Ставропольский филиал
ГОУ ВПО «Московский государственный гуманитарный университет
имени М.А. Шолохова»

С. И. МАКАРЕНКО

ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ

Учебное пособие

Ставрополь
СФ МГГУ им. М. А. Шолохова
2009

УДК 681.322
ББК 32.973

Макаренко С. И. Информационная безопасность: учебное пособие. – Ставрополь: СФ МГГУ им. М. А. Шолохова, 2009. – 372 с.: ил.

Рецензенты:

доцент кафедры прикладной информатики и математики Ставропольского филиала Московского государственного гуманитарного университета имени М. А. Шолохова кандидат технических наук, доцент Федосеев В. Е.,

доцент кафедры прикладной информатики и математики Ставропольского филиала Московского государственного гуманитарного университета имени М. А. Шолохова кандидат технических наук Дятлов Д. В.

Учебное пособие адресовано студентам, обучающимся по специальности 080801 (351400) «Прикладная информатика в экономике» изучающих дисциплины «Информационная безопасность» и «Безопасность компьютерных систем», а также может быть использовано специалистами в области проектирования и организации систем информационной безопасности.

Утверждено на заседании кафедры прикладной информатики и математики Ставропольского филиала Московского государственного гуманитарного университета имени М. А. Шолохова в качестве методического пособия для студентов по специальности 080801 (351400) - «Прикладная информатика в экономике».

Оглавление

Список сокращений

AD - Active Directory - служба каталогов, являющаяся масштабируемой структурой домена управляемого ОС Windows;
APEG - Automatic Patch-based Exploit Generation – технология автоматической генерации кода атаки по имеющейся коду заплатки к ПО;
ARP - Address Resolution Protocol – протокол разрешения адресов;
ARPANET - Advanced Research Projects Agency Network – глобальная сеть, которая являлась праобразом сети Internet;
CA - Certificate Authority - центральное бюро сертификатов;
CRC - Cyclic Redundancy Code — алгоритм вычисления контрольной суммы, предназначенный для проверки целостности передаваемых данных;
DES - Data Encryption Standard — симметричный алгоритм шифрования,
DMZ - Demilitarized Zone — демилитаризованная зона;
DNS - Domain Name System – система доменных имён;
DSL - Digital Subscriber Line - цифровая абонентская линия;
EAP - Extensible Authentication Protocol — расширяемый протокол аутентификации;
EFS - Encrypting File System - шифрующая файловая система;
FTP - File Transfer Protocol – протокол передачи файлов;
GC - Global Catalog - глобальный каталог в службе каталогов ОС Windows;
GP - Group Policies - групповые политики в ОС Windows;
GSP - Generic Services Proxy - технология модуля доступа прикладного уровня для поддержки внешних протоколов обеспечения безопасности;
HTTP - HyperText Transfer Protocol — протокол передачи гипертекстовых Internet страниц;
HTTPS - Hypertext Transfer Protocol Secure — расширение протокола HTTP, поддерживающее шифрование;
ID IDentification – идентификатор;
IDS - Intrusion Detection System - системы обнаружения вторжений;
IIS Internet Information Services - сервер службы Web определяющий тип запрашиваемого ресурса;
IP - Internet Protocol Address — уникальный сетевой адрес узла в компьютерной сети;
ISDN - Integrated Services Digital Network — цифровая сеть интегрального обслуживания
ISP - Internet Service Provider — поставщик интернет-услуги (провайдер);

LSA - Local Security Authority — локальный администратор безопасности используемый в ОС Windows;

IT - Information Technology – информационные технологии;

MAC – Media Access Control – адрес по которому ведется доступ абонентов к общему каналу связи на канальном уровне OSI;

MAC – Message Authentication Code – идентификационный код сообщения;

NFS - Network File System – сетевая файловая система;

NIDS - Network Intrusion Detection System - сетевая система обнаружения вторжений;

OSI - Open System Interconnection – Эталонная модель взаимодействия открытых систем;

OSPF - Open Shortest Path First — протокол динамической маршрутизации, основанный на технологии отслеживания состояния канала;

OU - Organizacion Unit - организационная единица описывающая тип объектов службы каталога ОС Windows;

PAE - процесс доступа через порт;

PGP - Pretty Good Privacy – протокол с открытым ключом для шифрования сообщений электронной почты;

RPC - Remote Procedure Call - Удалённый вызов процедур;

RSA - криптографический алгоритм с открытым ключом;

SAM - Security Account Manager — RPC-сервер ОС Windows, оперирующий базой данных учетных записей;

SID - Security IDentifier - идентификатор безопасности, используемый в ОС Windows;

SRM - Security Reference Monitor - Диспетчер доступа ОС Windows;

SSH - Secure SHell — сетевой протокол прикладного уровня «безопасная оболочка», позволяющий туннелировать TCP-соединения;

SSL - Secure Socket Layer – протокол защищенной связи через Интернет по системе «клиент – сервер»;

TCP - Transmission Control Protocol -протокол управления передачей;

TLS - Transport Layer Security — протокол обеспечения безопасности транспортного уровня;

UDP - User Datagram Protocol — протокол передачи пользовательских данных;

URL Uniform Resource Locator – формат символьного указателя ресурса в сети Internet;

VPN - Virtual Private Network - виртуальная частная сеть;

WEP - Wired Equivalent Privacy - алгоритм для обеспечения безопасности сетей Wi-Fi;

Wi-Fi - беспроводная сеть стандарта IEEE 802.11;

WLAN - Wireless Local Area Network — беспроводная локальная сеть;

АБС - автоматизированная банковская система;
АНБ - агентство национальной безопасности;
АС - автоматизированная система;
АСОИ - автоматизированная система обработки информации;
ВК - виртуальный канал;
ВС – вычислительная система;
ВТ - виртуальный терминал;
ГВС - глобальная вычислительная сеть;
ИБ - информационная безопасность;
ИС - информационная система;
ИТ - информационные технологии;
КС - компьютерная система;
ЛВС - локальная вычислительная сеть;
МСЭ - межсетевой экран;
МЭ - межсетевой экран;
НСД - несанкционированный доступ;
ОЗУ - оперативное запоминающие устройство;
ОК - общие критерии;
ОС - операционная система;
ПЗ - профиль защиты;
ПЗУ - постоянное запоминающие устройство;
ПИБ - политика информационной безопасности;
ПО - программное обеспечение;
РВС – распределенная вычислительная система;
РД – руководящий документ;
РПС - разрушающее программное средство;
СБИ - система безопасности информации;
СИУБ - система управления информационной безопасности;
УА - удаленная атака;
ФАПСИ - федеральное агентство правительственной связи;
ФБР - федеральное бюро расследований;
ЦРУ - центральное разведывательное управление;
ЭВМ - электронно-вычислительная машина;
ЭЦП - электронная цифровая подпись.

Введение

Учебное пособие написано по опыту преподавания автором дисциплин «Информационная безопасность» и «Безопасность компьютерных систем» в Ставропольском филиале МГГУ им. М. А. Шолохова и в первую очередь адресовано студентам, обучающимся по специальности «Прикладная информатика в экономике». Также учебное пособие может быть использовано специалистами в области проектирования и организации систем информационной безопасности.

Учебное пособие учитывает требования государственного образовательного стандарта и структурно соответствует учебной программе и тематическому плану изучения дисциплины «информационная безопасность». Отдельные части пособия соответствуют темам дисциплины, а отдельные главы – лекционным занятиям. Дополнительно глава 14 «Основы борьбы с вредоносными программами», глава 20 «Обеспечение безопасности в операционных системах Windows 2000/XP и Windows 2003 server», глава 21 «Использование службы каталогов и групповых политик в Windows 2000/XP и Windows 2003 server» и глава 22 «Основы безопасности операционных систем семейства Unix» могут быть использованы для проведения практических занятий и самостоятельно изучения студентами соответствующего материала с использованием ПК.

При написании пособия автор придерживался принципа необходимости дополнения общетеоретических и концептуальных основ информационной безопасности, изучение которых предусмотрено государственным образовательным стандартом (изложены в главах 1-10, 13), практическими сведениями по способам обеспечения безопасности ЭВМ и компьютерных сетей, которые являются актуальными для специалистов в информационной сфере. В связи с этим автор постарался расширить и дополнить текст, поэтому пособие содержит много дополнительных и справочных сведений о безопасности компьютерных сетей и ориентировано на читателя целью которого является более глубокое изучение вопросов информационной безопасности по сравнению с материалом, изучаемым на лекциях по дисциплине. Вместе с тем, пособие может быть использовано в качестве конспекта лекций, так как в тексте пособия материал, рекомендуемый к конспектированию на лекциях, выделен *курсивом*.

