УДК 681.3.06:519.248.681

Г.Н. ГУЛАК, Е.Г.КАЧКО, канд. техн. наук., С.С. ЛАВРИНЕНКО

ЗАЩИЩЕННОСТЬ ОПЕРАЦИОННЫХ СИСТЕМ: ТРЕБОВАНИЯ, ОСНОВНЫЕ АТАКИ, СРАВНИТЕЛЬНЫЙ АНАЛИЗ

Введение

По данным годового отчета «2001 Computer Crime and Security Survey» [1] Института компьютерной безопасности в Сан-Франциско и ФБР, финансовые потери от компьютерных преступлений в США за минувший год выросли на 43% с 265,6 млн. долл. до 377,8 млн. При этом 85% респондентов из 538, в основном из промышленных и государственных структур, заявили о фактах нарушения компьютерной безопасности, причем не только из-за атак злоумышленников. Почти 64% были озабочены понесенными убытками, но лишь 35% смогли оценить их в денежном выражении. Около 70% респондентов заявили, что чаще всего атакам подвергались Internet-каналы, а 31% показали, что атакам подвергались внутрикорпоративные системы. Случаи вторжения извне подтверждали 40% респондентов (в 2000 г. — 25%), а 38% фиксировали отказ в обслуживании (27% в 2000 г.). На нарушение привилегий из-за злоупотребления сотрудниками работой в Сети жаловались 91% респондентов, а 94% обнаружили в своих системах вирусы (в 2000 г. это отмечали 85%).

Даже из этих скупых цифр видна явно негативная тенденция — Internet не только возводит мосты между странами и континентами, но и приближает преступника к жертве. Если оставить в стороне извечные вопросы разведки и промышленного шпионажа и сосредоточиться только на «бытовой» стороне дела, то одними из ведущих проблем в области информационной безопасности в минувшем году стали атаки на платежные системы, дискредитация компаний (отказ в обслуживании), производственный саботаж, вскрытие корпоративных секретов, нарушение прав интеллектуальной собственности. По оценкам отдела по науке и технологиям при президенте США, ежегодный урон, наносимый американскому бизнесу компьютерными злоумышленниками в последние годы, достигал 100 млрд. долл. Потери от несанкционированного доступа к информации, связанной с деятельностью финансовых институтов США, составляли не менее 1 млрд. долл. в год. Таким образом, американский бизнес вплотную подошел к тому рубежу, когда своевременное и адекватное решение вопросов безопасности для него становится экономически целесообразным.

Еще более критическая ситуация в части уровня защищенности ОС возникла в большинстве стран мира, в том числе в Украине. Дело в том, что поставляемые в Украину ОС могут содержать ненадежное системное ПО, выполнять незадекларированные функции, иметь ограничение в использовании, например использовании механизмов и средств защиты. Выходом из этой ситуации мог бы быть подробный анализ исходных кодов ОС, однако они недоступны потребителю по ряду причин.

Целью настоящей статьи является анализ основных угроз безопасности ОС, классификация и анализ основных атак на ОС, а также определение основных методов защиты от них. При изложении будем ориентироваться на методологию, изложенную в ISO-15408 [2].

1. Требования к защищенной ОС

1.1. Классификация угроз безопасности операционной системы

Применительно к ОС Windows NT, Windows 2000 и Linux угрозы можно классифицировать по цели реализации, принципам воздействия на ОС, характеру воздействия на ОС, типу используемой слабости, по способам воздействия на объект атаки, по способу действия злоумышленника, по объекту атаки, используемым средствам атаки, по состоянию атакуемого объекта ОС на момент атаки.

По цели реализации угрозы:

- несанкционированное чтение информации;

- несанкционированное изменение информации;
- несанкционированное уничтожение информации;
- полное или частичное разрушение операционной системы. Под разрушением операционной системы понимается целый комплекс разрушающих воздействий от кратковременного вывода из строя отдельных программных модулей системы до физического стирания с диска системных файлов;
- несанкционированное использование ресурсов ОС;
- несанкционированная модификация отдельных программных модулей, библиотек и др.

По принципу воздействия на операционную систему:

- используя известные (легальные) каналы получения информации. К этому классу относится, например, угроза несанкционированного чтения файла, доступ пользователей к которому, согласно адекватной политике безопасности, должен быть запрещен;
- используя скрытые каналы получения информации. Например, угроза использования злоумышленником недокументированных возможностей операционной системы;
- создавая новые каналы получения информации с помощью программных закладок.

По характеру воздействия на операционную систему:

- активное воздействие - несанкционированные действия злоумышленника в системе, приводящее к изменению состояния ОС;
- пассивное воздействие - несанкционированное наблюдение злоумышленника за процессами, происходящими в системе;
- адаптивное состояние – когда злоумышленник может, в зависимости от состояния ОС, воздействовать то активно то пассивно.

