УДК681.3.06

# АНАЛИЗ СРЕДСТВ КОНТРОЛЯ ДОСТУПА И ЗАЩИТЫ ИНФОРМАЦИИ ОТ СЕТЕВЫХ АТАК

*ДУДАРЬ З.В., ЗБИТНЕВА М.В.*

Проводится анализ, выявляются достоинства и недостатки современных средств контроля доступа и защиты информации от сетевых атак. Результат исследования оформляется в виде классификации, для построения которой выбирается иерахический метод.

**Введение**

Современные сетевые технологии уже трудно представить без механизмов защиты. Однако при их детальном анализе возникают следующие вопросы: насколько эффективно реализованы имеющиеся механизмы защиты, каковы достоинства и недостатки технологий, на которых они основаны.

В настоящее время существует несколько технологий защиты информации и узлов в компьютерной сети, которые сильно отличаются между собой и используют различные подходы реализации поставленных задач:

– брандмауэры и технология фильтрации сетевого трафика;

– системы обнаружения атак;

– системы анализа защищенности.

Каждая из этих технологий имеет свои достоинства и недостатки, место на рынке средств защиты и область применения.

**1. Постановка и актуальность рассматриваемой задачи**

*Целью* работы является исследование особенностей средств контроля доступа и защиты информации от сетевых атак. *Задачи* определяются следующим образом:

– анализ технологий и механизмов, на которых основаны современные средства контроля доступа и защиты информации от сетевых атак, предоставляемых ими возможностей, а также их достоинств и недостатков;

– представление полученных результатов в виде классификации, сопоставляющей вероятные угрозы информации с соответствующими программными средствами защиты.

**2. Содержание исследования**

Технология фильтрации сетевого трафика. С технологией фильтрации сетевого трафика тесно связано понятие брандмауэра, который является инструментом управления доступом к защищаемой сети. Брандмауэр представляет собой систему или комбинацию систем, позволяющих разделить сеть на две или более частей и реализовать набор правил, определяющих условия прохождения пакетов из одной части в другую [1]. Он реализует политику сетевого доступа, вынуждая проходить все соединения с сетью через брандмауэр, где они могут быть проанализированы, а затем разрешены либо отвергнуты. Все брандмауэры выполняют одну и ту же задачу – защищают внутренние ресурсы от внешних атак. Однако механизмы их выполнения принципиально отличаются. На современном рынке доминируют брандмауэры следующих категорий:

– фильтры пакетов;

– фильтры пакетов с фиксацией состояния;

– шлюзы сеансового уровня;

– шлюзы прикладного уровня;

– брандмауэры экспертного уровня.

Данная классификация брандмауэров на категории выполнена на основании уровня в сетевой архитектуре, на котором располагается брандмауэр этой категории. Уровень расположения в модели соединения открытых систем (Open System Interconnection reference model – OSI) однозначно определяет возможности, предоставляемые системой для фильтрации передающихся по стеку пакетов. Чем ниже уровень – тем с более низкоуровневой информацией работает средство защиты.

Фильтр пакетов (packet filter) анализирует пакеты на сетевом уровне независимо от приложения [1]. Благодаря этому он обеспечивает высокую производительность при относительной дешевизне, однако одновременно является и самым слабым средством защиты, обеспечивая лишь минимальную защиту, так как проверяется только заголовок пакета, но не данные внутри него. Другим типом анализа является контекстная проверка сеансов между клиентами и серверами. Брандмауэры с контекстной проверкой, находясь на том же уровне, что и фильтры пакетов, принимают решения на основании высокоуровневой информации. Наиболее эффективным является объединение этих двух технологий – гибридная технология фильтрации сетевого трафика. Наиболее приемлема пакетная фильтрация с запоминанием исходного состояния (Stateful Dynamic Packet Filtering). Реализующие SPI брандмауэры сочетают скорость и гибкость фильтров пакетов с высокой степенью защиты средств прикладного уровня. Это компромиссное решение весьма эффективно в реализации жесткой политики безопасности периметра сети.

