаконе" — WDasm: www.wasm.ru/baixado.php?mode=tool&id=178 и, судя по всему остается доволен, хотя чтобы взломать с его помощью что-то серьезное, легче задницу напополам разорвать.

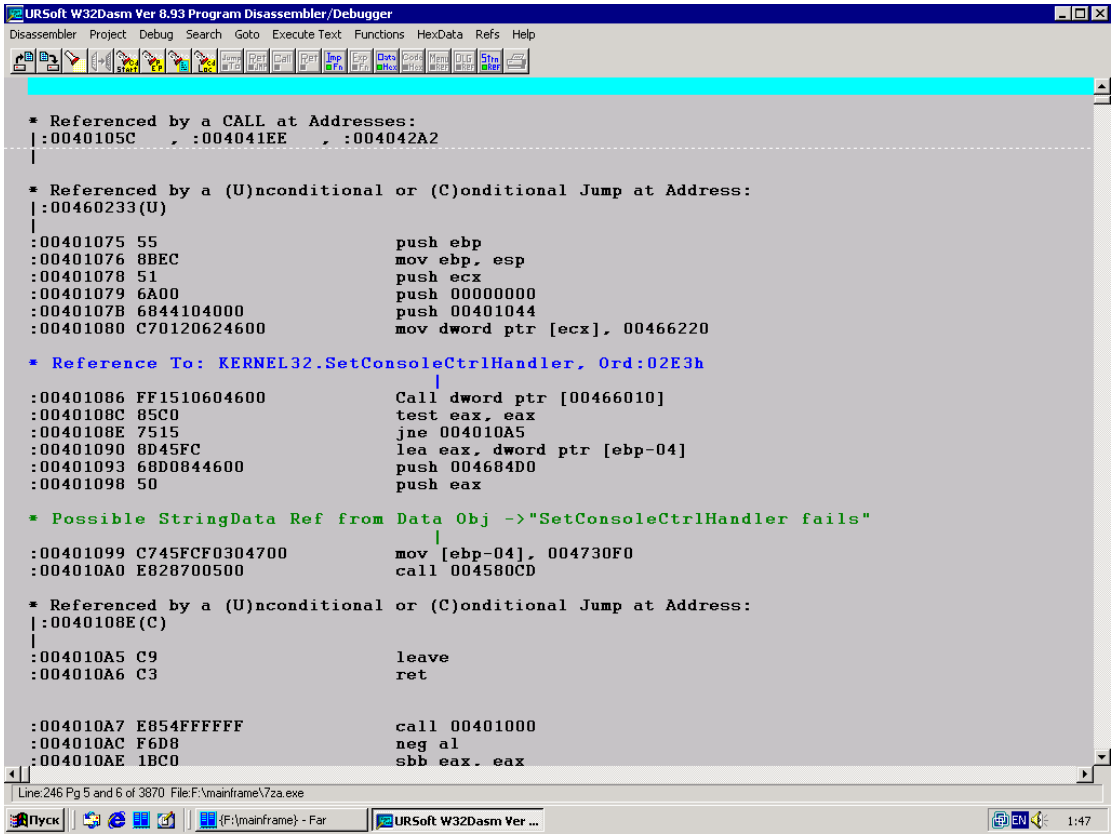

Рисунок 14 дизассемблер WDASM

Остальные дизассемблеры выглядят еще более убого, поэтому не будем их рассматривать, разве что отметим **Hacker Disassembler Engine** (patkov-site.narod.ru/lib.html), представляющий собой дизассемблер длин, распространяющийся в исходных текстах и предназначенный для встраивания в различные хакерские программы, занимающиеся перехватом функций, автоматической распаковкой, генерацией полиморфного кода и т. д.

декомпиляторы

Декомпиляцией называется процесс получения исходного текста программы (или нечто очень на него похожего) из двоичного файла. В полном объеме декомпиляция невозможна в принципе, поскольку компиляция — однонаправленный процесс, причем с потерей данных. Однако, декомпиляторы все-таки существуют и со своей задачей достойно справляются.

Для программ, написанных на DELPHI и Borland Builder с использованием RTTI, возможно восстановить исходную структуру классов вплоть до имен функций-членов, а так же реконструировать формы и "вычислить" адреса обработчиков каждого из элементов. Допустим, у нас имеется диалоговое окно "registration" с кнопкой "OK" и мы хотим знать, какая процедура считывает серийный номер и что с ним делает. Нет ничего проще! Берем бесплатный **DeDe** (programmerstools.org/node/120), декомпилируем программу и вперед!

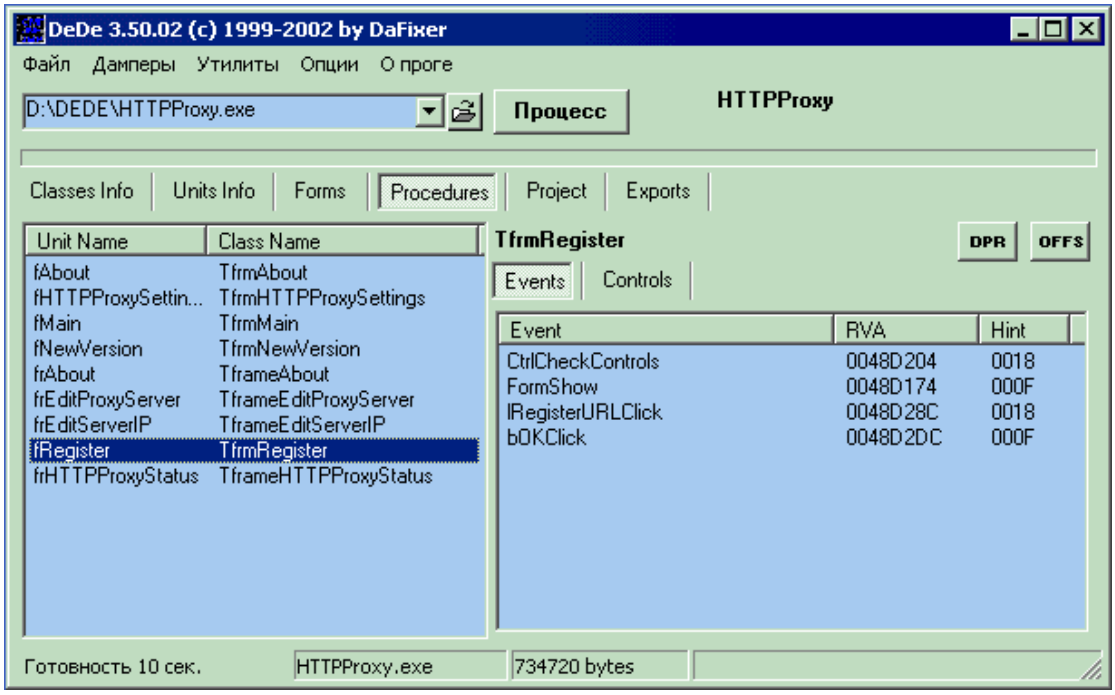

Рисунок 15 декомпилятор DeDe ломает HTTPProxy

Для Visual Basic'a существует свои декомпиляторы, лучшим из которых считается **VB Decompiler** от GPcH (www.vb-decompiler.org/index.php?p=Products). Другие бейсик-декомпиляторы: VB RezQ (www.vbrezq.com/), VBDE (programmerstools.org/node/129) и Spices.Decompiler (programmerstools.org/node/635) так же полезно положить в свой хакерский чемоданчик.