По типу используемой слабости защиты:

- используя неадекватную политику безопасности, в том числе и ошибки администратора системы;
- используя ошибки и недокументированные возможности программного обеспечения операционной системы, в том числе и так называемые люки - случайно или преднамеренно встроенные в систему "служебные входы", позволяющие обходить систему защиты, используя ранее внедренную программную закладку.

По способу воздействия на объект атаки:

- непосредственное воздействие, в том числе дистанционное;
- превышение пользователем своих полномочий;
- работа от имени другого пользователя;
- использование результатов работы другого пользователя (например, несанкционированный перехват информационных пакетов, инициированных другим пользователем).

По способу действий злоумышленника:

- в интерактивном режиме (например, вручную);
- в пакетном режиме (с помощью специально написанной программы, которая выполняет негативные воздействия на операционную систему без непосредственного участия пользователя-злоумышленника).

По объекту атаки:

- операционная система в целом;
- объекты операционной системы (файлы, устройства и т. д.);
- библиотеки, драйверы, протоколы взаимодействия;
- субъекты операционной системы (пользователи, системные процессы и т. д.);

- информационно-телекоммуникационные каналы.

По используемым средствам атаки:

- штатными средствами операционной системы без использования дополнительного программного обеспечения;
- программным обеспечением третьих фирм. К этому классу программного обеспечения относятся как компьютерные вирусы и другие вредоносные программы (exploits), которые можно легко найти в Internet, так и программное обеспечение, изначально разработанное для других целей (отладчики, сетевые мониторы и сканеры и т. д.);
- специально разработанным программным обеспечением;
- средствами, заложенными в систему при ее проектировании и изготовлении.

По состоянию атакуемого объекта операционной системы на момент атаки:

- хранение информации;
- обработка информации;
- передача информации;
- решение задач;
- активация критической информации, например, ввод ключей или паролей и др.

1.2. Понятие защищенной операционной системы

Основные определения

Операционная система является защищенной, если она предусматривает средства защиты от основных классов угроз, описанных выше. Защищенная операционная система обязательно должна содержать средства разграничения доступа пользователей к своим ресурсам, а также средства проверки подлинности пользователя, начинающего работу с операционной системой. Кроме того, защищенная операционная система должна содержать средства противодействия случайному или преднамеренному выводу операционной системы из строя.
Если операционная система предусматривает защиту не от всех основных классов угроз, а только от некоторых, то такая операционная система называется частично защищенной. Например, операционная система MS-DOS с установленным антивирусным пакетом является частично защищенной системой - она защищена от компьютерных вирусов.

Политикой безопасности называется набор норм, правил и практических приемов, регулирующих порядок хранения, обработки и использования ценной информации. В отношении операционной системы политика безопасности определяет то, какие пользователи могут работать с операционной системой, какие пользователи имеют доступ к каким объектам операционной системы, какие события должны регистрироваться в системных журналах и т. д.

Адекватной политикой безопасности называется такая политика безопасности, которая обеспечивает достаточный уровень защищенности операционной системы. Следует особо отметить, что адекватная политика безопасности - это не обязательно та политика безопасности, при которой достигается максимально возможная защищенность системы. По сути это политика, которая минимизирует потери в ОС до допустимых значений.

1.3. Комплексная система защиты

Существуют два основных подхода к созданию защищенных операционных систем - фрагментарный и комплексный. При фрагментарном подходе вначале организуется защита от одной угрозы, затем от другой и т. д. Примером фрагментарного подхода может служить ситуация, когда за основу берется незащищенная операционная система (например, Windows 95), на нее устанавливаются антивирусный пакет, система шифрования, система регистрации действий пользователей и т. д.

Основной недостаток фрагментарного подхода очевиден - при применении этого подхода подсистема защиты операционной системы представляет собой набор разрозненных про-

граммных продуктов, как правило, от разных производителей. Эти программные средства работают независимо друг от друга, организовать их тесное взаимодействие практически невозможно. Кроме того, отдельные элементы такой подсистемы защиты могут некорректно работать в присутствии друг друга, что приводит к резкому снижению надежности системы. Поскольку подсистема защиты, созданная на основе фрагментарного подхода, не является неотъемлемой компонентой операционной системы, при отключении отдельных защитных функций в результате несанкционированных действий пользователя-злоумышленника остальные элементы операционной системы продолжают нормально работать, что еще более снижает надежность защиты.