Шлюз сеансового уровня (circuit level gateway) – межсетевой экран, который исключает прямое взаимодействие между авторизированным клиентом и внешним хостом [1]. Сначала он принимает запрос доверенного клиента на определенные услуги и, после проверки допустимости запрошенного сеанса, устанавливает соединение с внешним хостом. При фильтрации пакетов шлюз сеансового уровня основывается на информации, содержащейся в заголовках паке-

тов сеансового уровня. После установления соединения от имени клиента шлюз просто копирует пакеты в обоих направлениях, не осуществляя их фильтрации. Поскольку шлюз уровня канала не может исследовать каждый пакет на прикладном уровне, а после завершения процедуры квитирования и установления соединения фильтрация вообще не применяется, это позволяет чужим приложениям использовать порты TCP, открытые для другого, законного приложения. Данная потенциальная опасность не позволяет использовать шлюз сеансового уровня как единственное средство защиты сети.

Шлюз прикладного уровня (application level gateways) – межсетевой экран, который исключает прямое взаимодействие между авторизированным клиентом и внешним хостом, фильтруя все входящие и исходящие пакеты на прикладном уровне модели OSI [1]. Связанные с приложением программы-посредники перенаправляют через шлюз информацию, генерируемую конкретными сервисами TCP/IP. Шлюз прикладного уровня обеспечивает более высокий уровень защиты, чем фильтр пакетов, но достигается это за счет потери "прозрачности" для приложений. Шлюз прикладного уровня выступает в роли посредника для таких приложений как электронная почта, FTP, Telnet и WWW. Брандмауэр проверяет формат приложений. Он может также осуществлять дополнительную аутентификацию и регистрацию информации, а также выполнять в случае необходимости преобразование данных. Недостатком использования шлюза прикладного уровня является непрозрачность для конечного пользователя.

Шлюзы сеансового и прикладного уровней основаны на использовании посредника, принимающего и организующего соединения по поручению клиента. Следствием этого являются дополнительные накладные расходы на обслуживание соединения. Решения на базе посредников обеспечивают так называемую защиту по периметру. Вместо того чтобы защищать все хосты, концепция защиты по периметру предусматривает укрепление защиты нескольких из них, так как посредник берет на себя защиту находящихся за ним хостов. Также шлюзы сеансового и прикладного уровней выполняют еще одну важную функцию защиты: они используются в качестве сервера-посредника (proxy server). Для этого им необходимо удовлетворять следующему условию – сервер-посредник выполняет процедуру трансляции адресов, при которой происходит преобразование внутренних IP-адресов в один адрес, который ассоциируется с брандмауэром. Это исключает прямой контакт между внутренней (авторизованной) и потенциально опасной внешней сетью. IP-адрес сервера-посредника становится единственным активным IP-адресом, который попадает во внешнюю сеть. Однако посредники, используемые шлюзом прикладного уровня, существенно отличаются от канальных посредников шлюзов сеансового уровня: во-первых, они связаны с приложениями, а во-вторых – могут фильтровать пакеты на прикладном уровне модели OSI. Вследствие этого данные посредники могут фильтровать отдельные виды команд или информации в протоколах прикладного уровня, которые им поручено обслуживать. Шлюзы прикладного уровня обеспечивают один из самых высоких на сегодняшний день уровней защиты. Однако такая высокая степень защиты имеет и обратную сторону – отсутствие "прозрачности" для пользователей. В реальной жизни они вносят задержки в процесс передачи данных или требуют от пользователей выполнения нескольких процедур регистрации при подключении к Интернет или корпоративной сети.

Межсетевой экран экспертного уровня (stateful inspection firewall) – проверяет содержимое принимаемых пакетов на трех уровнях модели OSI: сетевом, сеансовом и прикладном [2]. При выполнении этой задачи используются специальные алгоритмы фильтрации пакетов, с помощью которых каждый пакет сравнивается с известным шаблоном авторизированных пакетов, что, теоретически, должно обеспечить более эффективную их фильтрацию. В отличие от шлюзов прикладного уровня, брандмауэры экспертного уровня допускают прямые соединения между клиентами и внешними хостами.

Брандмауэры с возможностью посредничества (т.е. шлюзы сеансового и прикладного уровней) также можно классифицировать по функциональным возможностям и назначению используемых ими программ-посредников.