При составлении учебного пособия автор ориентировался на известные учебные материалы в предметной области, а также использовал ресурсы сети Internet посвященные вопросам информационной безопасности.

В основу глав учебного пособия был положен материал следующих источников:

- глава 1 «Основы информационной безопасности» - учебный курс [1], статьи [2-5];

- глава 2 «Безопасность компьютерных систем. Традиционный подход к анализу проблем информационной безопасности» - работа [6];
- глава 3 «Объектно-ориентированный подход – перспективный принцип анализа вопросов информационной безопасности» - уч. курс [1];
- глава 4 «Основные определения и критерии классификации угроз» - уч. курс [1];
- главы 5 «Законодательный уровень информационной безопасности» - уч. курсы [1, 7];
- глава 6 «Стандарты и спецификации в области информационной безопасности» - уч. курсы [1, 7];
- глава 7 «Административный уровень информационной безопасности» - уч. курс [1];
- глава 8 «Процедурный уровень информационной безопасности» - уч. курс [1];
- глава 9 «Основные программно-технические меры обеспечения информационной безопасности» - уч. курс [1];
- глава 10 «Основные принципы криптографической зашиты информации» - работа [6];
- глава 11 «Асимметричные криптосистемы» - работа [6], уч. курс [8], статья [9];
- глава 12 «Симметричные криптосистемы» - работа [6], уч. курс [8];
- глава 13 «Вредоносные программы и компьютерные вирусы» – уч. курсы [10, 11];
- глава 14 «Основы борьбы с вредоносными программами» – уч. курсы [10, 11];
- глава 15 «Типовые удаленные атаки в глобальных компьютерных сетях» – работы [6, 12];
- глава 16 «Механизмы реализации удаленных атак в глобальной сети Internet» – работы [6, 12]
- глава 17 «Обеспечение безопасности систем входящих в состав глобальных компьютерных сетей» – работы [6, 12, 15], уч. курсы [8, 13, 14];
- глава 18 «Обеспечение безопасного взаимодействия в глобальных компьютерных сетях» – работы [6, 12, 15, 17, 18, 19, 58], уч. курсы [8, 13, 14, 16];
- глава 19 «Средства управления безопасностью в архитектуре операционных систем Windows» – уч. пособия [20, 21, 59], уч. курс [22];
- глава 20 «Обеспечение безопасности в операционных системах Windows 2000/XP и Windows 2003 server» - уч. пособие [59], уч. курс [13];

- глава 21 «Использование службы каталогов и групповых политик в Windows 2000/XP и Windows 2003 server» - уч. курс [13];
- глава 22 «Основы безопасности операционных систем семейства Unix» - уч. пособие [59], уч. курс [13];
- глава 23 «Безопасность программного обеспечения» - работа [23].

Таким образом, литература [1-23] составляет основную литературу по дисциплине и рекомендуется к изучению при освоении материала дисциплины. Также для более полного понимания процесоов представленных в части 5 целесообразно повторить дисциплину «Вычислительные системы, сети и телекоммуникации» материалы которой изложены в учебном пособии [58]. А для процессов представленных в части 6 - дисциплину «Операционные системы, среди и оболочки», материалы которой изложены в учебном пособии [59].

Для читателей, интересующихся отдельными вопросами информационной безопасности, автор рекомендует самостоятельно ознакомиться с указанными ниже работами.

Для знакомства с общими концептуальными вопросами информационной безопасности, общенаучными принципами построения систем обеспечения безопасности программных и аппаратных средств рекомендуются помимо работы [1] ознакомиться с работами [24-26, 38]. Особенно стоит отметить работы [26, 38] так как в них приводиться математический аппарат формализации несанкционированного доступа и средств защиты, в связи с чем данные работы могут быть полезны при проведении исследований, особенно [38].

Материалы по каналам утечки конфиденциальной информации, возможностям злоумышленников по несанкционированному доступу к различным системам обработки информации наиболее полно рассмотрены в работе [27], а также в работах [28-30, 32]. Стоит отметить работу [30] целиком посвященную вопросу защиты от прослушивания. Вопросам защиты от несанкционированного доступа по различным каналам информационных систем, а также фундаментальные принципы построения технических средств защиты изложены в работе [31]. Необходимо также рекомендовать работу [32] по сути, являющуюся фундаментальным исследованием зарубежного экспертного сообщества общих вопросов защиты информации при современном развитии технических средств и повсеместном внедрении глобальной сети Internet.

Достаточно долгое время информационная безопасность ассоциировалась с криптографической защитой информации. Поэтому вопросам криптографической защиты информации посвящено много работ. Однако фундаментальные работы в этой области, как правило, носят закрытый характер. Для интересующихся подробностями реализации алгоритмов шифрования, стоит рекомендовать работы [32, 33]. К менее всеобъемлющим, однако более доступным, стоит отнести работы [34, 35, 36], а также работу [37] по повышению скрытности передачи информации.

В связи с широким внедрением сетевых технологий объединения вычислительных систем существенно актуальной стала проблема обеспечения информационной безопасности в глобальных сетях. Актуальными работами по данному направлению являются публикации [12, 31, 32, 44]. Здесь интересен взгляд на данную проблему не только экспертов по безопасности, но злоумышленников. В данном плане интересны работы [39, 40], рассматривающие вопросы получения несанкционированного доступа со стороны злоумышленника, а так же работа [41] посвященная уязвимости беспроводных сетей связи.

В сети Internet находится множество порталов посвященных актуальным вопросам обеспечения антивирусной защиты. К ним в первую очередь относятся аналитические бюллетени разработчиков антивирусного программного обеспечения (например [43] или [44]), а также публикации в специализированных журналах и изданиях (например [42]).

По вопросам уязвимости и защиты программного обеспечения и операционных систем помимо работ [13, 20, 21, 22, 23], которые были использованы при подготовке пособия, рекомендуется изучить работы [32, 39, 40, 44, 45]. Особенно стоит отметить работу [45], представляющую собой фундаментальное исследование вопросов уязвимости программного кода и рекомендуемую специалистам в области программирования для углубленного изучения вопросов безопасности в своей области.

Также для самостоятельного изучения вопросов информационного безопасности рекомендуются для изучения следующие порталы в сети Internet: www.infosec.ru, bugtraq.ru, bozza.ru, bezpeka.com, www.citforum.ru, www.help-antivirus.ru, www.securitylab.ru.

Хотелось бы отметить, что в материалах глав 2-4 и 15-18 учебного пособия нашли отражение часть результатов научно-исследовательской работы автора по анализу математического аппарата обеспечения качества обслуживания в сетях радиосвязи [46]; исследованию качества фукционирования сетей радиосвязи в нестационарных условиях и при воздействии дестабилизурующих факторов (которые могут специально создаваться злоумышленниками) [47-51]; исследованию качества функционирования автоматизированных систем управления и обработки информации в условиях, когда подситема связи в их составе функционирует в нестационарном режиме [52-54]; а также исследованиям по управлению ресурсами подсистемы связи в интересах компенсации нестационарных условий и дестабилизурующих факторов при обеспечении функционировании автоматизированных систем управления и обработки информации [55-57].

Автор выражает благодарность рецензентам за кропотливый труд по поиску ошибок и неточностей, а также ценные замечания, которые помогли сделать материал пособия лучше и доступние.

Предложения и замечания по учебному пособию автор просит направлять на email: mak-serg@yandex.ru.

ЧАСТЬ 1. ОСНОВЫ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

1. ОСНОВЫ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

Под информационной безопасностью (ИБ) следует понимать защиту интересов субъектов информационных отношений. Ниже описаны основные ее составляющие – конфиденциальность, целостность, доступность. Приводится статистика нарушений ИБ, описываются наиболее характерные случаи.

1.1 Понятие информационной безопасности

*Под **информационной безопасностью** будем понимать защищенность информации и поддерживающей инфраструктуры от случайных или преднамеренных воздействий естественного или искусственного характера, которые могут нанести неприемлемый ущерб субъектам информационных отношений, в том числе владельцам и пользователям информации и поддерживающей инфраструктуры.*

***Защита информации** – это комплекс мероприятий, направленных на обеспечение информационной безопасности.*

Таким образом, правильный с методологической точки зрения подход к проблемам информационной безопасности начинается с выявления субъектов информационных отношений и интересов этих субъектов, связанных с использованием информационных систем (ИС).

Угрозы информационной безопасности – это оборотная сторона использования информационных технологий.