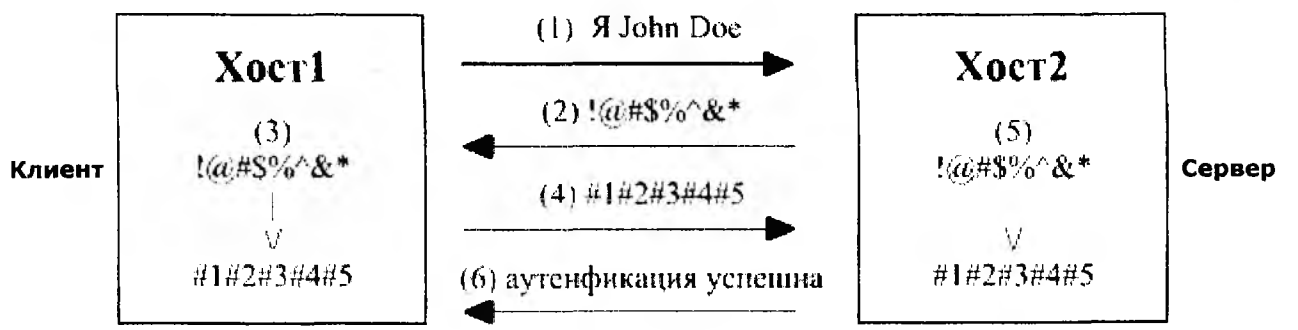
При комплексном подходе защитные функции внедряются в операционную систему на этапе проектирования архитектуры операционной системы и являются ее неотъемлемой частью. Комплексный подход предусматривает применение организационных и физических мер, организационно-технических мер и мероприятий, юридических и законодательных норм и др. Отдельные элементы подсистемы защиты, созданной на основе комплексного подхода, тесно взаимодействуют друг с другом при решении различных задач, связанных с организацией защиты информации. Поскольку вся подсистема защиты разрабатывается и тестируется в совокупности, конфликты между ее отдельными компонентами практически невозможны. Подсистема защиты, созданная на основе комплексного подхода, может быть устроена так, что при фатальных сбоях в функционировании ее ключевых элементов она вызывает крах операционной системы, что не позволяет злоумышленнику отключать защитные функции системы. При использовании фрагментарного подхода такая организация подсистемы защиты невозможна.

Как правило, подсистему защиты операционной системы, созданную на основе комплексного подхода, проектируют так, что отдельные ее элементы являются заменяемыми и соответствующие программные модули могут быть заменены другими модулями, реализующими предусмотренный интерфейс взаимодействия соответствующего программного модуля с другими элементами подсистемы защиты.

2. Основные атаки

2.1. Локальные атаки

Рассмотрим, 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 помощью хэш-функции генерирует хэш (от английского "hash" - мешанина), где входными данными являются пароль и полученные данные.
4. Передача полученного хэша серверу.

5. Сервер генерирует на своей стороне хеш, используя те же входные данные (пароль и случайные данные).
6. Сверив два хэша, сообщается результат аутенфикации.

Хэш-функция необратима, т.е. нельзя получить пароль, имея только хэш (не перебирая все варианты). Как видно - при данной схеме избегается передача пароля в незашифрованном виде. В Windows NT, Windows2000 пароли, а вернее хэш-значения паролей, для локального и удаленного входа в систему хранятся в файле *%systemroot%\system32\config\SAM*. Однако просмотреть этот файл, даже имея права администратора, не удастся - система блокирует обращения к этому файлу. В файле *SAM* (Security Account Manager) хранятся хеш-значения 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 хэш, 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 хэш 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 хеш-значения.

Алгоритм формирования LanMan-hash:

1. Веденный пароль переводится в верхний регистр.
2. Затем константная строка шифруется алгоритмом DES, используя в качестве ключа 7 первых байтов 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.

Основными целями атак являются [3]:

- получение прав привилегированного пользователя или администратора;
- нарушение доступности сервера;
- нарушение функциональности сервера.

Анализ показал, что существует несколько методов локального получения прав привилегированного пользователя или администратора. Рассмотрим основные из них:

- подмена пароля - этот метод базируется на подмене данных авторизации пользователя, которые хранятся в файле SAM (а именно в V-блоке). На основе известного пароля генерируются NT-hash и LM-hash и записываются в V-блок пользователя (встроенной учетной записи администратора). После этого можно будет спокойно локально войти в систему, используя логин встроенной учетной записи администратора и уже известный пароль;
- подбор пароля - имея на руках файл SAM, в котором хранится хэш-значение пароля, можно найти пароль, используя полный перебор;
- модификация исполняемого кода - очень эффективный метод, который состоит в том, чтобы обойти исполняемый код проверки пароля. Данная проверка осуществляется в библиотеке MSV1_0.DLL;
- подмена системных файлов - суть этого метода состоит в замене системного сервиса. Суть этого метода состоит в том, чтобы найти сервис (service), запускающий от имени system и расположенный в каталоге, куда обычный пользователь имеет полный доступ, и заменить его на свой . Это может быть брандмауэр или какие-нибудь другие дополнительно установленные сервисы. Недостаток данного метода в том, что далеко не на всех системах можно будет найти нужные сервисы

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).