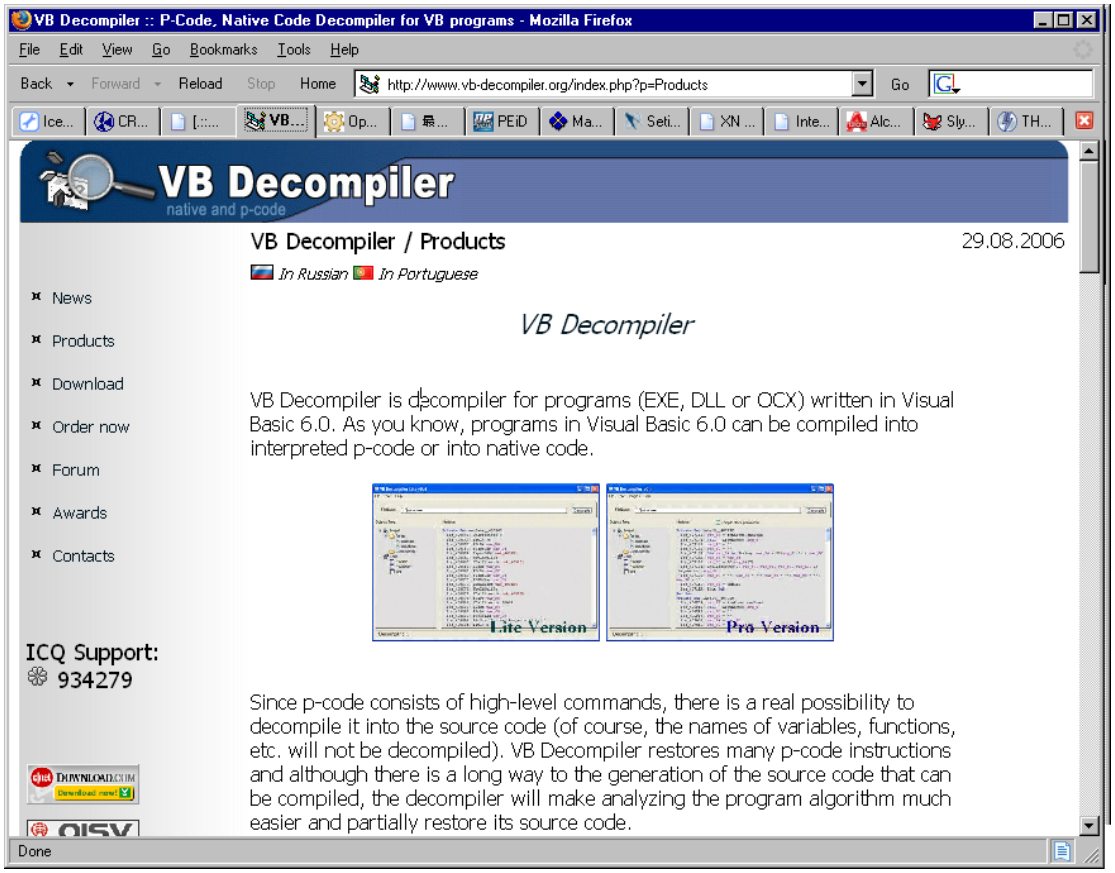

Рисунок 16 официальный сайт одного из лучших декомпиляторов Visual Basic'a

Большой интерес представляют декомпиляторы инсталляторов, поскольку многие проверки (на истечение трального срока, на серийный номер или ключевой файл) производятся как раз на стадии инсталляции. Самый популярный инсталлятор Install Shield и вот куча декомиляторов к нему: **InstallShield X Unpacker** (programmerstools.org/node/154), **Windows Installshield Decompiler** (programmerstools.org/node/118), **InstallShield Decompiler** (programmerstools.org/node/114) и всякая мелочь типа isDcc (programmerstools.org/node/115).

Что же касается Java и платформы .NET, то с ними замечательно справляется IDA Pro, а если ее под рукой нет, можно воспользоваться специализированными декомпиляторами, которые можно найти на сайтах www.cracklab.ru и www.wasm.ru вместе с декомпиляторами Fox Pro, Clipper'a и прочей экзотики.

hex-редакторы

Давным-давно hex-редакторы представали собой простые программы, умеющие всего лишь отображать двоичный файл в шестнадцатеричном виде и править байты по указанным адресам (кстати, вместо них часто использовался редактор диска Norton Disk Editor), но со временем они обросли дизассемблерами, ассемблерами, встроенными калькуляторами, функциями регулярного поиска, научились работать с блоками, понимать различные форматы файлов и даже расшифровывать/зашифровывать фрагменты кода/данных. В общем, эдакий швейцарский ножичек с шестнадцатью лезвиями.

Наибольшую популярность завоевал **HIEW** (webhost.kemtel.ru/~sen). Вплоть до версии 6.11 (поддерживающей MZ/PE/NE/LE/ELF-форматы) он распространяется на бесплатной основе, а теперь за него просят денежку, которую мыщъх платить не хочет и продолжает пользоваться своей любимой 6.04, в которой гораздо меньше ошибок и багов.

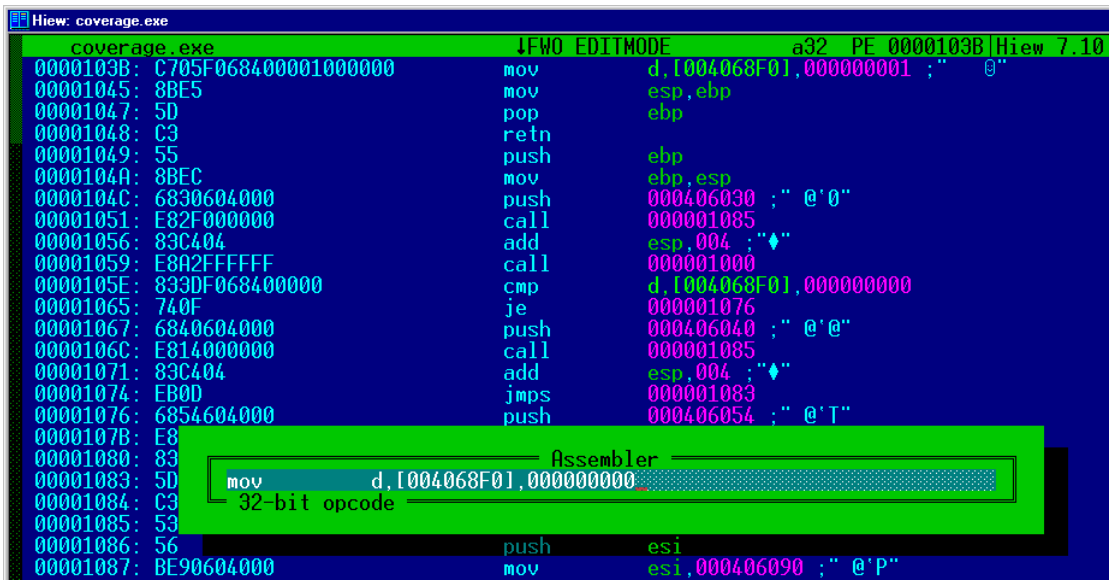

Рисунок 17 старый добрый hex-редактор hiew

Другой хороший редактор, по своим возможностям не только не уступающий hiew'y, но даже превосходящий его — это **HTE** (hte.sourceforge.net), распространяющийся в исходных кодах на бесплатной основе и в отличии от hiew'a позволяющий выбирать способ ассемблирования инструкции (если инструкция может быть ассемблирована более чем одним путем), а так же поддерживающий мощную систему перекрестных ссылок, вплотную приближающий его к IDA Pro.