Существуют следующие типы посредничества:

– NAT-proxy – самый простой вид посредничества. Этот посредник работает прозрачно для пользователя, никаких специальных настроек в программах не требуется. Влиять на работу "общего доступа" (например, ограничивать список доступных сайтов для отдельных пользователей) он не может. Посредники данного типа не "вникают" в тонкости тех прикладных протоколов, которые через себя пропускают, поэтому и не имеют средств управления ими;

– HTTP-proxy – самый распространенный тип посредника. Он предназначен для организации работы браузеров и других программ, использующих протокол HTTP. Браузер передает серверу-посреднику URL ресурса, посредник получает его с запрашиваемого Web-сервера (или с другого сервера-посредника) и направляет браузеру;

– FTP-proxy бывает двух основных видов в зависимости от протокола работы самого посредника. С ftp-серверами этот посредник, конечно, всегда работает по протоколу FTP. А вот с клиентскими программами – браузерами и ftp-клиентами он может работать как по FTP, так и по HTTP. Второй способ удобнее для браузеров, так как исторически является для них "родным". Браузер запрашивает ресурс у посредника, указывая протокол целевого сервера в URL - http или ftp. В зависимости от этого посредник выбирает

протокол работы с целевым сервером, а протокол работы с браузером не меняется – HTTP. Поэтому, как правило, функцию работы с FTP-серверами также вставляют в HTTP-proxy , т.е. HTTP-proxy, описаный выше, обычно с одинаковым успехом работает как с HTTP, так и с FTP-серверами. Но при "конвертировании" протоколов FTP и HTTP теряется часть полезных функций протокола FTP. Поэтому специализированные клиенты предпочитают и специальный сервер-посредник, работающий с обеими сторонами по FTP. Такой тип посредничества называется FTP-gate, чтобы подчеркнуть отличие от FTP-proxy в составе HTTP-proxy;

– HTTPS-proxy – фактически часть HTTP-proxy. "S" в названии означает "secure", т.е. безопасный. Несмотря на то, что программно это часть HTTP-proxy , обычно HTTPS выделяют в отдельную категорию. Этот протокол применяют, когда требуется передача секретной информации, например, номеров кредитных карт. Поэтому в таких случаях применяют secure HTTP – всё передаваемое при этом шифруется. Серверу-посреднику при этом дается только команда "соединиться с таким-то сервером", и после соединения посредник передает в обе стороны шифрованный трафик. Но при этом отсутствует возможность узнать подробности (соответственно, отсутствуют и многие средства управления доступом – такие, как фильтрация изображений – не могут быть реализованы для HTTPS, так как посреднику в этом случае неизвестно, что именно передается). Собственно, в процессе шифрации/дешифрации посредник тоже участия не принимает – это делают клиентская программа и целевой сервер. Наличие команды "соединиться с таким-то сервером" в HTTPS-proxy приводит к интересному и полезному побочному эффекту, которым все чаще пользуются разработчики клиентских программ. Так как после соединения с указанным сервером HTTPS-proxy лишь пассивно передает данные в обе стороны, не производя никакой обработки этого потока вплоть до отключения клиента или сервера, это позволяет использовать посредник для передачи почти любого TCP-протокола, а не только HTTP, т.е. HTTPS-proxy одновременно является и простым POP3-proxy, SMTP-proxy, IMAP-proxy, NNTP-proxy и т.д. Никаких модификаций целевого сервера не требуется;

– Mapping-proxy – способ заставить работать через посредника те программы, которые умеют работать с Интернетом только напрямую. При настройке такого сервиса администратор как бы создает "копию" целевого сервера, но доступную через один из портов сервера-посредника для всех клиентов локальной сети – устанавливает локальное "отображение" заданного сервера. Неудобство Mapping-proxy в том, что для каждого необходимого внешнего сервера нужно вручную устанавливать отдельный порт на посреднике. Но зато не требуется модификации ни серверов, ни клиентов. Особенно это помогает в случае необходимости "проксирования" многочисленных собственных протоколов, реализованных в играх или финансовых программах. Они часто игнорируют существование посредника и стандартных протоколов. Такие программы можно "обмануть" и направить через посредника практически всегда, если они не делают другой ошибки – передачи клиентского IP-адреса внутри протокола и пытаются с ним соединяться напрямую еще раз (что невозможно, так как локальные адреса недоступны извне);