Если система, к которой необходимо получить пароли, находится в локальной сети, можно попробовать анализировать трафик, с целью перехвата хэш-значений паролей при авторизации в домен (*NT-hash* или *LanMan-hash*). Это можно осуществить как с помощью просто программ-анализаторов, так и программой *L0phtCrack* (*LC3*) - что намного удобнее. В ней имеется функции прослушивания сети на предмет передачи *LanMan-* и *NT*-хэшей. Если локальная сеть коммутирована свитчами, то хэш-значения паролей будет получить весьма сложно. 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 машины. После открытия письма почтовым клиентом будет запрошен файл с общих ресурсов. В этот момент можно будет "поймать" LanMan-хэш (т.к. будет проведена процедура аутенфикации). Для того чтобы "поймать" *LanMan*-хэш (при попытке подключения общего ресурса), можно воспользоваться утилитой *smbrelay*.

Как же защитить свою систему от локального/удаленного взлома?

Можно предложить несколько основных рекомендаций по защите системы на базе NT от атак, связанных с локальным доступом:

- Windows NT должна быть единственной операционной системой на вашем компьютере;
- у пользователя не должно быть возможности загрузиться с дискеты;
- системный блок должен находиться в недоступном для пользователей месте;
- в качестве файловой системы следует использовать только NTFS;
- число пользователей с правом локальной регистрации должно быть максимально ограничено (для этого в программе UserManager в меню Policies выберите пункт UsersRights, в списке Rights укажите log on locally и отредактируйте список разрешенных пользователей);
- при появлении новых версий сервисных пакетов и "заплат" на сервере http://www.microsoft.com их по возможности следует тут же устанавливать.
- Пароль 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.
- Используйте утилиту *syskey*. 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.
- Также никогда не оставляйте машину без присмотра незаблокированной.

2.2. Атака PipeBomb

Системные сервисы в Windows NT не ограничивают максимальное количество создаваемых экземпляров канала, а каждый канал, как правило, обрабатывается отдельным потоком (т. е. происходит классическое, популярное со времен UNIX, расщепление процесса-обработчика при запросе на очередное подключение). Все потоки и каждый экземпляр канала требуют некоторого количества оперативной памяти, и, если злоумышленник вздумает в бесконечном цикле устанавливать все новые и новые соединения, оперативной памяти может попросту не хватить!

При создании канала система размещает входящий и исходящий буферы в неоткачиваемой памяти (non-paged pool). Поэтому максимальное количество экземпляров канала определяется объемом неоткачиваемой памяти, выделенной процессу. Таким образом, существует возможность как заблокировать создание новых экземпляров канала, так и замедлить работу системы, отобрав у системных процессов всю свободную оперативную память и заставляя их за каждой страницей обращаться к диску. Такая атака получила название PipeBomb [4].

2.3. Атака AdminTrap

В программном интерфейсе Win32 существует функция ImpersonateNamedPipeClient, выполняющая олицетворение (impersonation) клиента канала. Олицетворение клиента канала заключается в том, что потоку, вызвавшему данную функцию, назначается маркер доступа (access token) клиента экземпляра канала, handle серверного конца которого указан в качестве параметра функции. При этом поток процесса-сервера, обслуживающий данный экземпляр канала, получает полномочия пользователя, который подключился к этому экземпляру канала в качестве клиента.

Если прикладная программа, выполняющаяся с правами обычного пользователя, создаст экземпляр канала, дождется подключения клиента и выполнит олицетворение, эта программа получит полномочия клиента, которые могут превышать изначальные полномочия программы. Таким образом можно получить права привилегированного клиента, в том числе и администратора. Эта идея была положена в основу атаки AdminTrap [5].

Таким образом, используя атаку PipeBomb, можно заблокировать создание системных каналов и, создав свой, дождавшись удаленного входа администратора, захватить его права.

Эта атака удаленного перехвата не может быть устранена даже при помощи правильного администрирования.

Как же защититься от этих атак?

Необходимо написать драйвер, который должен перехватывать запросы к драйверу npfs.sys, который ответственен за создание новых экземпляров каналов. После выполнения запроса драйвер должен с помощью вызова NtSetSecurityObject установить только что созданному экземпляру канала корректные атрибуты защиты.

Однако создание такого драйвера, перехватывающего те или иные системные вызовы, является весьма трудоемкой задачей. Кроме того, драйверы, перехватывающие системные вызовы kernel mode, как правило, жестко привязаны к конкретной версии ядра Windows NT. Установка нового сервис-пака чаще всего делает такой драйвер неработоспособным. А изготовлять для каждого сервис-пака свою версию драйвера явно нецелесообразно.

Таким образом, полная ликвидация вышеописанных слабостей NPFS ведет к тому, что должен быть полностью переписан драйвер npfs.sys, что является весьма непростой задачей, особенно если учесть необходимость обеспечения обратной совместимости со старыми спецификациями NPFS.

2.4. Атака LKM (Loadable kernel module)

Драйвер это фактически часть ядра ОС со всеми вытекающими отсюда последствиями. Возможностью динамически загружать драйверы в процессе работы системы обладают многие современные ОС (Solaris,FreeBSD,Linux,Win9x,WinNT,Win2000 и т.д.)