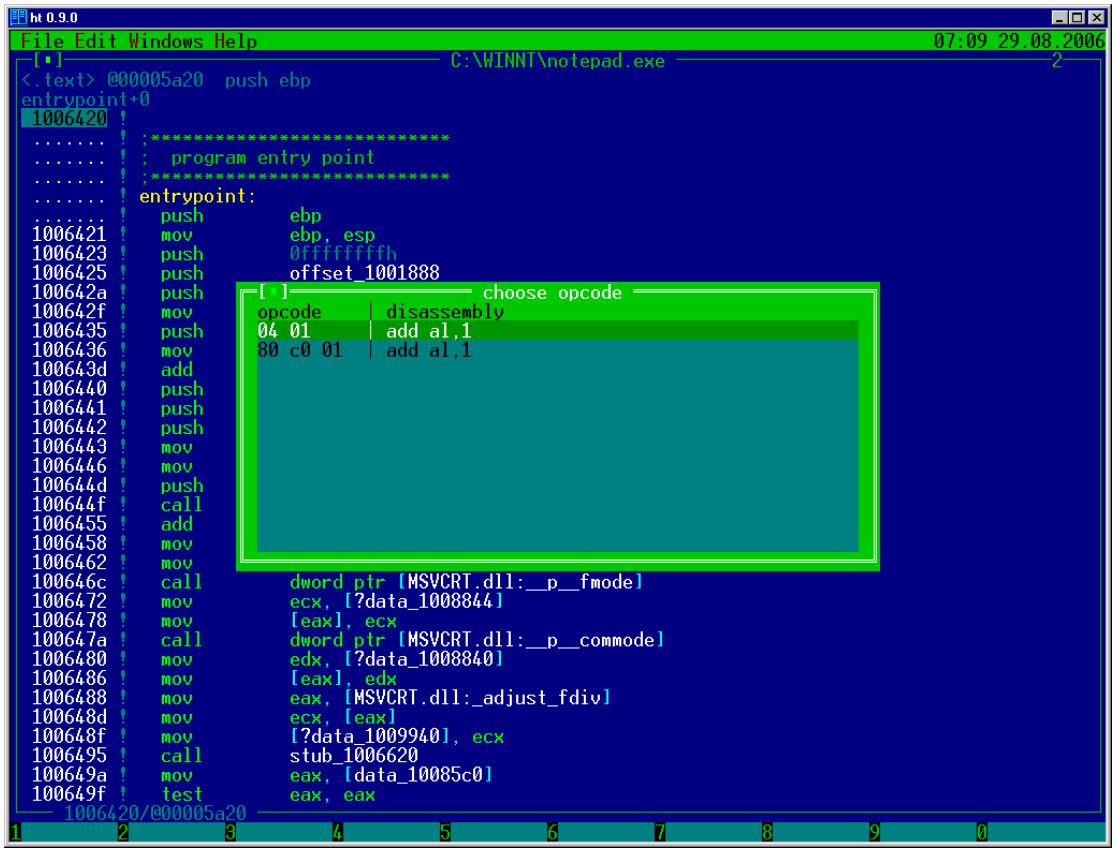

Рисунок 18 hex-редактор HTE — достойная замена hiew'y

Западанные хакеры во всю прутся от коммерческих **WinHex'**а (www.winhex.com/winhex/index-m.html) и **Hex Workshop**'а (www.bpsoft.com). Чего они в них нашли — непонятно.

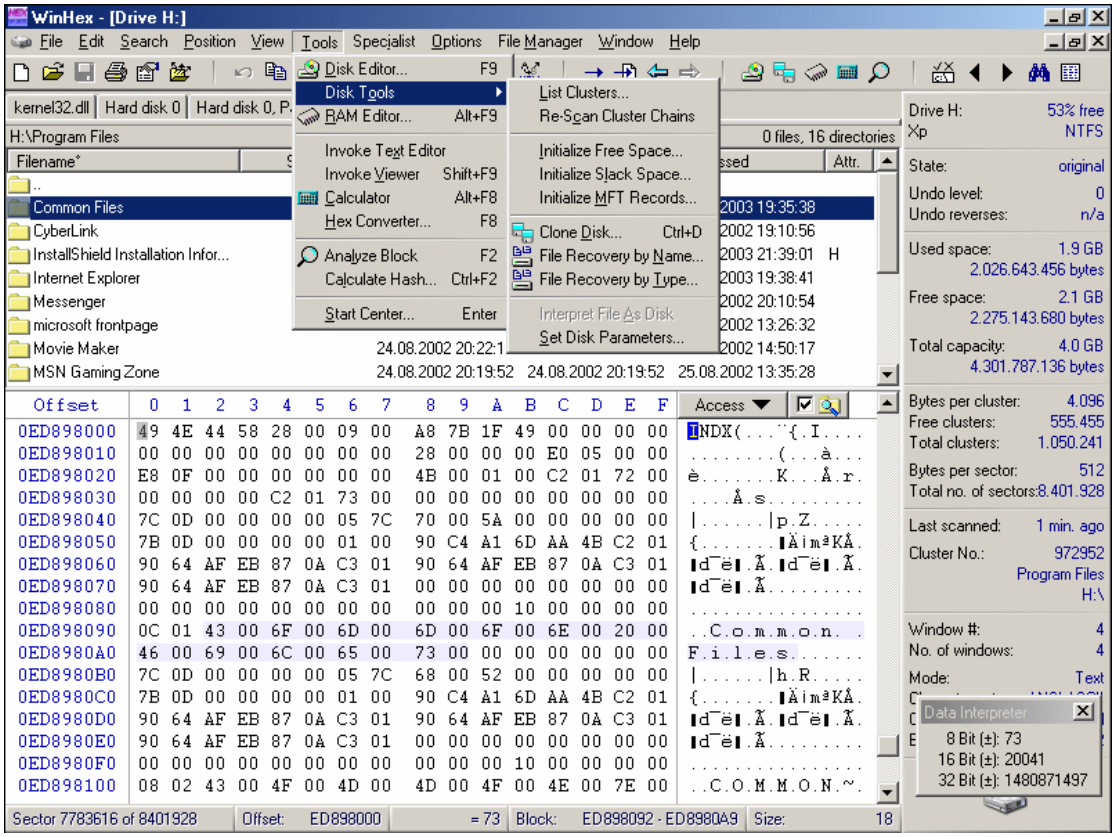

Рисунок 19 коммерческий hex-редактор WinHex популярный на западе

Ни ассемблера, ни дизассемблера в нем нет и навряд ли появятся в дальнейшем, зато есть калькулятор контрольных и хэш сумм (типа CRC16, CRC32, MD5, SHA-1), что в некоторых случаях оказывается очень удобным.

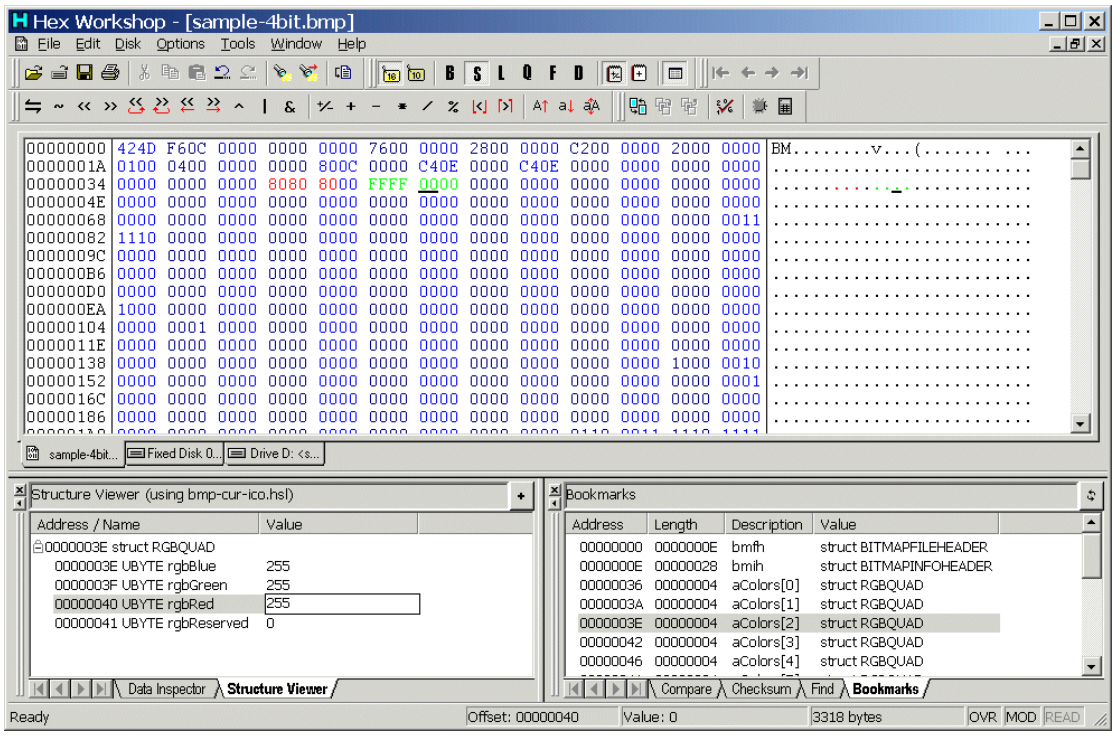

Рисунок 20 коммерческий hex-редактор Hex Workshop популярный на западе

распаковщики

Все больше и больше программ распространяются в упакованном виде (или защищаются протекторами, что еще хуже), в результате чего их непосредственное дизассемблирование становится невозможным, а, поскольку, многие упаковщики/проекторы содержат анти-отладочные приемы, то страдает и отладка.

Попытки создать универсальный распаковщик многократно предпринимались еще со времен MS-DOS и всякий раз проваливались, поскольку разработчики защит придумывали новую гадость. Тем не менее в состав большинства хакерских инструментов (IDA Pro, OllyDbg) входят генетические распаковщики, справляющие с несложными защитами. Сложные же приходится распаковывать руками (тому, как это сделать посвящено множество статей, которые легко найти в сети). Когда же один и тот же упаковщик встречается хакеру десятый раз кряду, он матерится и пишет автоматический/полуавтоматический распаковщик, чтобы облегчить себе работу. Коллекции таких распаковщиков собраны на www.exetools.com/unpackers.htm, programmerstools.org/taxonomy/term/16, www.woodmann.com/crackz/Packers.htm и других сайтах. Проблема в том, что каждый такой распаковщик рассчитан на строго определенную версию упаковщика/протектора и с другими работать просто не может! Чем чаще обновляется упаковщик/протектор, тем сложнее найти подходящий распаковщик, поэтому, лучше полагаться только на самого себя, распаковывая программу руками (и мы уже писали как).