– Socks-proxy – на сегодняшний день существует две версии протокола: Socks4 и Socks5. Несмотря на то, что Socks5 более функционален, сейчас в одинаковой степени распространены серверы с поддержкой как старой, так и новой версии. Протокол представляет собой транслятор (что-то вроде сервера-посредника), но в отличие от обычных посредников Socks-клиент находится между прикладным и транспортным уровнем в сетевой модели, а Socks-сервер находится на прикладном уровне. Это означает, что такой сервер не привязан больше к протоколам высокого уровня. Сам протокол разработан для того, чтобы приложения, работающие на основе TCP и UDP, могли использовать ресурсы сети, доступ к которым ограничен архитектурой или настройками (например, доступ к ресурсам Интернета из локальной сети для приложений, у которых вообще не предусмотрена работа с использованием посредника).

По методу физической реализации брандмауэры можно классифицировать следующим образом [1]:

– хост-ориентированные брандмауэры; в данном случае брандмауэр устанавливается и выполняется как отдельное приложение, поверх коммерческой операционной системы. Этот прием известен под названием укрепления (hardening) операционной системы и должен осуществляться на любой системе, подключенной к небезопасной сети. Применение большинства брандмауэров-приложений, выполняющихся поверх существующих операционных систем, не исключает ряда дополнительных мер по улучшению безопасности хоста, включая замену некоторых из сетевых демонов операционной системы на более надежные, замену или изменение стека TCP/IP, изменение файлов запуска, файлов конфигурации, записей в системном реестре и добавление новых процессов;

– брандмауэры, ориентированные на маршрутизаторы – в общей архитектуре безопасности, реализация маршрутизатора в качестве устройства защиты весьма популярна. Маршрутизатор, выполняющий предварительную фильтрацию пакетов, снижает нагрузку на брандмауэры следующего уровня. Это оптимизирует архитектуру брандмауэров и маршрутизаторов: маршрутизатор выполняет предварительную проверку пакетов, а брандмауэр, обладающий большими функциональными возможностями, исследует только те их них, которые проходят через первый набор фильтров;

– интегрированные хост - ориентированные брандмауэры – являются, как правило, составной частью

программного обеспечения, установленного на одиночной системе для защиты только ее. Если к небезопасной сети подключены только одна или две машины, то хост - ориентированные брандмауэры являются наилучшим экономическим решением. В корпоративной среде, где защиты требуют сотни или тысячи машин, подключенных в корпоративную сеть, интегрированный хост - ориентированный брандмауэр не обеспечит ни централизованного управления, ни достаточной масштабируемости;

– брандмауэры - устройства – это специализированные устройства (состоящие из аппаратных средств и программного обеспечения), предназначенные для контроля поступающего от них трафика и принятия решения о необходимости его передачи. Такие устройства обладают чрезвычайно высокой производительностью, так как они не обременены множеством дополнительных функций операционной системы. Устройства обычно настраивают при помощи интерфейса командной строки, их собственных утилит или Web-ориентированных интерфейсов, предоставляющих доступ по протоколу HTTP.

Брандмауэры могут быть сконфигурированы в виде одной из нескольких архитектур, что обеспечивает различные уровни безопасности при различных затратах на установку и поддержание работоспособности. Типичными архитектурами брандмауэра являются:

– хост, подключенный к двум сегментам сети – это такой хост, который имеет более одного интерфейса с сетью, причем каждый интерфейс с сетью подключен физически к отдельному сегменту сети. Самым распространенным примером является хост, подключенный к двум сегментам. В этой конфигурации ключевым принципом обеспечения безопасности является запрет прямой маршрутизации трафика из небезопасной сети в доверенную – брандмауэр всегда должен быть при этом промежуточным звеном;

– экранированный хост – при архитектуре такого типа используется хост (называемый хостом-бастионом), с которым может установить соединение любой внешний хост, но запрещен доступ ко всем другим внутренним, менее безопасным хостам. Для этого фильтрующий маршрутизатор конфигурируется так, что все соединения с внутренней сетью из внешних сетей направляются к хосту-бастиону;

– экранированная подсеть – архитектура экранированной сети по существу совпадает с архитектурой экранированного хоста, но добавляет еще одну линию защиты с помощью создания сети, в которой находится хост-бастион, отделенной от внутренней сети. Экранированная подсеть должна внедряться с помощью добавления сети-периметра для того, чтобы отделить внутреннюю сеть от внешней. Это гарантирует, что даже при успехе атаки на хост-бастион атакующий не сможет пройти дальше сети-периметра из-за того, что между внутренней сетью и сетью-периметром находится еще один экранирующий маршрутизатор.