Таким образом, загрузив свой драйвер и перехватив, к примеру, операции файловой системы, мы сможем:

- перехватывать операции создания, открытия, чтения, записи в любой файл (у нас привилегии ядра системы);
- перехватывать операции сканирования каталогов;
- перенаправлять файловые операции (например, перенаправить вывод в log-файл).

Такая атака получила название LKM-атаки.

Основным отличием данного перехватчика от остальных является то, что он не существует для системы. То есть при загрузке драйвера он копирует себя в новое место, а системе говорит, что его загрузка завершилась неудачей.

2.5. Подмена MAC

Как может выглядеть атака с подменой MAC- и IP-адреса? Рассматривается случай с провайдером.

Первый этап выполняется, когда "жертва" в сети. С помощью программы-анализатора проверяется, чтобы "жертва" находилась в том же сегменте сети. Сегмент сети в данном случае область, внутри которой все машины равнозначны, т.е. получают одну и ту же информацию. Например, все машины, соединенные хабом (хабами), находятся в одном сегменте, так как хаб просто рассылает получаемую от одной машины информацию по всем остальным.

Критерием того, что ты находишься в одном сегменте сети с X, является то, что ты видишь пакеты, отправленные X, но не предназначенные тебе. Информация по сети передается в виде пакетов - кусков информации определенного размера.

Пакеты бывают двух типов: адресованные кому-то конкретно либо "всем". Последние выделяются тем, что у них MAC-адрес получателя равен FF-FF-FF-FF-FF-FF. Если вы получили пакет, в котором MAC-адрес получателя не равен вашему MAC-адресу, то можете быть уверены, что его отправитель находится в том же сегменте сети. Адрес можно узнать при помощи команды arp -a - она выводит список MAC-адресов, хранящихся в кэше компьютера. Если адреса в кэше не оказалось, стоит попробовать сделать ping на этот адрес и одновременно arp -a, если же его там так и не оказалось - скорее всего, жертва не в ваше сегменте сети.

Будьте внимательны, в том же сегменте сети находится машина, имеющая полученный MAC-адрес, а не IP-адрес. IP-адрес может принадлежать системе, находящейся на другом континенте, дело в том, что при передаче через маршрутизаторы MAC-адрес отправителя заменяется MAC-адресом маршрутизатора, и увидеть вы сможете только его. Узнать MAC-адрес реального отправителя в общем случае невозможно (да и нужно ли?).

После проверки, что вы в одном сегменте с жертвой, нужно узнать ее настройки маршрутизации, конкретнее - шлюз по умолчанию. Шлюзом называется маршрутизатор, обеспечивающий связь с другой сетью, например, с Internet. Для нахождения маршрутизаторов используется их вышеуказанная особенность: от одного MAC-адреса приходят пакеты со многих IP-адресов.

Можно воспользоваться пассивным методом - анализировать (с помощью программы-анализатора), с кем жертва обменивается пакетами, и таким образом вычислить его. Можно воспользоваться активным: назначить у себя в настройках жертву шлюзом и послать пакет, допустим, на www.microsoft.com с помощью команды ping. Жертва, получив пакет, перешлет его маршрутизатору. Но такой способ уже более опасен, чем предыдущий, т.к. теоретически жертва может это обнаружить.

Можно вообще не искать маршрутизатор, а оставить тот, что выдан вам (предполагается, что вы подключены к тому же провайдеру). Это еще более опасный способ, но в большинстве случаев это срабатывает.

Выяснив все детали о жертве (IP, MAC, шлюз по умолчанию), можно приступать ко второму этапу. Нужно дождаться момента, когда жертва не находится в сети (выключена), установить параметры идентичными ей и спокойно работать. Когда жертва вернется в сеть, работа нарушится, кроме того, она может получить предупреждение о конфликтующих IP-адресах, но это ничем особым не грозит, т.к. определить, кто именно из вашего сегмента сети пытается притвориться ею, невозможно.

Как можно достоверно идентифицировать машину в сети? Необходимо использовать авторизацию и криптографические системы защиты. Доверять компьютеру просто на основании его IP и MAC - небезопасно.

Полностью надежной защитой от использования чужого трафика было бы использование для связи с маршрутизатором протокола, поддерживающего безопасную авторизацию и защиту от перехвата.

Ниже приводятся два метода защиты. Оба - достаточно надежные, но не защищают от одновременной работы нескольких машин под одним IP. Впрочем, это можно заметить.

1. Microsoft Winsock proxy client-server. Программа, заменяющая стандартный winsock в win9x, NT и win2k у клиента и MS Proxy server. Авторизуется через netbios входом в домен NT.
2. Нужно самому написать "клиент", который бы периодически посылал запрос "откройте доступ такому-то IP" на маршрутизатор. Для отсылки запроса нужен пароль. Если запроса нет, то доступ закрывается. В результате только знающий пароль откроет себе доступ, злоумышленник, изменивший настройки, ничего сделать не сможет.