Из этого положения можно вывести два важных следствия:

1. Трактовка проблем, связанных с информационной безопасностью, для разных категорий субъектов может существенно различаться. Для иллюстрации достаточно сопоставить режимные государственные организации и учебные институты. В первом случае «пусть лучше все сломается, чем враг узнает хоть один секретный бит», во втором – «да нет у нас никаких секретов, лишь бы все работало».
2. Информационная безопасность не сводится исключительно к защите от несанкционированного доступа к информации, это принципиально более широкое понятие. Субъект информационных отношений может пострадать (понести убытки и/или получить моральный ущерб) не только от несанкционированного доступа, но и от поломки системы, вызвавшей перерыв в работе. Более того, для многих открытых организаций (например, учебных) собственно защита от несанкционированного доступа к информации стоит по важности отнюдь не на первом месте.

Отметим, что термин «**компьютерная безопасность**» (как эквивалент или заменитель понятия «информационная безопастность») представляется слишком узким. Компьютеры – только одна из составляющих информационных систем, и хотя в первую очередь внимание будет сосредоточено на информации, которая хранится, обрабатывается и передается с помощью компьютеров, ее безопасность определяется всей совокупностью составляющих и, в первую очередь, самым слабым звеном, которым в подавляющем большинстве случаев оказывается человек (записавший, например, свой пароль на листочке, прилепленном к монитору).

Согласно определению информационной безопасности, она зависит не только от компьютеров, но и от поддерживающей инфраструктуры, к которой можно отнести системы электро-, водо- и теплоснабжения, кондиционеры, средства коммуникаций и, конечно, обслуживающий персонал. Эта инфраструктура имеет самостоятельную ценность, однако нас будет интересовать лишь то, как она влияет на выполнение информационной системой предписанных ей функций.

Обратим внимание, что в определении ИБ перед существительным «ущерб» стоит прилагательное «неприемлемый». Очевидно, застраховаться от всех видов ущерба невозможно, тем более невозможно сделать это экономически целесообразным способом, когда стоимость защитных средств и мероприятий не превышает размер ожидаемого ущерба. Значит, с чем-то приходится мириться и защищаться следует только от того, с чем смириться никак нельзя. Иногда таким недопустимым ущербом является нанесение вреда здоровью людей или состоянию окружающей среды, но чаще порог неприемлемости имеет материальное (денежное) выражение, а целью защиты информации становится уменьшение размеров ущерба до допустимых значений.

1.2 Основные составляющие информационной безопасности

Спектр интересов субъектов, связанных с использованием информационных систем, можно разделить на следующие категории: обеспечение ***доступности***, ***целостности*** *и* ***конфиденциальности*** *информационных ресурсов и поддерживающей инфраструктуры.*

Поясним понятия доступности, целостности и конфиденциальности.

Доступность *– это возможность за приемлемое время получить требуемую информационную услугу.*

Целостность *- актуальность и непротиворечивость информации, ее защищенность от разрушения и несанкционированного изменения.*

Конфиденциальность *– это защита от несанкционированного доступа к информации.*

Информационные системы создаются (приобретаются) для получения определенных информационных услуг. Если по тем или иным причинам

предоставить эти услуги пользователям становится невозможно, это, очевидно, наносит ущерб всем субъектам информационных отношений. Поэтому, не противопоставляя **доступность** остальным аспектам, мы выделяем ее как важнейший элемент информационной безопасности.

Целостность можно подразделить на:

- **статическую,** понимаемую как неизменность информационных объектов;
- **динамическую,** относящуюся к корректному выполнению сложных действий. Средства контроля динамической целостности применяются, в частности, при анализе потока финансовых сообщений с целью выявления кражи, переупорядочения или дублирования отдельных сообщений.

Если вернуться к анализу интересов различных категорий субъектов информационных отношений, то почти для всех, кто реально использует ИС, на первом месте стоит доступность. Практически не уступает ей по важности целостность – какой смысл в информационной услуге, если она содержит искаженные сведения?

Наконец, конфиденциальные моменты есть также у многих организаций (даже в упоминавшихся выше учебных институтах стараются не разглашать сведения о личных данных сотрудников) и отдельных пользователей (например, пароли).

1.3 Важность и сложность проблемы информационной безопасности

Информационная безопасность является одним из важнейших аспектов интегральной безопасности, на каком бы уровне мы ни рассматривали последнюю – национальном, отраслевом, корпоративном или персональном. Для иллюстрации этого положения ограничимся несколькими нижеприведенными примерами [2-4].

- В Доктрине информационной безопасности Российской Федерации (здесь, подчеркнем, термин «информационная безопасность» используется в широком смысле) защита от несанкционированного доступа к информационным ресурсам, обеспечение безопасности информационных и телекоммуникационных систем выделены в качестве важных составляющих национальных интересов РФ в информационной сфере.
- По распоряжению президента США Клинтона (от 15 июля 1996 года, номер 13010) была создана Комиссия по защите критически важной инфраструктуры как от физических нападений, так и от атак, предпринятых с помощью информационного оружия. В начале октября 1997 года при подготовке доклада президенту глава вышеупомянутой комиссии Роберт Марш заявил, что в настоящее время ни правительство, ни частный сектор не располагают

средствами защиты от компьютерных атак, способных вывести из строя коммуникационные сети и сети энергоснабжения.

- Американский ракетный крейсер «Йорктаун» был вынужден вернуться в порт из-за многочисленных проблем с программным обеспечением, функционировавшим на платформе Windows NT 4.0 (Government Computer News, июль 1998). Таким оказался побочный эффект программы ВМФ США по максимально широкому использованию коммерческого программного обеспечения с целью снижения стоимости военной техники.
- Заместитель начальника управления по экономическим преступлениям Министерства внутренних дел России сообщил, что российские хакеры с 1994 по 1996 год предприняли почти 500 попыток проникновения в компьютерную сеть Центрального банка России. В 1995 году ими было похищено 250 миллиардов рублей (ИТАР-ТАСС, АР, 17 сентября 1996 года).
- Как сообщил журнал Internet Week от 23 марта 1998 года, потери крупнейших компаний, вызванные компьютерными вторжениями, продолжают увеличиваться, несмотря на рост затрат на средства обеспечения безопасности. Согласно результатам совместного исследования Института информационной безопасности и ФБР, в 1997 году ущерб от компьютерных преступлений достиг 136 миллионов долларов, что на 36% больше, чем в 1996 году. Каждое компьютерное преступление наносит ущерб примерно в 200 тысяч долларов.
- В середине июля 1996 года корпорация General Motors отозвала 292860 автомобилей марки Pontiac, Oldsmobile и Buick моделей 1996 и 1997 годов, поскольку ошибка в программном обеспечении двигателя могла привести к пожару.
- В феврале 2001 года двое бывших сотрудников компании Commerce One, воспользовавшись паролем администратора, удалили с сервера файлы, составлявшие крупный (на несколько миллионов долларов) проект для иностранного заказчика. К счастью, имелась резервная копия проекта, так что реальные потери ограничились расходами на следствие и средства защиты от подобных инцидентов в будущем. В августе 2002 года преступники предстали перед судом.
- Одна студентка потеряла стипендию в 18 тысяч долларов в Мичиганском университете из-за того, что ее соседка по комнате воспользовалась их общим системным входом и отправила от имени своей жертвы электронное письмо с отказом от стипендии.

К сожалению, современная технология программирования не позволяет создавать безошибочные программы, что не способствует быстрому развитию средств обеспечения ИБ. Следует исходить из того, что необходимо конструировать надежные системы (информационной безопасности) с привлечением ненадежных компонентов (программ). В

принципе, это возможно, но требует соблюдения определенных архитектурных принципов и контроля состояния защищенности на всем протяжении жизненного цикла ИС.

1.3.1 Наиболее опасные угрозы информационной безопасности

Большая часть угроз ИБ с которой пришлось столкнуться Российским организациям - это внутренние угрозы. Анализ исследований [2-4] позволяет обобщить информацию о состоянии вопросов ИБ в российских организациях.

Индекс опасности утечки внутренней информации в российских организациях на 50% опережает аналогичный показатель для любой из внешних угроз. Государственные структуры и представители частного сектора поставили на первое место утечку информации поскольку большая часть негативных последствий связанно с этим инцидентом: прямые финансовые убытки (46%), удар по репутации (42,3%) и потерю клиентов (36,9%). При этом организации начинают присматриваться к своим служащим все более пристально. Свыше 40% респондентов уже зафиксировали за 2006 год более одной утечки, а почти 20% — более пяти утечек.

Доля организаций, внедривших защиту от утечек, возросла за 2007 год на 500%, то есть в пять раз. Однако, пока лишь каждая десятая компания внедрила эффективное решение на основе информационных технологий (ИТ) реализующих элементы ИБ. Только девять из десяти планируют сделать это в ближайшие два-три года. Таким образом, есть все основания полагать, что проникновение систем защиты от утечек на российский рынок продолжится и дальше, причем затронет абсолютно все отрасли экономики.