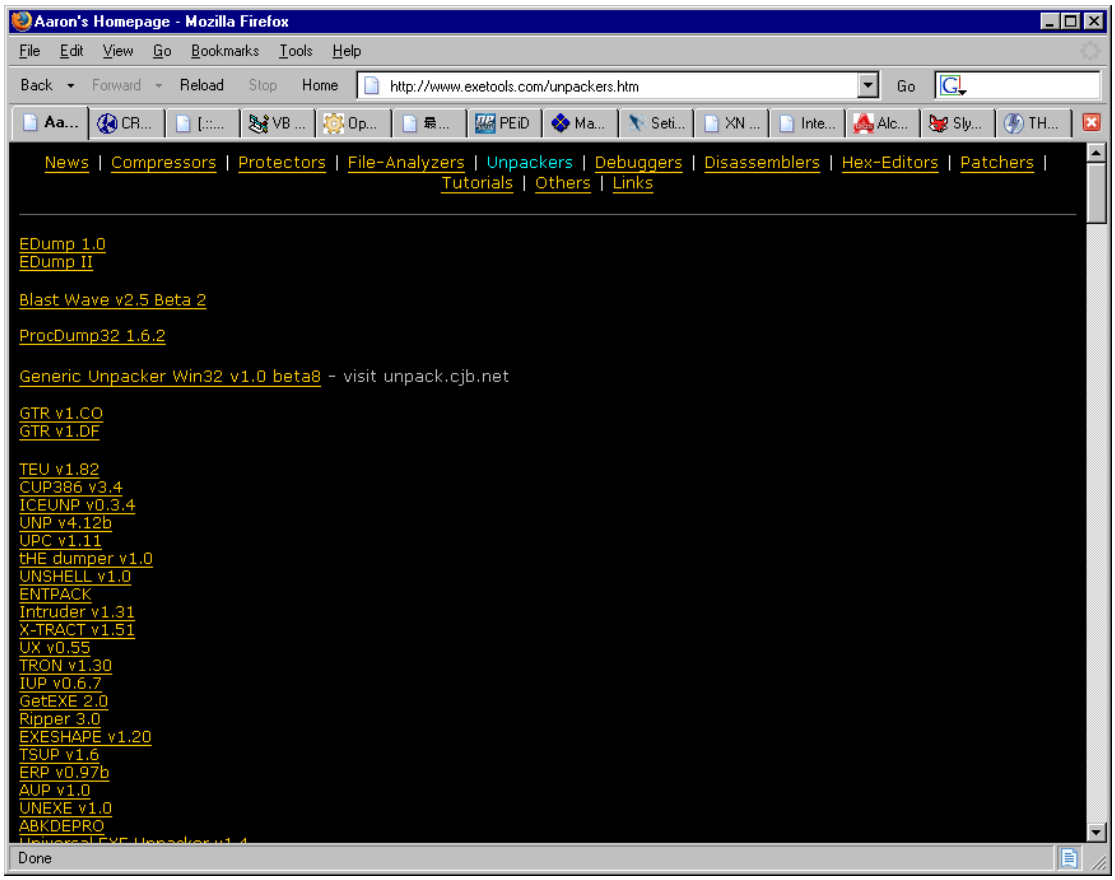

Рисунок 21 большая коллекция распаковщиков на exetools

Кстати, прежде чем искать распаковщик, неплохо бы для начала выяснить: чем же вообще защищена ломаемая программа? В этом поможет бесплатная утилита **PEiD** (peid.has.it), содержащая огромную базу сигнатур, хотя довольно часто ошибающаяся или дающая расплывчатый результат, но, тем не менее, это все-таки лучше, чем совсем ничего.

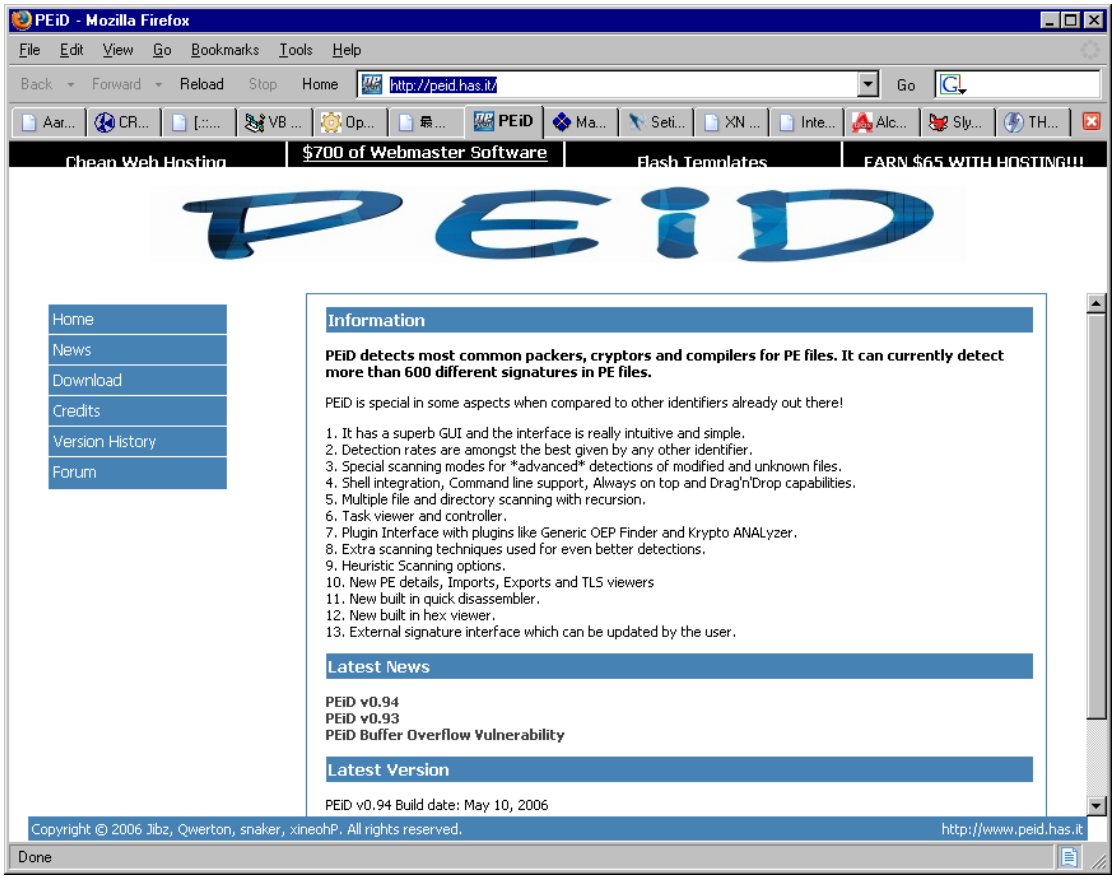

Рисунок 22 официальный сайт утилиты PEiD

дамперы

Снятие дампа с работающей программы — универсальный способ распаковки, убивающий практически все упаковщики и большую часть протекторов, правда, над полученным дампом еще предстоит как следует поработать и мыщъх рекомендует использовать дамп лишь для дизассемблирования. Сдампленная программа может работать неустойчиво, периодически падая в самый ответственный момент. Но это ладно, это все лирика. Забьем на лирику и обратимся к практике.

Самым первым (и самым неумелым) был **ProcDump**, затем появился **Lord PE**, учитывающий горький опыт своего предшественника и способный сохранять дамп даже в тех случаях, когда PE-заголовок умышленно искажен защитной, а доступ к некоторым страницам памяти отсутствует (атрибут PAGE_NOACCESS). Венцом эволюции стал **PE-TOOLS**, базовый комплект поставки которого можно найти практически на любом хакерском сервере, например, на WASM'e (www.wasm.ru/baixado.php?mode=tool&id=124) или на CrackLab'e (www.cracklab.ru/download.php?action=get&n=MTU1), а свежие обновления лежат на "родном" сайта проекта neox.iatp.by, кстати говоря, уже несколько раз поменявшим свой адрес (по непонятым причинам, базовый пакет на нем отсутствует).