3. Команды для удаленной атаки в Linux

3.1. Поиск файлов с паролями

Команда find позволяет найти файлы по какому-то схожему признаку (дата создания, атрибут, имя файла и т.д.).

Формат команды find: find <директория начала поиска> <шаблон значение> Шаблоны бывают разные, вот некоторые из них:

- name - поиск по имени
- user - поиск на принадлежность к какому-либо пользователю
- group - аналогично, только с группой.
- perm - поиск по атрибуту
- type - поиск по типу (l - ссылка, f - файл, d - директория)

Ниже приведена команда для поиска файла shadow [6] – это файл теневых паролей. Набираем команду либо через cgi-скрипт, либо в telnet`e:

http://yourprov.com/path/to/script/test.pl?find -type f -name shadow

Если вы нашли его, то вам осталось только запустить программу John the Ripper и подождать, пока не будет взломан пароль.

3.2. Авторизация для сервисов

Большинство сервисов в Linux требуют авторизации, и их passwd-файлы также могут лежать на сервере с account-ми пользователей (порой даже незашифрованных). Пример: uucp, webmin, sockd и многие другие сервисы.

Чтобы посмотреть, а не запущены ли они на сервере, надо просто выполнить команду "ps ax" и искать нечто похожее на программу-сервис.

Допустим вы обнаружили, что на взламываемой машине работает uucp (unix to unix connect protocol). Смотрим в каталог /etc/uucp на предмет файла passwd (в большинстве случаев он там есть). Остается только взломать пароли при помощи программы John the Ripper.

3.3. Причины существования уязвимостей в UNIX-системах

На рис.1 перечислены причины, по которым происходит до 90% всех случаев вскрытия UNIX-хостов:

- Наличие демонов.

- Механизм *SUID/SGID*-процессов. Эти механизмы, являющиеся неотъемлемой частью идеологии UNIX, были и будут лакомым кусочком для хакеров, т. к. в этом случае пользователь всегда взаимодействует с процессом, имеющим большие привилегии, чем у него самого, и поэтому любая ошибка или недоработка в нем автоматически ведет к возможности использования этих привилегий.
- Излишнее доверие. Об этом уже достаточно говорилось выше. Повторим, что в UNIX достаточно много служб, использующих доверие, и они могут тем или иным способом быть обмануты.
- Человеческий фактор с весьма разнообразными способами его проявления - от легко вскрываемых паролей у обычных пользователей до ошибок у квалифицированных системных администраторов, многие из которых как раз и открывают путь для использования механизмов доверия.

Рассмотрим теперь более подробно причины, по которым оказываются уязвимы демоны и *SUID/SGID*-процессы:

- возможность возникновения непредусмотренных ситуаций, связанных с ошибками или недоработками в программировании;
- наличие скрытых путей взаимодействия с программой, называемых "люками" ;
- возможность подмены субъектов и объектов различным образом.

К первым можно отнести классическую ситуацию с переполнением буфера или размерности массива, ведущую к затиранию области стека и записи туда специальных команд, которые будут затем исполнены. Этот способ, несмотря на свою популярность, всегда будет системозависимым и ориентирован только на конкретную платформу и версию UNIX.

Хорошим примером непредусмотренной ситуации в многозадачной операционной системе является неправильная обработка некоторого специального сигнала или прерывания. Часто хакер имеет возможность смоделировать ситуацию, в которой этот сигнал или прерывание будет послано (в UNIX'е посылка сигнала решается очень просто: командой kill).

Наконец, одна из самых распространенных программистских ошибок является неправильная обработка входных данных (это является некоторым обобщением случая переполнения буфера.) Так в 1990 и 1995 годах были подвергнуты автоматизированному тестированию около 80 программ на 9 различных платформах UNIX . Специальная программа подавала на вход строки длиной до 100000 символов. Результатом явилось то, что 25- 33% в 1990 г. и 18- 23% в 1995 г. работали некорректно: зависали, сбрасывали аварийный дамп и т. п. (Интересно, что в коммерческих версиях UNIX этот процент доходил до 43, тогда как в свободно распространяемых он был меньше 10.) Впрочем, справедливости ради надо отметить, что только 2 программы-демона вели себя таким образом в 1990 г., а через 5 лет эти ошибки были исправлены. Ну, а если программа неправильно обрабатывает случайные входные данные, то очевидно, что можно подобрать такой набор специфических входных данных, которые приведут к желаемым для хакера последствиям. Примером этого может служить innd.

Люком или "черным входом" (*backdoor*) часто называют оставленную разработчиком недокументированную возможность взаимодействия (чаще всего входа в систему), например, известный только разработчику универсальный "пароль" . Люки оставляют в конечных программах вследствие ошибки, не убрав отладочный код или вследствие необходимости продолжения отладки уже в реальной системе в связи с ее высокой сложностью, или же их корыстных интересов. Люки - это любимый путь входа в удаленную систему не только у хакеров, но и у журналистов и режиссеров вкупе с подбором "главного" пароля перебором за минуту до взрыва, но в отличие от последнего способа люки реально существуют. Классический пример люка - это, конечно, отладочный режим в sendmail.