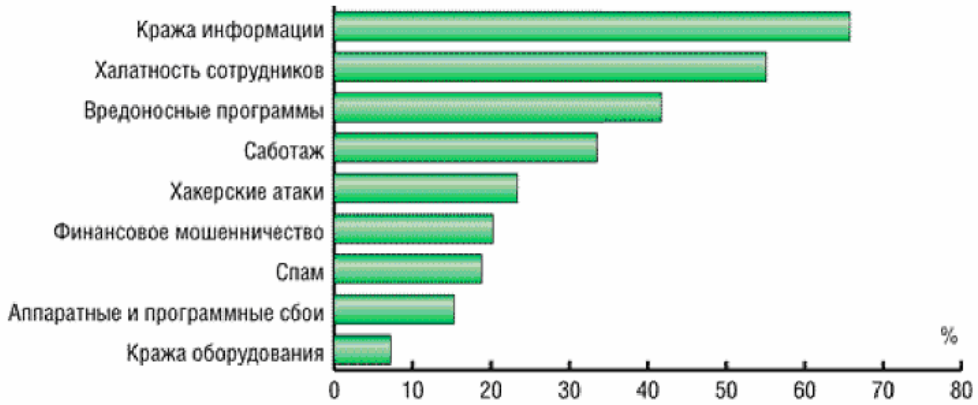


Рис. 1.1 - Угрозы ИБ по оценкам за 2007 год

Спектр самых опасных угроз ИБ по оценкам за 2007 выглядит следующим образом:

- На первом месте **кража информации** (65,8%).
- на втором месте оказалась **халатность сотрудников** (55,1%).

- **вирусные атаки** заняли третье место, набрав 41,7% голосов респондентов.
- На четвертом месте оказалась угроза **саботажа** (33,5%). Судя по всему, можно утверждать, что высокий рейтинг опасности саботажа обусловлен тем, что респонденты постепенно теряют чувство страха перед внешними угрозами.
- **хакерские атаки** занимают пятое место с 23,4% голосов.

Если разделить угрозы на внутренние и внешние, то можно увидеть, что внутренние угрозы превалируют над вирусами, хакерами и спамом. Для построения соответствующей диаграммы (рис. 1.2) в категорию внутренних угроз были отнесены халатность сотрудников, саботаж и финансовое мошенничество, а в категорию внешних угроз — вирусы, хакеры и спам.

Рис. 1.2 - Соотношение опасности внутренних и внешних угроз ИБ

Необходимо отметить, что угрозы кражи информации, различных сбоев и кражи оборудования не были отнесены ни к одной из групп. Так как они могут быть реализованы как изнутри, так и снаружи, либо вообще без вмешательства человека (например, аппаратные сбои).

Таким образом, респонденты гораздо больше обеспокоены внутренней ИБ, чем защитой от внешних угроз. Кроме того, следует учитывать, что неклассифицированные риски, например кражу информации или оборудования, чаще всего относят к внутренним угрозам. То есть внешние риски заметно уступают внутренним угрозам.

1.3.2 Внутренние угрозы информационной безопасности

Определив, что самые опасные угрозы ИБ исходят изнутри организации, необходимо изучить структуру внутренних рисков. Как показали результаты исследований [2-4] (респондентов просили оценить угрозы внутренней ИБ являются наиболее опасными для опрашиваемых) — рис. 1.3, в списке самых опасных внутренних угроз с огромным отрывом лидирует:

1. нарушение конфиденциальности информации (70,1%);

2. искажение информации (38,4%) — отстает на целых 31,7%. Другими словами, риск утечки ценной информации волнует респондентов почти в два раза больше, чем любая другая инсайдерская угроза.

Рис. 1.3. Наиболее опасные угрозы внутренней ИБ

В 2006 году были зафиксированы пять крупных утечек, в России и СНГ, а также почти полторы сотни инцидентов внутренней ИБ в других частях света (таблица 1.1).

Таблица 1.1 - Самые крупные утечки 2006 года в России и СНГ

Дата	**Организация**	**Потенциальный ущерб**
Август, 2006 год	Российские банки, занимающиеся потребительским кредитованием	Удар по репутации и серьезный подрыв доверия к отечественному финансовому сектору
Август, 2006 год	Банк «Первое ОВК» (поглощен Росбанком в 2005 году)	Ухудшение имиджа, плохое паблисити, массовый отток клиентов
Сентябрь, 2006 год	МЦС (Мобильная цифровая связь), владелец марки Velcom	Удар по репутации, потеря лояльных клиентов и трудности с привлечением новых
Октябрь, 2006 год	«Вэб Хостинг» (владелец марки Valuehost)	Массовый отток клиентов, юридические издержки, удар по имиджу
Декабрь, 2006 год	«Русский стандарт», ХКФ-банк, Росбанк, Финансбанк, Импэксбанк и др.	Ухудшение репутации всего банковского сектора

Таким образом, с точки зрения респондентов, наиболее опасной угрозой ИБ является утечка конфиденциальной информации, совершаемая

инсайдерами. В результате такой утечки, более всего респонденты были озабочены следующими последствиями (рис. 1.4):

1. прямыми финансовыми убытками (46%);
2. ухудшение имиджа и общественного мнения (42,3%);
3. потеря клиентов (36,9%).

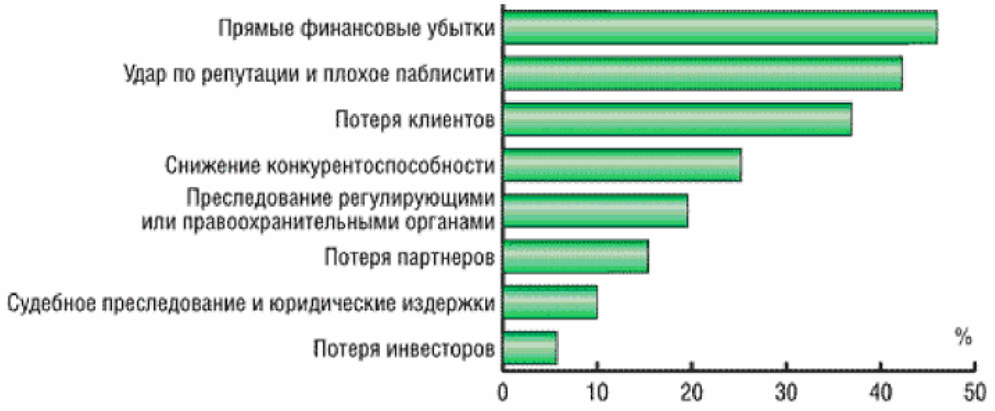


Рис. 1.4 - Наиболее существенные последствия утечки конфиденциальных данных

Кроме того, респонденты озабочены снижением конкурентоспособности (25,2%) организации, что является скорее следствием целого ряда других негативных последствий утечки. Между тем лишь каждый десятый (10%) упомянул среди наиболее плачевных последствий юридические издержки и судебное преследование, что свидетельствует о неразвитости правоприменительной практики в России.

Результаты анализа [2-4] по выявлению самых распространенных каналов утечки информации представлено на рис. 1.5.

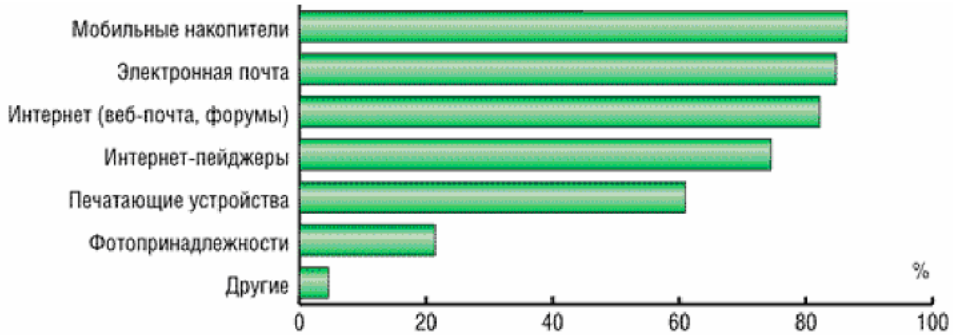


Рис. 1.5 - Каналы утечки конфиденциальных данных

Заметим, что наибольшей популярностью среди инсайдеров пользуются:

1. мобильные накопители (86,6%),
2. электронная почта (84,8%),
3. web-браузеры (82,2%),
4. ICQ клиенты (78 %).