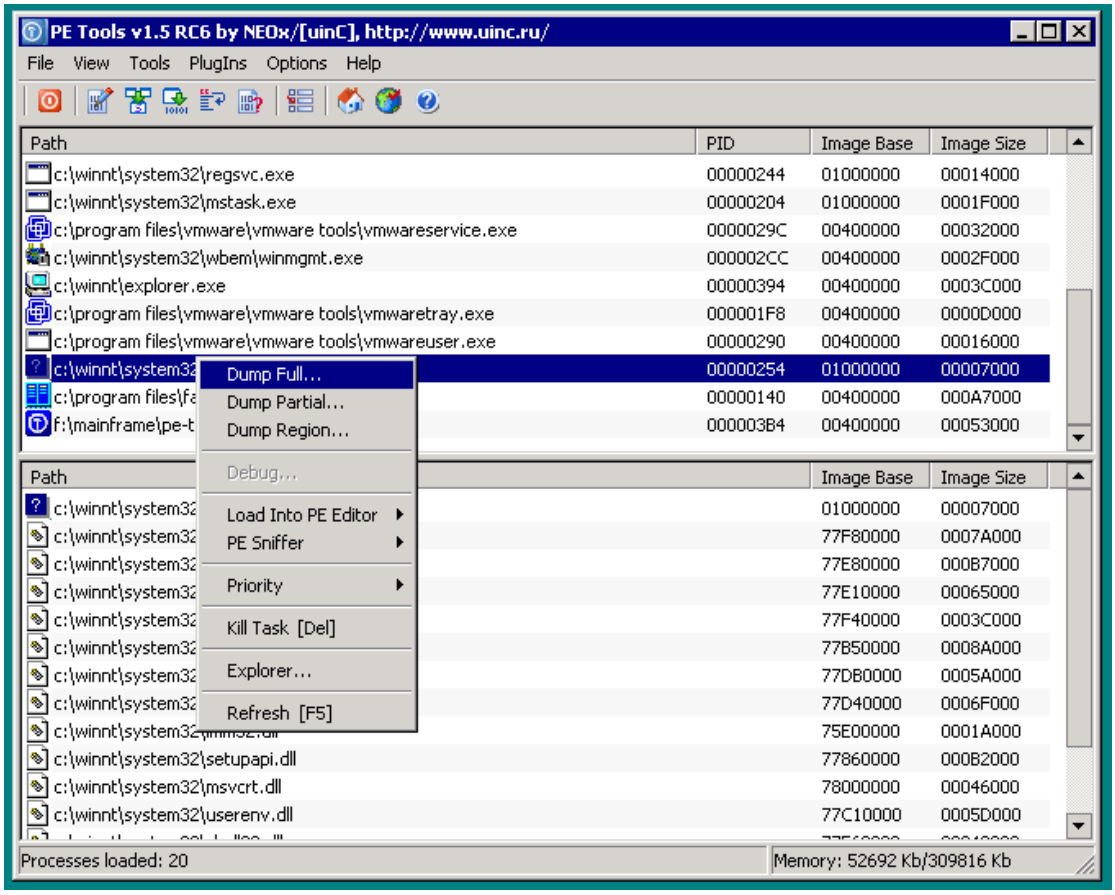

Рисунок 23 PE-TOOLS – один из лучший дамперов PE-файлов

После снятия дампа необходимо как минимум восстановить таблицу импорта, а иногда еще и таблицу перемещаемых элементов вместе с секцией ресурсов. Импорт лучше всего восстанавливать знаменитым **Import REConstructor**'ом, который вместе с **ReloX**'ом (восстанавливающим таблицу перемещаемых элементов) и минимально работающим генетическим распаковщиком можно найти там же, где и PE-TOOLS: wave.prohosting.com/mackt/main.htm. А вот здесь лежит коллекция программ для восстановления таблицы ресурсов www.wasm.ru/baixado.php?mode=tool&id=156, если же ни одна из них не справится со своей задачей, то, быть может, поможет бесплатный **Resource Binder**: www.setisoft.com/ru/redirect.php?dlid=89

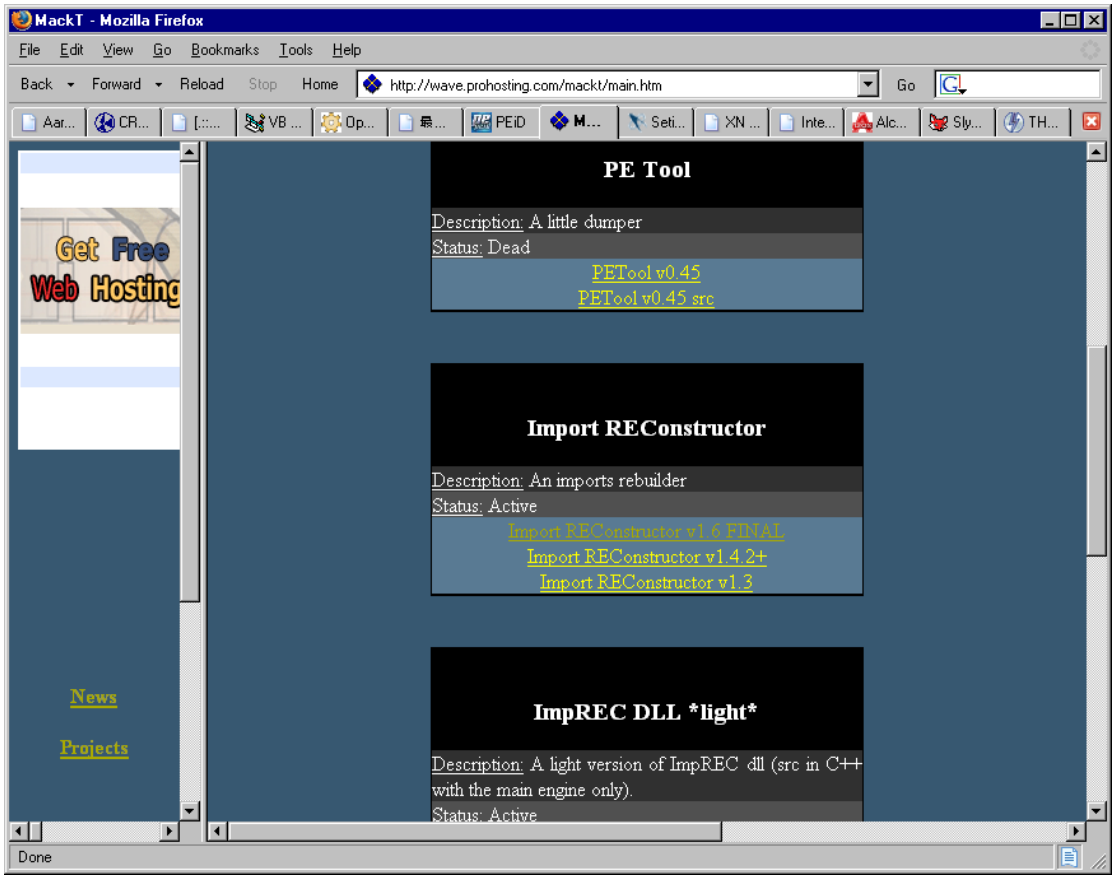

Рисунок 24 на этом сайте можно найти Import REConstructor и другие программы

редакторы ресурсов

Редактировать ресурсы приходится во многих случаях. Например, чтобы сменить текст диалогового окна, разблокировать элемент управления, перебить логотип и т. д. и т. п. Формально, редактор ресурсов входит в каждый Windows-компилятор, в том числе и в Microsoft Visual Studio, вот только после редактирования ресурсов файл зачастую становится неработоспособным! Это потому что штатный редактор ресурсов к таким задачам неприспособлен!