Рис.1

Наконец, вследствие многих особенностей UNIX, таких как асинхронное выполнение процессов, развитый командный язык и файловая система, злоумышленниками могут быть использованы механизмы подмены одного субъекта или объекта другим. Например, часто применяется замена имени файла, имени получателя и т. п. именем программы.

Аналогично может быть выполнена подмена некоторых специальных переменных. Так, для некоторых версий UNIX существует атака, связанная с подменой символа разделителя команд или опций "¦" на символ "/" . Это приводит к тому, что когда программа вызывает /bin/sh, вместо него вызывается файл bin с параметром sh в текущем каталоге. Наконец, очень популярным в UNIX видом подмены является создание ссылки (*link*) на критичный файл. После этого файл-ссылка некоторым образом получает дополнительные права доступа и тем самым осуществляется несанкционированный доступ к исходному файлу. Аналогичная ситуация с подменой файла возникает, если путь к файлу определен не как абсолютный (/bin/sh), а относительный (../bin/sh или $(BIN)/sh).

И последнее - нельзя приуменьшать роль человека при обеспечении безопасности любой системы. Возможно, он даже является слабейшим звеном. О необходимости выбора надежных паролей уже говорилось. Неправильное администрирование - такая же актуальная проблема, а для UNIX она особенно остра, т. к. сложность администрирования UNIX-систем давно уже стала притчей во языцех и часто именно на это упирают конкуренты. Но за все надо платить, и это обратная сторона переносимости и гибкости UNIX. Более того, если говорить о слабости человека, защищенные системы обычно отказываются и еще от одной из основных идей UNIX - наличия суперпользователя, имеющего доступ ко всей информации и никому не подконтрольного. Его права могут быть распределены среди нескольких людей:

администратора персонала, администратора безопасности, администратора сети и т. п., а сам он может быть удален из системы после ее инсталляции. В результате вербовка одного из администраторов не приводит к таким катастрофическим последствиям, как вербовка суперпользователя.

Настройки некоторых приложений, того же sendmail, настолько сложны, что для поддержания работоспособности системы требуется специальный человек - системный администратор, - но даже он не всегда знает о всех возможностях того или иного приложения и о том, как правильно их настроить. И если хакеры смогли проникнуть в систему, то это не всегда говорит о халатности администратора, а, зачастую, о его ограниченном знании того или иного продукта.

3.4. Windows NT, Windows 2000 или Linux

Согласно исследованию, проведенному компанией Attrition.org, 52 % из всех серверов, которые были взломаны в течение 2000 г., работали именно под Windows NT. На втором месте Win2000 – 29.55 % взломанных серверов. А вот Linux показала себя весьма надежной системой - серверы под Linux пострадали всего в 3.96 % случаев от общего количества успешных хакерских атак. Следующая таблица иллюстрирует общее количество и процентное соотношение взломанных операционных систем.

Таблица 1

Май 2001 г.

Операционная система	Количество	Проценты
Win-NT	594	52,24
Windows 2000	336	29,55
Linux (unknown distro)	49	4,31
Linux (RedHat)	45	3,96
Irix	37	3,25
Solaris	36	3,17
FreeBSD	15	1,32
BSDI	6	0,53
SCO	3	0,26
Generic Unix	2	0,18
Linux (Cobalt)	2	0,18
Linux (Debian)	2	0,18
Linux (SuSE)	2	0,18
Unknown	2	0,18
AIX	1	0,09
Compaq True64 Unix	1	0,09
Linux (Conectiva)	1	0,09
Linux (Mandrake)	1	0,09
MacOS	1	0,09
PowerBSD	1	0,09
Общее количество взломов на май 2001 г.	1137	100%

На рис.2 представлено количество взломанных операционных систем, а на рис.3 – их процентное соотношение.