Отметим что, крайние три позиции соответствуют использованию Internet приложений, в случае если в организации не используется система контроля доступа и обмена информацией с глобальной сетью Internet.

Исследование [5] позволило сформировать общую статистическую картину типовых уязвимостей при использовании организациями общего подключения к сети Internet (таблица 1.2). При этом уровень опасности данных угроз различен, а эффективность использования средств защиты от угроз существенно отличается и зачастую низко эффективно (рис. 1.6).

Таблица 1.2 - Безопасность использования организациями сети Internet

№	Параметр / среднее значение на одного сотрудника за неделю	Все организации	Банки
1	**Наличие потенциально опасного контента** (активных объектов) в инспектируемом входящем и исходящем Интернет-трафике	1,070	845
2	**Посещение** пользователями **опасных** и запрещенных **Web-сайтов**	639	472
3	**Идентифицированные угрозы**, инциденты, успешные атаки, действие spyware (шпионов), троянов, эксплойтов	308	127
4	**Нецелевое использование Интернет** и других ресурсов (Skype, IM, P2P, скачивание музыки и видео, потоковое видео/аудио и др.). Нарушения политик ИБ.	128	21
5	**Использование неавторизованных**, потенциально опасных **приложений**	5	1.7
6	**Скрытые каналы утечки** - неавторизованные и потенциально опасные каналы коммуникаций пользовательских компьютеров с внешними узлами (в т.ч. дистанционное управление компьютерами извне, использование их для массовой рассылки спама и/или распределенных DDoS атак)	3.6	1.8
7	**Передача/получение** через Интернет опасных, подозрительных и инфицированных **файлов**	5	0.02

Уровень опасности выявленных Интернет-угроз

Эффективность используемых средств защиты и работы служб ИБ

Рис. 1.6

Таким образом, основной угрозой ИБ, реализация которой приводит к максимальным последствиям является разглашение конфиденциальной информации. Анализ, проведенный выше показывает, что данные, представляющие собой коммерческую тайну, могут покинуть сетевой периметр предприятия несколькими путями: по электронной почте, через чаты, форумы и другие службы Интернета, с помощью средств мгновенного обмена сообщениями, копирования информации на мобильные носители, а также посредством распечатки ее на принтере. Таким образом, для обнаружения факта разглашения конфиденциальной информации необходимо контролировать все пути утечки данных, в частности анализировать исходящий трафик, передаваемый по протоколам SMTP, HTTP, FTP и TCP/IP.

Еще одной серьезной угрозой, требующей анализа входящего/исходящего трафика организации, является **нецелевое использование ресурсов компании**. Сюда входят:

- посещение сайтов общей и развлекательной направленности (не имеющих отношения к исполнению служебных обязанностей) в рабочее время;
- загрузка, хранение и использование мультимедиафайлов и ПО развлекательной направленности в рабочее время;
- использование ресурсов компании для рассылки информации рекламного характера, спама или информации личного характера, включая информацию о сотрудниках, номера кредитных карт и т.д.

Соответственно для борьбы с угрозами подобного типа необходимы технические решения, фильтрующие исходящие веб-запросы и почтовый трафик.

1.3.3 Средства защиты

Для борьбы с разными типами внутренних угроз информационной безопасности используют различные технические средства. Но только комплексное решение поможет действительно решить проблему защиты компьютерной инфраструктуры предприятия.

Анализ [3-5] показывает, что в сфере распространения средств ИБ среди организаций за последний год не произошло сколько-нибудь значительных изменений (рис. 1.7).

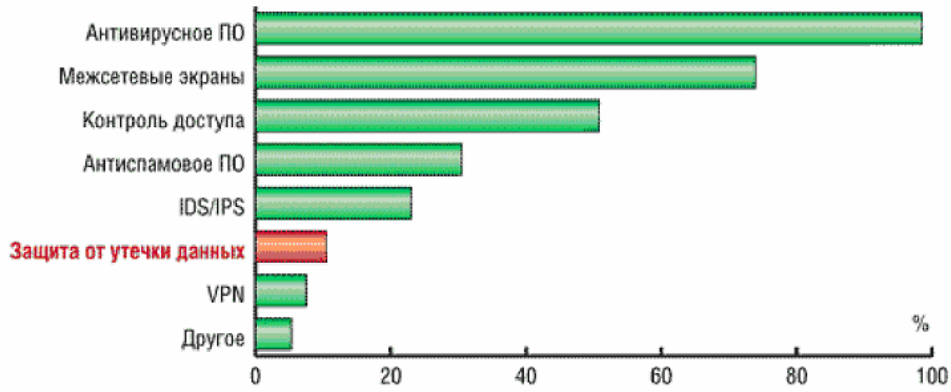


Рис. 1.7 - Средства обеспечения ИБ применяемые организациями

Основными средствами, которые используют современные компании для обеспечения ИБ являются:

- антивирусы (98,6%),
- межсетевые экраны (73,9%),
- средства контроля доступа (50,8%).

В качестве наиболее эффективных путей защиты от утечек и решениях, которые представляются организациям наиболее адекватными и приемлемыми для решения проблемы внутренней ИБ, но по ряду причин не используемых респондентами на практике респонденты назвали ряд мер популярность которых изображена на рис. 1.8.

Рис. 1.8 - Наиболее эффективные средства обеспечения ИБ

Как отмечено в исследованиях [2-5] наиболее эффективным средством зашиты от угроз ИБ являются комплексные информационные продукты (44,8%). Далее следуют организационные меры (25,3%), тренинги персонала (21,6%) и ограничение связи с внешними сетями (18,1%). При этом респонденты отметили, что в соответствии с разработанными планами обеспечения ИБ респонденты планирую у себя внедрить ту или иную систему обеспечения ИБ. Согласно распределению ответов (рис. 1.9), девять из десяти (89,9%) организаций планируют внедрить в ближайшие три года ту или иную систему защиту от утечек.

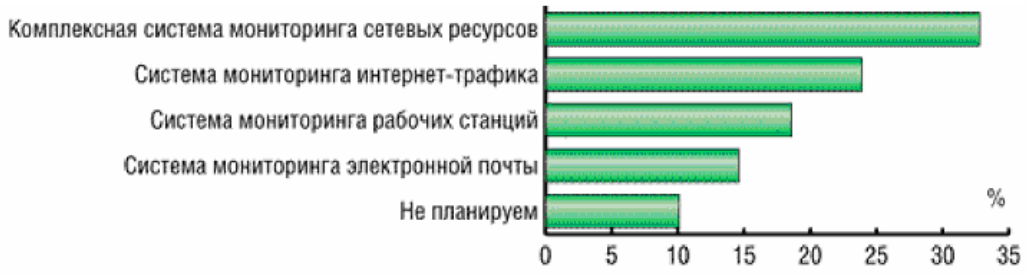


Рис. 1.9 - Планы по внедрению систем защиты ИБ

Как видно из ответов респондентов комплексные решения ИТ, а именно комплексные системы мониторинга сетевых ресурсов превалируют при выработке решения о выборе средства обеспечения ИБ, при этом они же являются наиболее эффективными средствами обеспечения ИБ (рис. 1.8).

Таким образом, российские компании постепенно в полной мере осознают угрозы и риски от халатного отношения к вопросам информационной безопасности и начинают проводить активную политику внедрения комплексных средств ИБ с информационные системы сопровождения бизнеса.

1.4 Сценарии реализации угроз информационной безопасности

Существует целый ряд сценариев, согласно которым могут быть реализованы внутренние угрозы. Каждый из этих сценариев учитывает конкретную цель неправомерных действий и технические средства ее достижения.

1.4.1 Разглашение конфиденциальной информации

Данный вид угроз подразумевает разглашение информации, представляющей коммерческую тайну, посредством отсылки ее по электронной почте, через Интернет (чат, форум и т. д.), с помощью средств обмена мгновенными сообщениями, путем копирования информации на переносные носители или распечатки данных. Для обнаружения факта разглашения конфиденциальной информации разные компании предлагают перехват почтового, внутрисетевого (на уровне TCP/IP) и Web-трафика (протоколы HTTP, FTP, IM и т. д.) с последующим анализом различными методами фильтрации, в том числе с помощью анализа контента.

1.4.2 Обход средств защиты от разглашения конфиденциальной информации

Обход средств защиты от разглашения конфиденциальной информации удается добиться при помощи различных преобразований данных, например, шифрования, архивации с большой глубиной вложенности, преобразования в графический формат или редкие текстовые форматы, изменения кодировки,

использования неизвестного ПО или общения на иностранном языке, незнакомом большинству сотрудников компании. Системы защиты от утечек конфиденциальных данных борются с такими действиями, поддерживая разнообразные форматы файлов и реагируя на любые подозрительные действия (неизвестный формат файла, шифрование и т. д.).