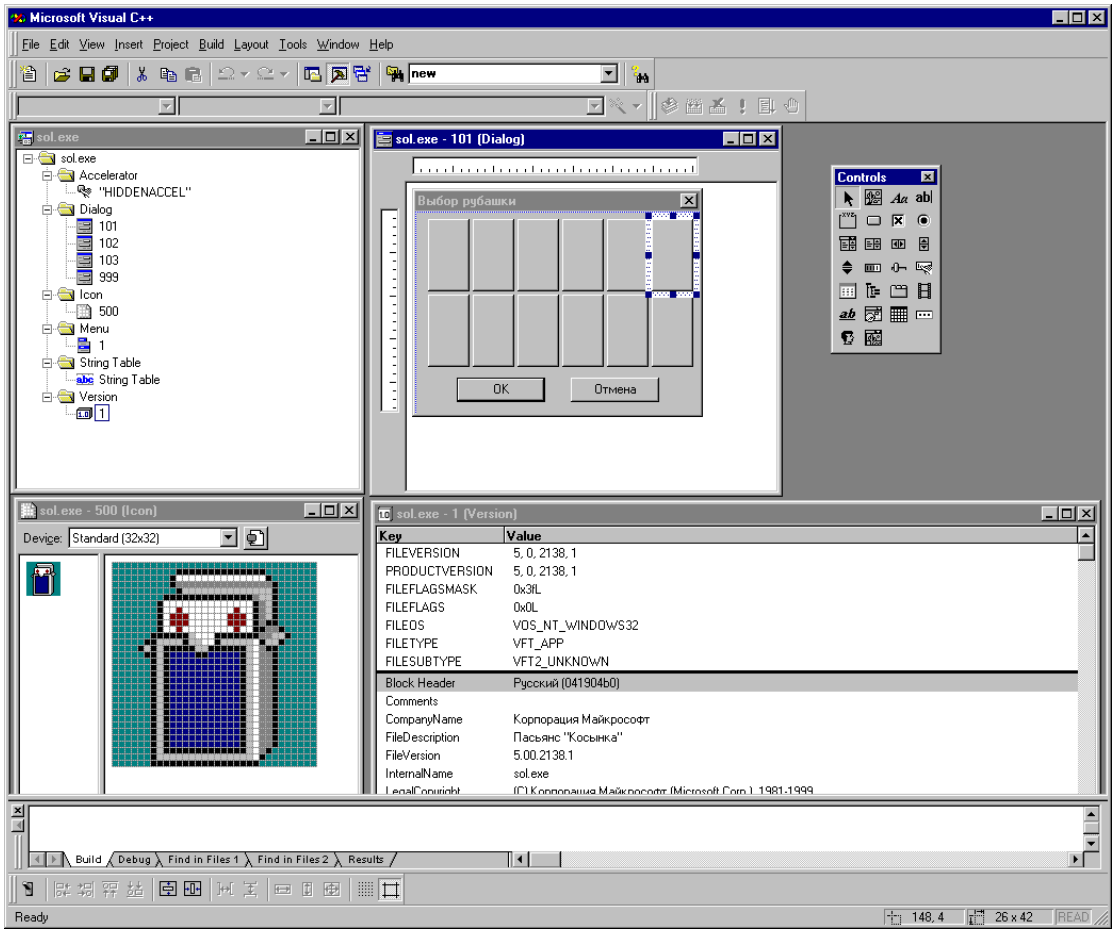

Рисунок 25 редактирование ресурсов в Microsoft Visual Studio

Лучим хакерским редактором был и остается коммерческий **Restorator Resource Editor** (www.bome.com/Restorator), который может практически все, что нужно и даже чуточку больше. Из бесплатных утилит в первую очередь хотелось бы отметить **XN Resource Editor** (www.wilsonc.demon.co.uk/d10resourceeditor.htm), написанный на совсем не хакерском языке DELPHI и распространяющегося в исходных текстах, что позволяет наращивать функционал программы, затачивая ее под свои собственные нужды (если вы знаете DELPHI, конечно).

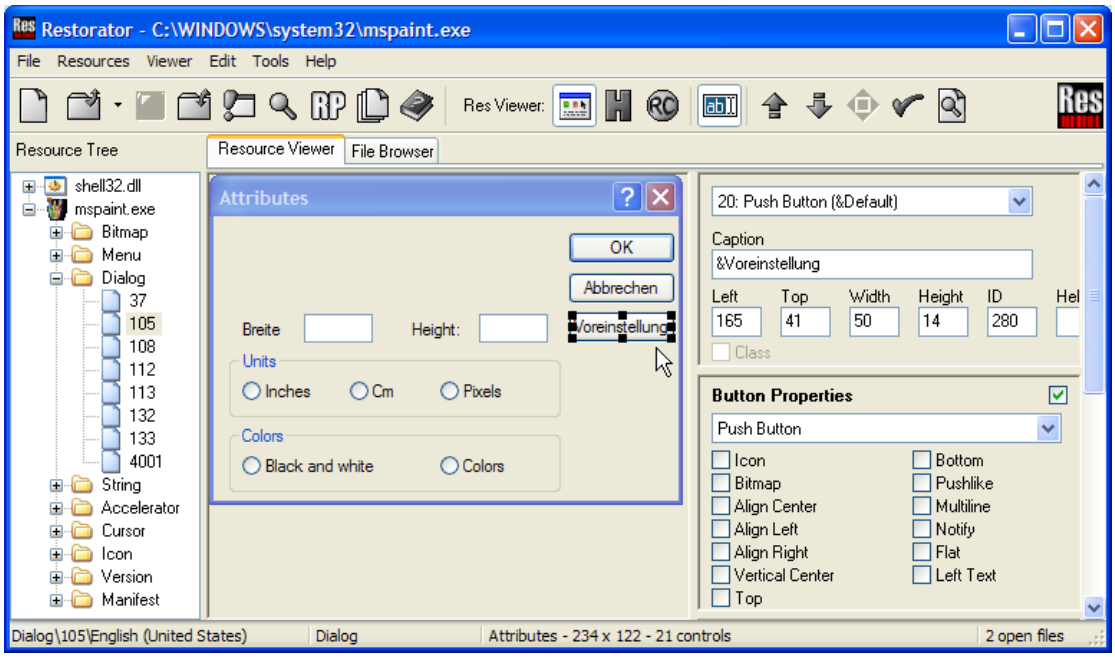

Рисунок 26 редактирование ресурсов в Restorator Resource Editor

ШПИОНЫ

В основном используется два типа шпионов — шпионы Windows-сообщений и API-шпионы. Первые следят за посылкой сообщений окнам и элементам управления, вторые — за вызовом API-функций, включая функции, экспортируемые динамическими библиотеками, поставляемыми вместе с программой. Шпионаж — лучшее (и наиболее дешевое — в смысле усилий и времени) средство, позволяющее узнать чем "дышит" защищенная программа.

Вполне достойный шпион сообщений входит в штатную поставку Microsoft Visual Studio и называется **Spyxx.exe**. Аналогичный по возможностям шпион, но только с открытыми исходными текстами лежит на www.catch22.net/software/winspy.asp и совершенно бесплатен.

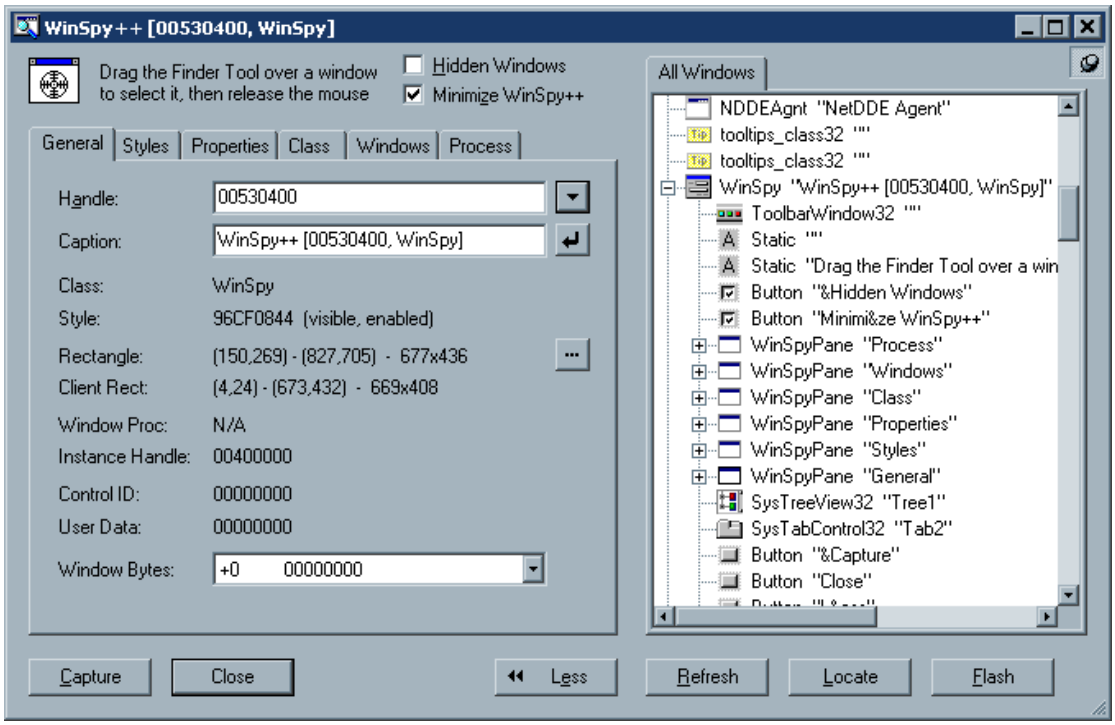

Рисунок 27 шпионаж за Windows-сообщениями при помощи бесплатной утилиты WinSpy

Из API-шпионов, лучшим на мой мыщъх'ый взгляд является **Kerberos** от Рустема Фасихова (www.wasm.ru/baixado.php?mode=tool&id=313), который взялся за клавиатуру тогда, когда остальные шпионы перестали его устаивать. Тем не менее, о вкусах не спорят и многие пользуются **APISpy32** (таким же бесплатным как и Kerberos) который можно раздобыть на www.internals.com. Впрочем, любой нормальный отладчик (как-то soft-ice, OllyDbg) можно настроить так, чтобы он выполнял функции API-шпиона, причем действуя по очень избирательному шаблону, избавляющему нас от просмотра многокилометровых листингов, генерируемых Kerberos'ом и APISpy32.