Системы обнаружения атак. Технологии, по которым строятся системы обнаружения атак (intrusion detection systems), принято условно делить на две категории:

– обнаружение аномального поведения (anomaly detection);

– обнаружение злоупотреблений (misuse detection).

В практической деятельности применяется другая классификация, учитывающая принципы практической реализации таких систем:

– обнаружение атак на уровне сети (network-based);

– обнаружение атак на уровне хоста (host-based).

Системы класса host-based можно разделить еще на три подуровня:

– обнаруживающие атаки на операционные системы (OS IDS);

– обнаруживающие атаки на системы управления базами данных (DBMS IDS);

– обнаруживающие атаки на конкретные приложения (Application IDS).

Выделение обнаружения атак на системы управления базами данных (СУБД) в отдельную категорию связано с тем, что современные СУБД уже вышли из разряда обычных приложений и по многим своим характеристикам, в том числе и по сложности, приближаются к операционным системам.

Классификация систем обнаружения атак – по этапам их осуществления:

– системы обнаружения атак в процессе их осуществления – функционирующие на этапе осуществления атак и позволяющие обнаружить их в процессе реализации, т.е. в режиме реального (или близкого к реальному) времени;

– системы обнаружения совершенных атак – позволяющие обнаружить уже совершенные атаки.

Преимущества и недостатки каждого из подходов зависят от того, как будет применяться система обнаружения атак. В случае высококритичных систем обнаружение атак в реальном режиме времени является обязательным, так как злоумышленник может проникнуть в систему, сделать все, что необходимо, и исчезнуть в течение нескольких минут или даже секунд. Однако и автономный анализ имеет немалое значение. Он позволяет проводить более подробное исследование того, когда и как злоумышленники проникли в вашу систему. Реализовать такой анализ можно по-разному, начиная от простой генерации отчета с информацией обо всех или выбранных прошедших событиях и заканчивая воспроизведением (playback) в реальном времени всех действий, производимых при атаке. Важно, чтобы системы обнаружения атак поддерживали эту возможность, предоставляя эффективные средства управления данными.

Существуют следующие методы обнаружения вторжений, различающиеся подходами к анализу событий, и определения их безопасности:

– сигнатурный анализ;

– анализ протоколов;

– статистический метод;

– обманный метод;

– метод отражения неизвестных атак.

Системы анализа защищенности. Средства анализа защищенности (security assessment) исследуют сеть и ищут "слабые" места в ней, анализируют полученные результаты и на их основе создают различного рода отчеты. В некоторых системах вместо "ручного" вмешательства со стороны администратора найденная уязвимость будет устраняться автоматически. Технология анализа защищенности является действенным методом реализации политики сетевой безопасности прежде, чем осуществится попытка ее нарушения снаружи или изнутри организации. Функционировать такие средства могут:

– на сетевом уровне (network-based);

– на уровне операционной системы (host-based);

– на уровне приложения (application-based).

Наибольшее распространение получили средства анализа защищенности сетевых сервисов и протоколов. Связано это, в первую очередь, с универсальностью используемых протоколов. Изученность и повсеместное использование таких протоколов [3], как IP, TCP, HTTP, FTP, SMTP и т.п., позволяют с высокой степенью эффективности проверять защищенность информационной системы, работающей в данном сетевом окружении. Вторыми по распространенности являются средства анализа защищенности операционных систем. Связано это также с универсальностью и распространенностью некоторых операционных систем (например, UNIX и Windows NT). Однако из-за того, что каждый производитель вносит в операционную систему свои изменения (ярким примером является множество разновидностей ОС UNIX), средства анализа защищенности ОС анализируют в первую очередь параметры, характерные для всего семейства одной ОС. И лишь для некоторых систем анализируются специфичные для нее параметры.