1.4.3 Кража конфиденциальной информации

Неправомерный доступ к информации возможен при ее передаче через обычное соединение (без шифрования) — путем взлома корпоративной сети; если данные размещены в местах, доступных посторонним людям или сотрудникам, не имеющим соответствующих прав; если посторонний человек имеет возможность читать с экрана монитора; если имеется доступ к напечатанным материалам, переносным носителям или компьютерам.

1.4.4 Нарушение авторских прав на информацию

В рамках угрозы нарушения авторских прав, возможно копирование частей документов одного автора в документы другого автора (а также в почтовые сообщения, Web-формы и т. д.); индивидуальное шифрование документов, при котором компания лишается возможности работать с документом после увольнения или перевода сотрудника или в случае утраты пароля; использование опубликованных в Интернете материалов без обработки в своих документах; использование мультимедиа-файлов (графики, аудио- и видеозаписей), ПО и прочих информационных объектов, защищенных авторским правом; подделка данных адресата или отправителя с целью опорочить его доброе имя или скомпрометировать компанию.

1.4.5 Нецелевое использование ресурсов

Под нецелевым использованием ресурсов подразумеваются посещение сайтов общей и развлекательной направленности (не имеющих отношения к исполнению служебных обязанностей) в рабочее время; загрузка, хранение и использование мультимедиа-файлов и ПО развлекательной направленности в рабочее время; использование ненормативной, грубой, некорректной лексики при ведении деловой переписки; загрузка, просмотр и распространение материалов для взрослых, а также материалов, содержащих нацистскую символику, агитацию или другие противозаконные материалы; использование ресурсов компании для рассылки информации рекламного характера, спама или информации личного характера, в том числе информации о сотрудниках, номерах социального страхования, кредитных карт и т. д.

Следует отметить, что не существует ни одного комплексного решения, которое позволяло бы полностью защитить компанию от воздействия этих и других угроз. Однако многие поставщики предлагают

различные варианты решений, позволяющих контролировать и предотвращать реализацию значительной части угроз. Так, продукты компаний ClearSwift (http://www.clearswift.com), Cobion (http://www.cobion.com) и SurfControl (http://www.surfcontrol.com) позволяют исключить нецелевое использование ИТ-ресурсов компании.

Для фильтрации Web-трафика используется база данных URL, хранящая миллионы классифицированных по темам записей. Когда сотрудник компании запрашивает в браузере какую-нибудь страницу, специальный фильтр проверяет записи базы и определяет категорию запрашиваемого ресурса. Список категорий насчитывает несколько десятков, туда входят почтовые, развлекательные, туристические, порнографические и другие группы ресурсов. Если информации о требуемой Web-странице еще нет в базе, то проводится анализ содержимого страницы и определение ее тематики в автоматическом режиме. Такой подход позволяет администратору строить политики на основании групп пользователей и их прав на доступ к страницам соответствующих категорий.

Чтобы исключить нецелевое использование почтовой службы компании, потребуется контентный анализ каждого письма. Современные решения позволяют анализировать не только формальные атрибуты сообщения, но и вложения различных форматов, графическое и текстовое содержимое тела письма, а также все содержащиеся в нем ссылки.

Решения, позволяющие исключить нецелевое использование ИТ-ресурсов компании, часто содержат некоторые возможности предотвращения утечки конфиденциальных данных. Например, продукты Cobion и SurfControl фильтруют исходящий почтовый трафик еще и на предмет содержания в нем конфиденциальной информации. Чтобы настроить такой анализ, администратору необходимо вручную указать в консоли управления те документы, утечку которых необходимо предотвратить.

Важно отметить, что задача предотвращения утечки конфиденциальных данных не решена в этих продуктах полностью. Так, без всякого присмотра остается рабочая станция сотрудника, с которой он может переписать файл на мобильный носитель или просто его распечатать. Особый контроль требуется и для Web-трафика. В Интернете есть множество чатов, форумов, открытых почтовых сервисов и т. д. Все эти ресурсы могут быть использованы для разглашения конфиденциальной информации.

Рынок систем, специализирующихся только на защите от утечек конфиденциальных данных (Anti-Leakage Software), только начинает формироваться. Среди существующих решений пока лишь InfoWatch Enterprise Solution компании InfoWatch (http://www.infowatch.ru) обеспечивает контроль над всеми видами коммуникаций в ИТ-инфраструктуре компании. Данное решение предусматривает средства анализа почтового и Web-трафика, а также механизмы контроля за рабочими станциями служащих.

2. БЕЗОПАСНОСТЬ КОМПЬЮТЕРНЫХ СИСТЕМ. ТРАДИЦИОННЫЙ ПОДХОД К АНАЛИЗУ ПРОБЛЕМ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

2.1 Актуальность задач компьютерной безопасности

По мере развития и усложнения средств, методов и форм автоматизации процессов обработки информации повышается зависимость общества от степени безопасности используемых им информационных технологий, от которых порой зависит благополучие, а иногда и жизнь многих людей.

Актуальность и важность проблемы обеспечения безопасности информационных технологий обусловлены следующими причинами:

- *резкое увеличение вычислительной мощности современных компьютеров при одновременном упрощении их эксплуатации;*
- *резкое увеличение объемов информации, накапливаемой, хранимой и обрабатываемой с помощью компьютеров и других средств автоматизации;*
- *сосредоточение в единых базах данных информации различного назначения и различной принадлежности;*
- *высокие темпы роста парка персональных компьютеров, находящихся в эксплуатации в самых разных сферах деятельности;*
- *резкое расширение круга пользователей, имеющих непосредственный доступ к вычислительным ресурсам и массивам данных;*
- *бурное развитие программных средств, не удовлетворяющих даже минимальным требованиям безопасности;*
- *повсеместное распространение сетевых технологий и объединение локальных сетей в глобальные;*
- *развитие глобальной сети Internet, практически не препятствующей нарушениям безопасности систем обработки информации во всем мире.*

2.2 Основные понятия информационной безопасности автоматизированных систем обработки информации

Безопасность автоматизированной системы обработки информации (АСОИ) *– свойство защищенности системы от случайного или преднамеренного вмешательства в нормальный процесс ее функционирования, а также от попыток хищения, изменения или разрушения ее компонентов.*

Природа воздействий на АСОИ может быть самой разнообразной. Это и стихийные бедствия (землетрясение, ураган, пожар), и выход из строя

составных элементов АСОИ, и ошибки персонала, и попытка проникновения злоумышленника.

Безопасность АСОИ достигается *принятием мер по обеспечению конфиденциальности и целостности обрабатываемой ею информации, а также доступности и целостности компонентов и ресурсов системы.*

Под доступом к информации понимается *ознакомление с информацией, ее обработка, в частности копирование, модификация или уничтожение информации.*

Различают следующие виды доступа к информации:

- ***санкционированный доступ*** *- доступ к информации, не нарушающий установленные правила разграничения доступа;*
- ***несанкционированный доступ*** **(*НСД*)** *- характеризуется нарушением установленных правил разграничения доступа*. Лицо или процесс, осуществляющие несанкционированный доступ к информации, являются нарушителями правил разграничения доступа. Несанкционированный доступ является наиболее распространенным видом компьютерных нарушений.

Правила разграничения доступа *служат для регламентации права доступа субъектов доступа к объектам доступа.*

Конфиденциальность данных *- это статус, предоставленный данным и определяющий требуемую степень их защиты. По существу - это свойство информации быть известной только допущенным и прошедшим проверку* (авторизированным) субъектам системы (пользователям, процессам, программам). Для остальных субъектов системы эта информация должна быть неизвестной.

Субъект *- это активный компонент системы, который может стать причиной потока информации от объекта к субъекту или изменения состояния системы.*

Объект *- пассивный компонент системы, хранящий, принимающий или передающий информацию. Доступ к объекту означает доступ к содержащейся в нем информации.*

Целостность информации обеспечивается в том случае, если данные в системе не отличаются в семантическом отношении отданных в исходных документах, т.е. если не произошло их случайного или преднамеренного искажения или разрушения.

Целостность компонента или ресурса системы *— это свойство компонента или ресурса быть неизменными в семантическом смысле при функционировании системы в условиях случайных или преднамеренных искажений или разрушающих воздействий.*

Доступность компонента или ресурса системы *- это свойство компонента или ресурса быть доступным для авторизованных законных субъектов системы.*

Под угрозой безопасности АСОИ понимаются *возможные воздействия на АСОИ, которые прямо или косвенно могут нанести ущерб ее безопасности.*

Ущерб безопасности *подразумевает нарушение состояния защищенности информации, содержащейся и обрабатывающейся в АСОИ.* С понятием угрозы безопасности тесно связано понятие уязвимости АСОИ.