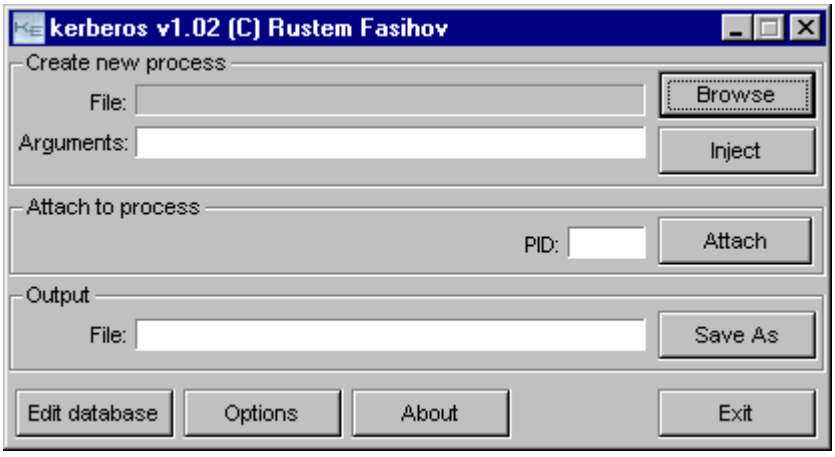

Рисунок 28 внешний вид API-шпиона Kerberos от Рустема Фасихова

мониторы

Чтобы узнать к каким файлам или ветвям реестра обращается подопытная программа достаточно воспользоваться файловым монитором и монитором реестра соответственно.

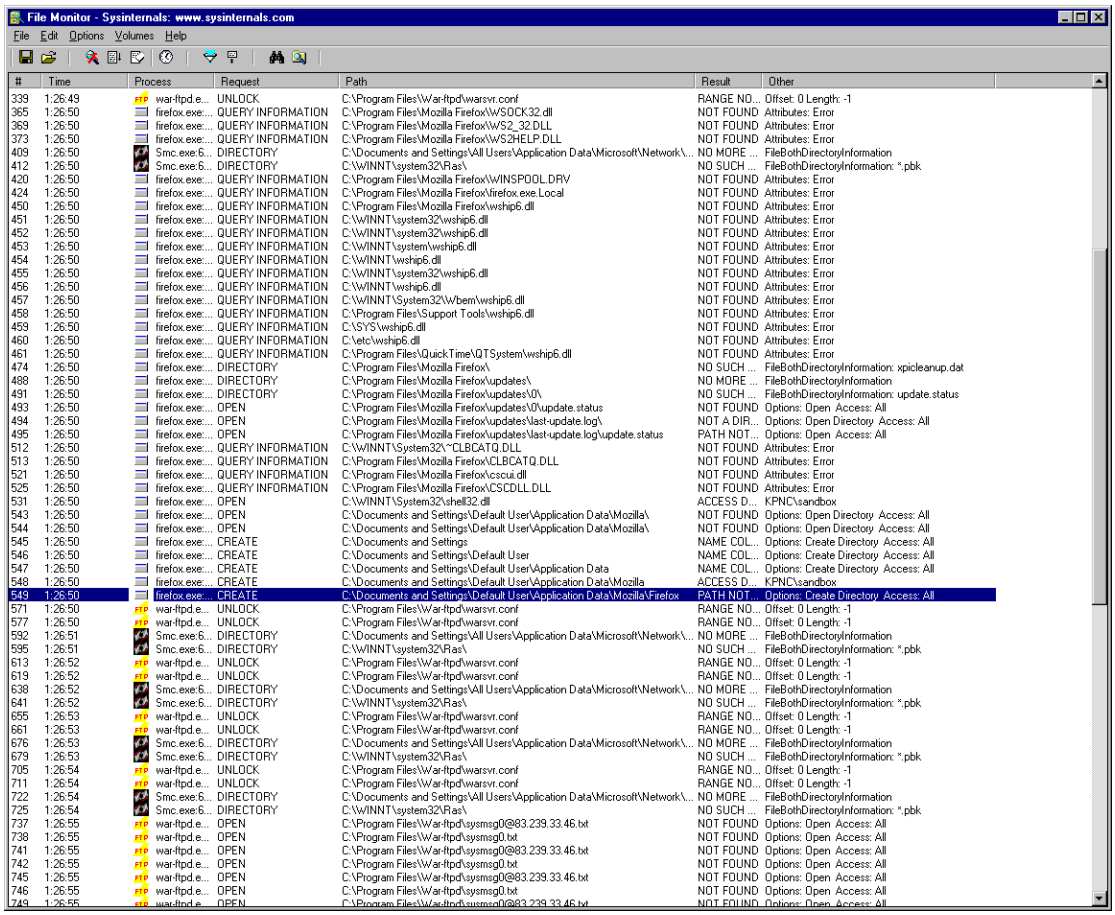

Рисунок 29 файловый монитор Марка Руссиновича за работой

Оба они были написаны легендарным исследователем недр Windows Марком Руссиновичем и долгое время распространялись совершенно бесплатно через некоммерческий сайт www.sysinternals.com, однако, в июне 2006 года Руссинович продался Microsoft и хотя его утилиты обещают остаться бесплатными и впредь, скорее всего они будет бесплатными *только для легальных пользователей Windows*, так что спешите качать пока дают на шару.