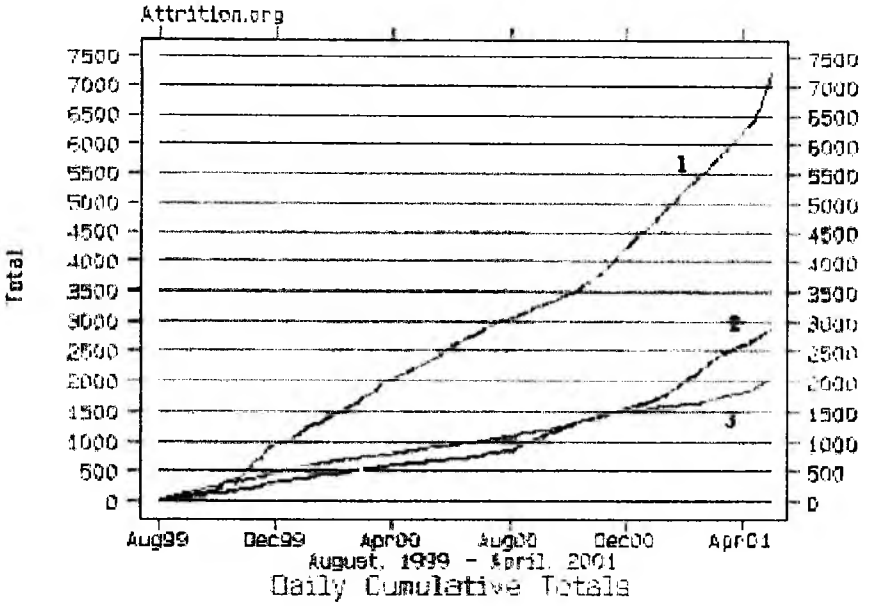


Рис. 2

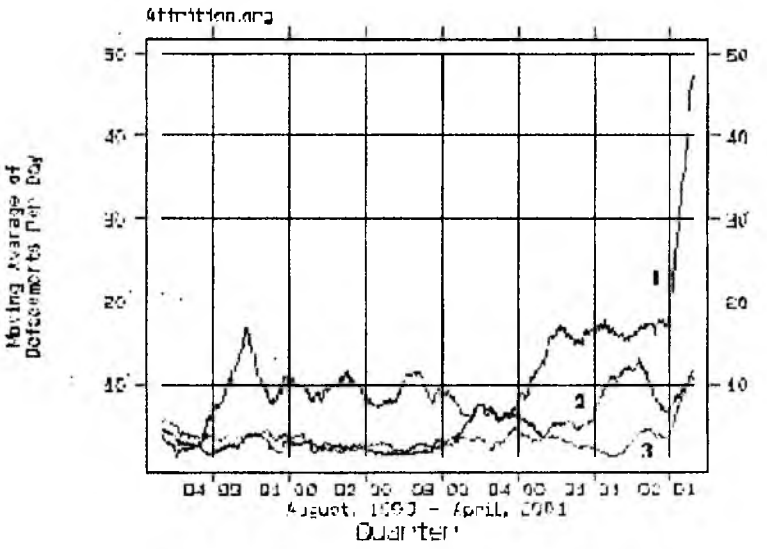


Рис. 3

Обозначение:

1 – WinNT, Win2000;

2 – Linux (все дистрибутивы);

3 – все остальные.

Заключение

Проведенный анализ возможностей и производительности Windows- и Unix-систем показал, что ОС Linux является несомненным лидером. Она предлагается большим числом поставщиков (нет угрозы монополии), хорошо масштабируется, более эффективна в использовании системных ресурсов, содержит средства удаленного администрирования, удаленного вычисления, многопользовательские возможности, полную палитру (профессионального) ПО, независимые стандарты разработчиков (POSIX). Далеко немаловажным фактором является цена: если Linux вместе со всем ПО Вы можете приобрести не более чем за 50$, то при приобретении Windows NT или Windows 2000 Вам придется платить за все программные

продукты отдельно и по баснословным ценам. Но самое важное при выборе между Windows NT и многими ОС Linux это то, что:

- ОС Linux дает Вам право выбора: любой тип "железа", CLI или GUI, коммерческое или GNU, разнообразное число поставщиков. Она динамична, т.е. можно собрать свое ядро, под свои конкретные нужды.
- Windows NT дает Вам сплошные ограничения: только Intel или Alpha; нет CLI, только GUI и только один GUI (нет того разнообразия оконных систем, которые есть под X-Window); только коммерческие MTA, только Microsoft и т.д. Windows NT статична, т.е. никогда не будет возможности собрать ядро на заказ. Одно ядро на все случаи.

Таким образом, на сегодняшний день Linux намного выгоднее чем Windows.

Список литературы: 1. 2001 Computer Crime and Security Survey // Computer Security Institute, San Francisco, March 12, 2001; 2. Common Criteria for Information Technology Security Evaluation (CCITSE) V2.1 // 1998; 3. *Зегжда Д. П., Ивашко А. М.* Как построить защищенную информационную систему/ Под науч. ред. Д. П. Зегжды и В. В. Платонова. СПб: Мир и семья 95, 1997. 312 с.: ил. 4. *Соболев. К.* Исследование системы безопасности в Windows NT, http://www.hackzone.ru/articles/ntadmin.html, 1998. 5. *Проскурин В. Г.* Проблемы защиты сетевых соединений в Windows NT, http://www.hackzone.rui/articles/ntadmintrap.html, 1999. 6. *Христов П. В.* Безопасность данных в ОС Unix / Открытые системы // М.: НИИСИ РАН, 1993.

Харьковский национальный университет радиоэлектроники

Поступила в редколлегию 13.04.2002