Уязвимость АСОИ *- это некоторое неудачное свойство системы, которое делает возможным возникновение и реализацию угрозы.*

Атака на компьютерную систему *— это действие, предпринимаемое злоумышленником, которое заключается в поиске и использовании той или иной уязвимости системы*. Таким образом. атака — это одна из реализаций угрозы безопасности.

Противодействие угрозам безопасности является целью защиты систем обработки информации.

Безопасная или защищенная система *- это система со средствами защиты, которые успешно и эффективно противостоят угрозам безопасности.*

Комплекс средств защиты *- программных и технических средств, создаваемых и поддерживаемых для обеспечения информационной безопасности АСОИ. Комплекс создается и поддерживается в соответствии с принятой в данной организации политикой безопасности.*

Политика безопасности *- это совокупность норм, правил и практических рекомендаций, регламентирующих работу средств защиты АСОИ от заданного множества угроз безопасности.*

2.3 Понятие «угрозы». Основные угрозы безопасности систем обработки информации

По цели воздействия различают три основных типа угроз безопасности АСОИ:

- *угрозы нарушения конфиденциальности информации;*
- *угрозы нарушения целостности информации;*
- *угрозы нарушения работоспособности системы (отказы в обслуживании).*

Угрозы нарушения конфиденциальности *направлены на разглашение конфиденциальной или секретной информации. При реализации этих угроз информация становится известной лицам, которые не должны иметь к ней доступ.* В терминах компьютерной безопасности угроза нарушения конфиденциальности имеет место всякий раз, когда получен несанкционированный доступ к некоторой закрытой информации, хранящейся в компьютерной системе или передаваемой от одной системы к другой.

***Угрозы нарушения целостности информации**, хранящейся в компьютерной системе или передаваемой по каналу связи, направлены на ее изменение или искажение, приводящее к нарушению ее качества или полному уничтожению.* Целостность информации может быть нарушена умышленно злоумышленником, а также в результате объективных воздействий со стороны среды, окружающей систему. Эта угроза особенно актуальна для систем передачи информации - компьютерных сетей и систем телекоммуникаций. Умышленные нарушения целостности информации не следует путать с ее санкционированным изменением, которое выполняется полномочными лицами с обоснованной целью (например, таким изменением является периодическая коррекция некоторой базы данных).

***Угрозы нарушения работоспособности** (отказ в обслуживании) направлены на создание таких ситуаций, когда определенные преднамеренные действия либо снижают работоспособность АСОИ, либо блокируют доступ к некоторым ее ресурсам.* Например, если один пользователь системы запрашивает доступ к некоторой службе, а другой предпринимает действия по блокированию этого доступа, то первый пользователь получает отказ в обслуживании. Блокирование доступа к ресурсу может быть постоянным или временным.

Современная автоматизированная система обработки информации представляет собой сложную систему, состоящую из большого числа компонентов различной степени автономности, которые связаны между собой и обмениваются данными.

АСОИ состоит из следующих компонент:

- *аппаратные средства* — ЭВМ и их составные части (процессоры, мониторы, терминалы, периферийные устройства-дисководы, принтеры, контроллеры, кабели, линии связи) и т.д.;
- *программное обеспечение* — приобретенные программы, исходные, объектные, загрузочные модули; операционные системы и системные программы (компиляторы, компоновщики и др.), утилиты, диагностические программы и т.д.;
- *данные* — хранимые временно и постоянно, на магнитных носителях, печатные, архивы, системные журналы и т.д.;
- *персонал* — обслуживающий персонал и пользователи.

Практически каждый компонент может подвергнуться внешнему воздействию или выйти из строя.

Опасные воздействия на АСОИ можно подразделить на:

- *случайные;*
- *преднамеренные.*

Случайные воздействия. Анализ опыта проектирований, изготовления и эксплуатации АСОИ показывает, что информация подвергается различным

случайным воздействиям на всех этапах цикла жизни и функционирования АСОИ.

Причинами случайных воздействий при эксплуатации АСОИ могут быть:

- *аварийные ситуации из-за стихийных бедствий и отключений электропитания;*
- *отказы и сбои аппаратуры;*
- *ошибки в программном обеспечении;*
- *ошибки в работе обслуживающего персонала и пользователей;*
- *помехи в линиях связи из-за воздействий внешней среды.*

Преднамеренные угрозы *связаны с целенаправленными действиями нарушителя.* В качестве нарушителя могут выступать служащий, посетитель, конкурент, наемник и т.д. Действия нарушителя могут быть обусловлены разными мотивами: недовольством служащего своей карьерой, сугубо материальным интересом (взятка), любопытством, конкурентной борьбой, стремлением самоутвердиться любой ценой и т. п.

Исходя из возможности возникновения наиболее опасной ситуации, обусловленной действиями нарушителя, можно составить ***гипотетическую модель потенциального нарушителя:***

1. *квалификация нарушителя может быть на уровне разработчика данной системы;*
2. *нарушителем может быть как постороннее лицо, так и законный пользователь системы;*
3. *нарушителю известна информация о принципах работы системы;*
4. *нарушитель выберет наиболее слабое звено в защите.*

В частности, для банковских АСОИ можно выделить следующие преднамеренные угрозы:

1. несанкционированный доступ посторонних лиц, не принадлежащих к числу банковских служащих, и ознакомление с хранимой конфиденциальной информацией;
2. ознакомление банковских служащих с информацией, к которой они не должны иметь доступ;
3. несанкционированное копирование программ и данных;
4. кража магнитных носителей, содержащих конфиденциальную информацию;
5. кража распечатанных банковских документов;
6. умышленное уничтожение информации;
7. несанкционированная модификация банковскими служащими финансовых документов, отчетности и баз данных;
8. фальсификация сообщений, передаваемых по каналам связи;
9. отказ от авторства сообщения, переданного по каналам связи;
10. отказ от факта получения информации;
11. навязывание ранее переданного сообщения;
12. разрушение информации, вызванное вирусными воздействиями;

13. разрушение архивной банковской информации, хранящейся на магнитных носителях;
14. кража оборудования.

В таблице 2.1 показаны основные пути реализации угроз безопасности АСОИ при воздействии на ее компоненты. Конечно, таблица 2.1 дает самую общую картину того, что может произойти с системой. Конкретные обстоятельства и особенности должны рассматриваться отдельно.

Таблица 2.1 - Основные пути реализации угроз безопасности АСОИ при воздействии на ее компоненты

Объекты воздействия	***Нарушение конфиденциальности информации***	***Нерушение целостности информации***	***Нарушение работоспособности системы***
Аппаратные средства	*НСД-подключение; использование ресурсов; хищение носителей.*	*НСД-подключение; использование ресурсов; модификация, изменение режимов*	*НСД-изменение режимов; вывод из строя; разрушение*
Программное обеспечение	*НСД-копирование; хищение; перехват.*	*НСД, внедрение «троянского коня», «вирусов». «червей»*	*НСД-искажение; удаление; подмена*
Данные	*НСД-копирование; хищение; перехват*	*НСД- искажение; модификация*	*НСД - искажение; удаление: подмена*
Персонал	*Разглашение: передача сведений о защите; халатность.*	*«Маскарад»; вербовка; подкуп персонала*	*Уход с рабочего места; физическое устранение*

Термин «вирус» в применении к компьютерам был предложен Фредом Коэном из Университета Южной Калифорнии. Исторически первое определение, котрое дал Ф. Коэн: «***Компьютерный вирус*** *- это программа, которая может заражать другие программы, модифицируя их посредством включения в них своей, возможно, измененной копии, причем последняя сохраняет способность к дальнейшему размножению»*. Ключевыми понятиями в определении компьютерного вируса являются способность вируса к саморазмножению и способность к модификации вычислительного процесса. Указанные свойства компьютерного вируса аналогичны паразитиоованию в живой природе биологического вируса.

Компьютерный вирус пытается тайно записать себя на компьютерные диски. Способ функционирования большинства вирусов заключается в таком изменении системных файлов компьютера, чтобы вирус начинал свою деятельность при каждой загрузке. Например, вирусы, поражающие загрузочный сектор, пытаются инфицировать часть дискеты или жесткого диска, зарезервированную. только для операционной системы и хранения файлов запуска. Эти вирусы особенно коварны, так как они загружаются в память при каждом включении компьютера. Такие вирусы обладают

наибольшей способностью к размножению и могут постоянно распространяться на новые диски.