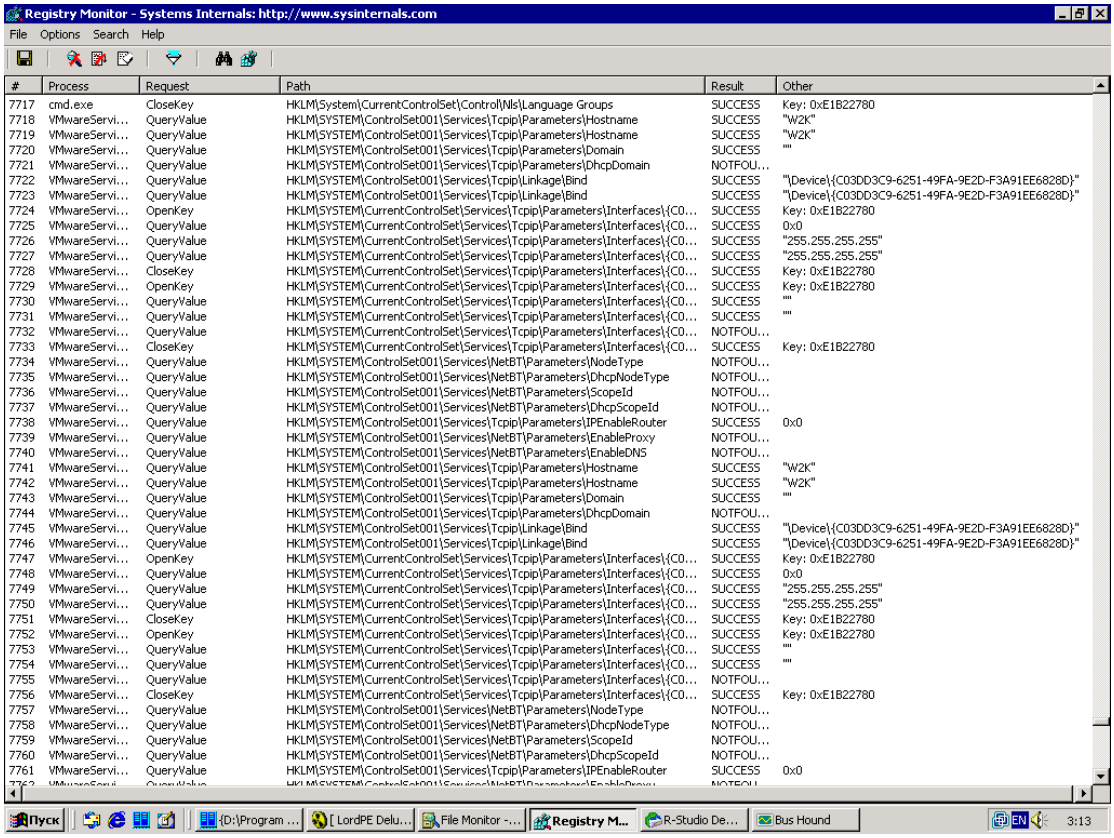

Рисунок 30 монитор реестра Марка Руссиновича

модификаторы

Существует два диаметрально противоположных полхода к взлому программ. Самое трудное (но самое идеологически правильное и наименее всего наказуемое) создание своих собственных генераторов серийных номеров, ключевых файлов и т. д. Проанализировав, как работает оригинальный генератор, хакер пишет точно такой же и раздает его всем кому надо (и кому не надо — тоже). Однако, это слишком утомительно, тем более что большинство защит нейтрализуются правкой нескольких байт. Вот только распространять взломанный файл нельзя. За это могут и по лапкам дать. К тому же, как правило, exe/dll слишком тяжелы для распространения, поэтому, возникает естественная идея — распространять не сам взломанный файл, а список байт с адресами, которые надо исправить. Понятное дело, никакой юзер с hiew'ом внутрь программы не полезет, поэтому на помощь приходит автоматизация.

Получить список различий между оригинальным и взломанным файлом поможет утилита **fc.exe**, входящая в штатный комплект поставки Windows, а вот, чтобы внести исправления в exe/dll, понадобиться утилита-модификатор, которую можно написать буквально за несколько минут, а если писать лень, то вот коллекция уже готовых: www.wasm.ru/baixado.php?mode=tool&id=35.

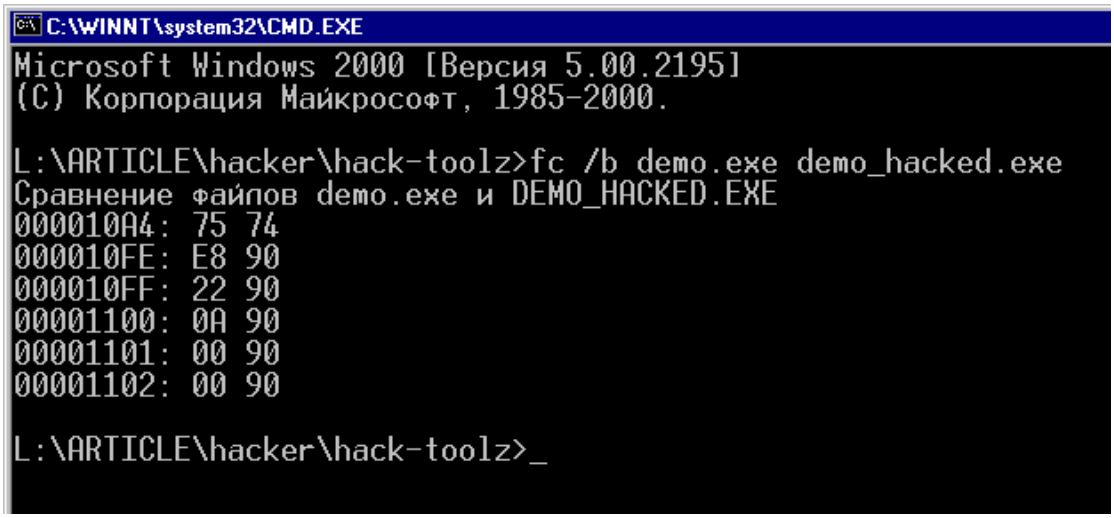

Рисунок 31 поиск различий между оригинальной и хакнутой версией файла с помощью штатной утилиты FC.EXE

Хуже есть программа упакована/защищена протектором. Тогда ее приходится править уже налету, непосредственно в оперативной памяти, для чего пригодится **Process Patcher** (www.wasm.ru/baixado.php?mode=tool&id=38), **R!SC's Process Patcher** (www.wasm.ru/baixado.php?mode=tool&id=39) или ***ABEL* Self Learning Loader Generator** (www.wasm.ru/baixado.php?mode=tool&id=144). Последняя программа отличается тем, что ищет исправляемые байты не по фиксированным смещениям, а по регулярным шаблонам, что позволяет ей в большинстве случаев переживать выход новой, слегка измененной версии ломаемой программы (если, конечно, изменения затронули не защитный механизм, а что-нибудь другое).

копировщики защищенных дисков

Копировать защищенные диски — это совсем не по-хакерски, а гораздо более по пиратски, тем не менее, заниматься этой деятельностью всем приходится, так что лишними они не будут.

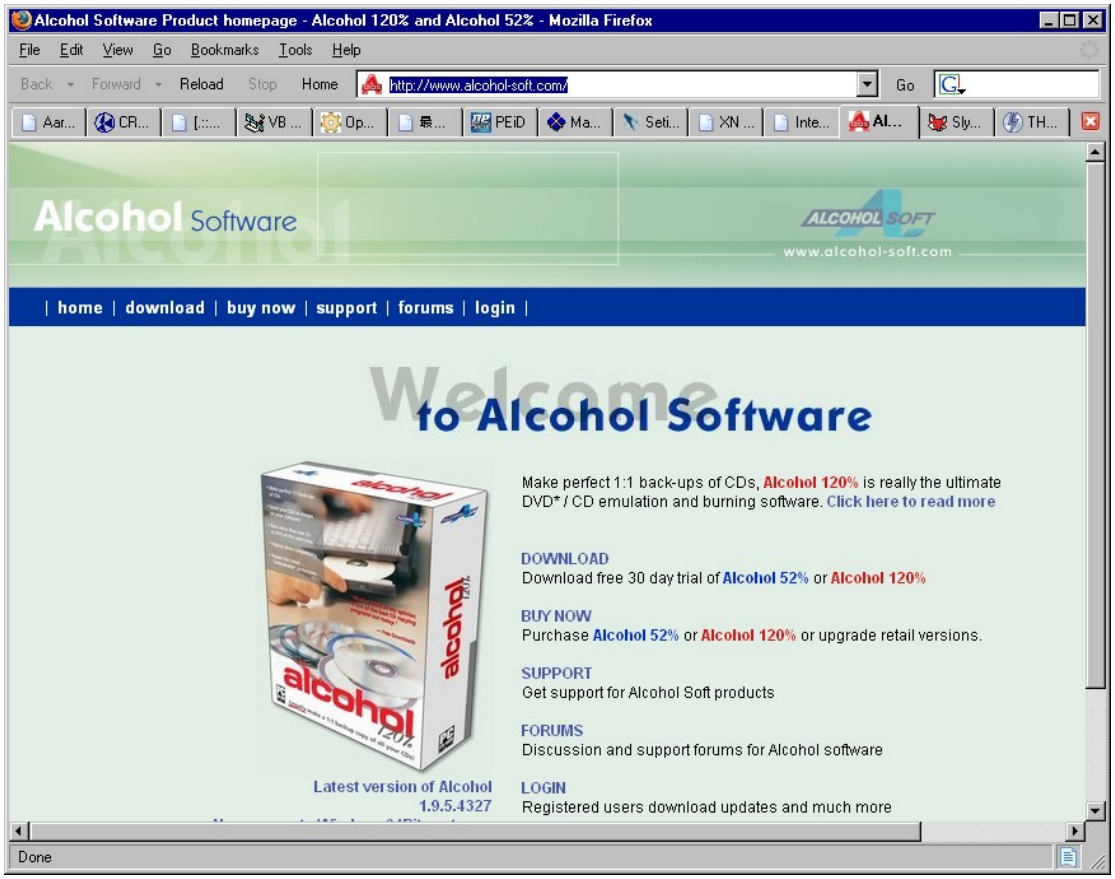

Рисунок 32 официальный сайт копировщика защищенных дисков Alcohol 120%

Пара лучших коммерческих копировщиков это, бесспорно, Alcohol 120% (www.alcohol-soft.com) и CloneCD (www.slysoft.com/en/clonecd.html). Бесплатная утилита Daemon Tools (www.daemon-tools.cc) позволяет монтировать образы, снятые двумя этими копировщиками как виртуальные диски, образ которых лежит на HDD. Очень удобно!

Рисунок 33 официальный сайт копировщика защищенных дисков CloneCD