Помимо обнаружения уязвимостей, при помощи средств анализа защищенности можно быстро определить все узлы корпоративной сети, доступные в момент проведения тестирования, выявить все используемые в ней сервисы и протоколы, их настройки и возможности для несанкционированного воздействия (как изнутри корпоративной сети, так и снаружи). Также эти средства вырабатывают рекомендации и пошаговые меры, позволяющие устранить выявленные недостатки.

Существует два основных механизма, при помощи которых сканер проверяет наличие уязвимости:

– сканирование (scan);

– зондирование (probe).

Сканирование – механизм пассивного анализа, с помощью которого сканер пытается определить наличие уязвимости без фактического подтверждения ее наличия – по косвенным признакам. Этот метод является наиболее быстрым и простым для реализации. В терминах компании ISS данный метод получил название "логический вывод" (inference). Согласно компании Cisco этот процесс идентифицирует открытые порты, найденные на каждом сетевом устройстве, и собирает связанные с портами заголовки (banner), найденные при сканировании каждого порта. Каждый полученный заголовок сравнивается с таблицей правил определения сетевых устройств, операционных систем и потенциальных уязвимостей. На основе проведенного сравнения делается вывод о наличии или отсутствии уязвимости.

Зондирование – механизм активного анализа, который позволяет убедиться, присутствует или нет на анализируемом узле уязвимость. Зондирование выполняется путем имитации атаки, использующей проверяемую уязвимость. Этот метод более медленный, чем сканирование, но почти всегда гораздо более точный, чем он. В терминах компании ISS данный метод получил название "подтверждение" (verification). Согласно компании Cisco этот процесс использует информацию, полученную при сканировании ("логическом выводе"), для детального анализа каждого сетевого устройства. Этот процесс также использует известные методы реализации атак для того, чтобы полностью подтвердить предполагаемые уязвимости и обнаружить другие уязвимости, которые не могут быть обнаружены пассивными методами, например подверженность атакам типа "отказ в обслуживании" (denial of service).

На практике указанные механизмы реализуются следующими несколькими методами:

– проверка заголовков (banner check);

– активные зондирующие проверки (active probing check);

– имитация атак (exploit check).

Механизм проверки заголовков представляет собой ряд проверок типа "сканирование" и позволяет делать вывод об уязвимости, опираясь на информацию в заголовке ответа на запрос сканера. Это наиболее быстрый и простой для реализации метод проверки присутствия на сканируемом узле уязвимости. Однако за этой простотой скрывается немало проблем. Эффективность проверок заголовков достаточно эфемерна. И вот почему. Во-первых, всегда можно изменить текст заголовка, предусмотрительно удалив из него номер версии или иную информацию, на основании которой сканер строит свои заключения. Во-вторых, зачастую версия, указываемая в заголовке ответа на запрос, не всегда говорит об уязвимости программного обеспечения. В-третьих, устранение

уязвимости в одной версии еще не означает, что в следующих версиях эта уязвимость отсутствует.

Активные зондирующие проверки – также относятся к механизму "сканирования". Однако они основаны не на проверках версий программного обеспечения в заголовках, а на сравнении "цифрового слепка" (fingerprint) фрагмента программного обеспечения со слепком известной уязвимости. Аналогичным образом поступают антивирусные системы, сравнивая фрагменты сканируемого программного обеспечения с сигнатурами вирусов, хранящимися в специализированной базе данных. Разновидностью этого метода являются проверки контрольных сумм или даты сканируемого программного обеспечения, которые реализуются в сканерах, работающих на уровне операционной системы. Специализированная база данных (в терминах компании Cisco - база данных по сетевой безопасности) содержит информацию об уязвимостях и способах их использования (атаках). Эти данные дополняются сведениями о мерах их устранения, позволяющих снизить риск безопасности в случае их обнаружения. Зачастую эта база данных используется и системой анализа защищенности, и системой обнаружения атак. Этот метод также достаточно быстр, но реализуется труднее, чем проверка заголовков.

Имитация атак – данные проверки относятся к механизму "зондирования" и основаны на эксплуатации различных дефектов в программном обеспечении. Некоторые уязвимости не обнаруживают себя, пока не будет сделана попытка к их осуществлению. Для этого против подозрительного сервиса или узла запускаются реальные атаки. Проверки заголовков осуществляют первичный осмотр сети, а метод "exploit check", отвергая информацию в заголовках, позволяет имитировать реальные атаки, тем самым с большей эффективностью (но меньшей скоростью) обнаруживая уязвимости на сканируемых узлах. Имитация атак является более надежным способом анализа защищенности, чем проверки заголовков, и обычно более надежны, чем активные зондирующие проверки. Однако имитация атак не всегда может быть реализована. Такие случаи можно разделить на две категории: ситуации, в которых тест приводит к "отказу в обслуживании" анализируемого узла или сети, и ситуации, при которых уязвимость в принципе не годится для реализации атаки на сеть.

По функциональному назначению можно выделить еще одно инструментальное средство, которое дает возможность самостоятельно исследовать сетевую активность, – анализатор сетевого трафика [2]. С его помощью администратор может получить полную последовательность действий и изучить взаимодействие распределенного программного обеспечения. В отличие от сканеров уязвимостей, которые выполняют свою работу автоматически, предоставляя только результат в виде отчета, анализатор предоставляет серию пакетов. Дальнейшая их обработка и анализ возлагается целиком на администратора.

Средства контроля доступа. Как показывает практика, вход пользователя в систему – одно из наиболее уязвимых мест защиты; известно множество случаев взлома пароля, входа без пароля, его перехвата. Под контролем доступа понимается ограничение возможностей использования ресурсов системы программами, процессами или другими системами в соответствии с установленной политикой безопасности. Поэтому для подтверждения подлинности субъекта, законности его прав на данный объект или на определенные действия, а также для обеспечения работы субъекта в информационной системе необходимы надежные механизмы контроля доступа [4, 5], такие как:

– идентификация – процесс распознавания элемента системы, обычно с помощью заранее определенного идентификатора или другой априорной информации; каждый субъект или объект должен быть однозначно идентифицируем;

– аутентификация – проверка идентификации пользователя, процесса, устройства или другого компонента системы (обычно осуществляется перед разрешением доступа); а также проверка целостности данных при их хранении или передаче для предотвращения несанкционированной модификации;

– авторизация – предоставление субъекту прав на доступ к объекту.

Управление доступом может быть достигнуто при использовании дискреционного или мандатного управления доступом.

Дискреционное управление доступом – наиболее общий тип управления, используемого в информационной системе. Основной принцип этого вида защиты состоит в том, что индивидуальный пользователь или программа, работающая от имени пользователя, имеет возможность явно определить типы доступа, которые могут осуществить другие пользователи (или программы, выполняющиеся от их имени) к информации, находящейся в ведении данного пользователя. Дискреционное управление доступом отличается от мандатной защиты тем, что оно реализует решения по управлению доступом, принятые пользователем [2].

Мандатное управление доступом реализуется на уровне результатов сравнения уровня допуска пользователя и степени конфиденциальности информации. Существуют механизмы управления доступом, поддерживающие степень его детализации на уровне следующих категорий:

– владелец информации;

– заданная группа пользователей;

– все другие авторизованные пользователи.

Это позволяет владельцу файла или каталога иметь права доступа, отличающиеся от прав всех других пользователей, и определять особые права доступа для указанной группы людей или всех остальных пользователей.

В большинстве информационных систем используется механизм идентификации и аутентификации на основе схемы идентификатор пользователя / пароль. Это самый простой, однако и самый ненадежный и уязвимый способ. Однако существуют и другие альтернативные и более защищенные типы механизмов защиты, которые могут быть реализованы для обеспечения службы идентификации и аутентификации:

– механизм, основанный на паролях – уязвимый и не обеспечивающий достаточную надежность;

– механизм, основанный на интеллектуальных картах – смарт-карта реализует аутентификацию с помощью схемы – запрос/ответ в реальном масштабе времени, что помогает предотвратить получение злоумышленником неавторизованного доступа путем воспроизведения сеанса регистрации пользователя;

– механизм, основанный на биометрии – базируется на опознавании пользователя по сугубо индивидуальным характеристикам;

– механизмы блокировки ПК или автоматизированного рабочего места – позволяет пользователям оставаться зарегистрированными, не делая при этом свое место потенциально доступным для злоумышленников;

– завершение соединения после нескольких ошибок при регистрации – позволяет исключить подбор пароля с помощью перебора возможных комбинаций;

– уведомление пользователя о "последней успешной регистрации" и "числе ошибок при регистрации" – информирует пользователя об использовании его регистрационного имени.

Методы, основанные на биометрии, достаточно сложны и требуют специального оборудования. Известны следующие методы:

– персональные: отпечатки пальцев, строение лица;

– квазистатические: геометрия руки, особенность глаз, отпечатки ладоней, рисунок кровеносных сосудов;

– квазидинамические: пульс, баллистокардиография, энцефалография;

– динамические: голос, почерк, стиль печатания.

Применение этих механизмов в разной мере необходимо для обеспечения предотвращения несанкционированного использования ресурсов системы и данных в зависимости от степени конфиденциальности защищаемой информации и принятой политики безопасности.

### 3. Выводы и перспективы дальнейших разработок

*Научная новизна*: результатом исследования является классификация и выделение существующих технологий современных средств защиты, их достоинств и недостатков, а также сферы применения. Это используется при комплексной оценке выбора средств защиты информации.

**Литература**: **1.** *Страссберг Кейт Е., Гондек Ричард Г., Роли Гари*. Полный справочник по брандмауэрам. М.: Вильямс, 2004. 836 с. **2.** *Домарев В.В*. Безопасность информационных технологий. Методология создания систем защиты. К.: ООО "ТИД "ДС", 2001. 688 с. **3.** *Филимонов А.* Протоколы Интернета. СПб.: изд. "BHV-СПб", 2003. 516 с. **4.** *Медведовский И.Д., Семьянов П.В., Платонов В.В*. Атака через Интернет. НПО "Мир и семья-95", 1997. **5.** *Козлов Д.А., Парандовский А.А., Парандовский А.К.* Энциклопедия компьютерных вирусов. М.: "СОЛОН-пресс", 2001. 464 с.

Поступила в редколлегию 19.01.2007

**Рецензент:** д-р техн. наук, проф. Самойленко Н.И.

**Дударь Зоя Владимировна**, канд. техн. наук, проф., зав. кафедрой программного обеспечения ЭВМ, декан ФПО ХНУРЭ. Научные интересы: математическое и программно-техническое обеспечение взаимодействия крупномасштабных систем баз данных в динамическом окружении, дистанционное образование. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 7021-446.

**Збитнева Майя Вячеславовна**, канд. техн. наук, старший преподаватель кафедры программного обеспечения ЭВМ ХНУРЭ. Научные интересы: автоматизированные системы диспетчерского управления электрическими сетями, стеганография, интеллектуальные агенты, программирование под Web. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 7021-446.

УДК681.30001.571

# ИСПОЛЬЗОВАНИЕ ИСКУССТВЕННОЙ НЕЙРОННОЙ СЕТИ В ПОДСИСТЕМЕ ВВОДА ГОЛОСОВОЙ ИНФОРМАЦИИ САПР ТП РОБОТИЗИРОВАННОГО ПРОИЗВОДСТВА

*НЕВЛЮДОВ И.Ш., ЦЫМБАЛ А.М., МИЛЮТИНА С.С.*

Для голосового задания управляющих команд робота предлагается использование искусственной нейронной сети (ИНС). Разрабатывается программное обеспечение, реализующее многослойный персептрон на основе переходных функций различного вида. Полученные результаты используются при создании подсистемы ввода голосовой информации САПР технологических процессов роботизированного производства.

### 1. Введение

Техническое зрение и тактильное очувствление призваны повысить возможности робота в восприятии информации об изменениях внешней среды. Применение средств адаптации роботов позволяет освободить человека от выполнения однообразных, повторяющихся операций и возложить на него исполнение более сложных в интеллектуальном отношении задач, например, по наблюдению за работой роботов, выполняющих эти операции.

Командовать машинами человек может не только с помощью кнопок, клавиатур, тумблеров управления, передающих цифровые или аналоговые сигналы кон-