Сетевой «червь» *представляет собой разновидность программы-вируса, которая распространяется по глобальной сети и не оставляет своей копии на магнитном носителе.* Термин «червь» пришел из научно-фантастического романа Джона Бруннера «По бурным волнам». Этот термин используется для именования программ, которые подобно ленточным червям перемещаются по компьютерной сети от одной системы к другой.

Первоначально «черви» были разработаны для поиска в сети других компьютеров со свободными ресурсами, чтобы получить возможность выполнить распределенные вычисления. При правильном использовании технология «червей» может быть весьма полезной. Например, «червь» World Wide Web Worm формирует индекс поиска участков Web. Однако «червь» легко превращается во вредоносную программу. «Червь» использует механизмы поддержки сети для определения узла, который может быть поражен. Затем с помощью этих же механизмов передает свое тело в этот узел и либо активизируется, либо ждет походящих условий для активизации.

2.4. Понятие несанкционированного доступа

Несанкционированный доступ (НСД) *состоит в получении пользователем (нарушителем) доступа к объекту в нарушение правил разграничения доступа, установленных в соответствии с принятой в организации политикой безопасности.* НСД является наиболее распространенным и многообразным видом компьютерных нарушений. НСД использует любую ошибку в системе зашиты и возможен при нерациональном выборе средств зашиты, их некорректной установке и настройке. НСД может быть осуществлен как штатными средствами АСОИ. так и специально созданными аппаратными и программными средствами.

Перечислим основные каналы несанкционированного доступа, через которые нарушитель может получить доступ к компонентам АСОИ и осуществить хищение, модификацию и/или разрушение информации:

- *все штатные каналы доступа к информации (терминалы пользователей, оператора, администратора системы; средства отображения и документирования информации; каналы связи) при их использовании нарушителями, а также законными пользователями вне пределов их полномочий;*
- *технологические пульты управления;*
- *линии связи между аппаратными средствами АСОИ;*
- *побочные электромагнитные излучения от аппаратуры, линий связи, сетей электропитания и заземления и др.*

Из всего разнообразия способов и приемов несанкционированного доступа ***наиболее распространенными нарушениями являются:***

- *перехват паролей;*
- *«маскарад»;*
- *незаконное использование привилегий.*

Перехват паролей осуществляется специально разработанными программами. При попытке законного пользователя войти в систему программа-перехватчик имитирует на экране дисплея ввод имени и пароля пользователя, которые сразу пересылаются владельцу программы-перехватчика, после чего на экран выводится сообщение об ошибке и управление возвращается операционной системе. Пользователь предполагает, что допустил ошибку при вводе пароля. Он повторяет ввод и получает доступ в систему. Владелец программы-перехватчика, получивший имя и пароль законного пользователя, может теперь использовать их в своих целях. Существуют и другие способы перехвата паролей.

«Маскарад» *- это выполнение каких-либо действий одним пользователем от имени другого пользователя, обладающего соответствующими полномочиями.* Целью «маскарада» является приписывание каких-либо действий другому пользователю либо присвоение полномочий или привилегий другого пользователя.

Примерами реализации «маскарада» являются:

- *вход в систему под именем и паролем другого пользователя (этому «маскараду» предшествует перехват пароля);*
- *передача сообщений в сети от имени другого пользователя.*

Незаконное использование привилегий. Большинство систем защиты устанавливают определенные наборы привилегий для выполнения заданных функций. Каждый пользователь получает свой набор привилегий: обычные пользователи — минимальный, администраторы — максимальный. Несанкционированный захват привилегий, например посредством «маскарада», приводит к возможности выполнения нарушителем определенных действий в обход системы защиты. Следует отметить, что незаконный захват привилегий возможен либо при наличии ошибок в системе защиты, либо из-за халатности администратора при управлении системой и назначении привилегий

Угрозы, компьютерных сетей.

Особо следует остановиться на угрозах, которым могут подвергаться компьютерные сети. Основная особенность любой компьютерной сети состоит в том, что ее компоненты распределены в пространстве. Связь между узлами (объектами) сети осуществляется физически с помощью сетевых линий связи и программно с помощью механизма сообщений. При этом управляющие сообщения и данные, пересылаемые между объектами сети, передаются в виде пакетов обмена.

При вторжении в компьютерную сеть злоумышленник может использовать как пассивные, так и активные методы вторжения.

Пассивное вторжение (перехват информации) *- нарушитель только наблюдает за прохождением информации по каналу связи, не вторгаясь ни в информационный поток, ни в содержание передаваемой информации.* Как правило, злоумышленник может определить пункты назначения и идентификаторы либо только факт прохождения сообщения, его длину и частоту обмена, если содержимое сообщения не распознаваемо, т.е. выполнить анализ трафика (потока сообщений) в данном канале.

Активное вторжение *- нарушитель стремится подменить информацию, передаваемую в сообщении. Он может выборочно модифицировать, изменить или добавить правильное или ложное сообщение, удалить, задержать или изменить порядок следования сообщений.* Злоумышленник может также аннулировать и задержать все сообщения, передаваемые по каналу. Подобные действия можно квалифицировать как отказ в передаче сообщений.

3. ОБЪЕКТНО-ОРИЕНТИРОВАННЫЙ ПОДХОД – ПЕРСПЕКТИВНЫЙ ПРИНЦИП АНАЛИЗА ВОПРОСОВ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

3.1 Необходимость применения объектно-ориентированного подхода к информационной безопасности

В настоящее время информационная безопасность является относительно замкнутой дисциплиной, развитие которой не всегда синхронизировано с изменениями в других областях информационных технологий. В частности, в ИБ пока не нашли отражения основные положения *объектно-ориентированного подхода*, ставшего основой при построении современных информационных систем. Не учитываются в ИБ и достижения в технологии программирования, основанные на накоплении и многократном использовании программистских знаний. Это очень серьезная проблема, затрудняющая прогресс в области ИБ.

Попытки создания больших систем еще в 60-х годах вскрыли многочисленные проблемы программирования, главной из которых является сложность создаваемых и сопровождаемых систем. Результатами исследований в области технологии программирования стали сначала структурированное программирование, затем *объектно-ориентированный подход*.

Объектно-ориентированный подход является основой современной технологии программирования, испытанным методом борьбы со сложностью систем. Представляется естественным и, более того, необходимым, стремление распространить этот подход и на системы информационной безопасности, для которых, как и для программирования в целом, имеет место упомянутая проблема сложности.

Любой разумный метод борьбы со сложностью опирается на *принцип «divide et impera» - «разделяй и властвуй». Этот* ***принцип*** *означает, что* сложная система *(информационной безопасности) на верхнем уровне должна состоять из небольшого числа относительно* независимых ***компонентов***. *Относительная независимость здесь и далее понимается как минимизация числа связей между компонентами. Затем* **декомпозиции** *подвергаются выделенные на первом этапе* ***компоненты****, и так далее до заданного* ***уровня детализации***. *В результате система оказывается представленной в виде иерархии с несколькими уровнями абстракции.* ***Структурный подход*** *опирается на* ***алгоритмическую*** **декомпозицию***, когда выделяются* ***функциональные элементы*** *системы.*

Основная проблема *структурного подхода* состоит в том, что он неприменим на ранних этапах анализа и моделирования предметной области, когда до алгоритмов и функций дело еще не дошло. Нужен подход

«широкого спектра», не имеющий такого концептуального разрыва с анализируемыми системами и применимый на всех этапах разработки и реализации *сложных систем*. Мы постараемся показать, что *объектно-ориентированный подход* удовлетворяет таким требованиям.

3.2 Основные понятия объектно-ориентированного подхода

Объектно-ориентированный подход использует объектную *декомпозицию*, то есть поведение системы описывается в терминах взаимодействия *объектов*.

Необходимо ввести понятие *класса*.

***Класс** - это абстракция множества сущностей реального мира, объединенных общностью структуры и поведения.*

***Объект** - это элемент класса, то есть абстракция определенной сущности.*

Подчеркнем, что **объекты** *активны, у них есть не только внутренняя структура, но и поведение, которое описывается так называемыми* **методами объекта**. Например, может быть определен *класс* «пользователь», характеризующий «пользователя вообще», то есть ассоциированные с пользователями данные и их поведение (*методы*). После этого может быть создан *объект* «пользователь Иванов» с соответствующей конкретизацией данных и, возможно, *методов*.

Следующую группу важнейших понятий объектного подхода составляют

- *инкапсуляция*,
- *наследование*,
- *полиморфизм*.

Основным инструментом борьбы со сложностью в *объектно-ориентированном подходе* является ***инкапсуляция** - сокрытие реализации* объектов *(их внутренней структуры и деталей реализации* методов*) с предоставлением вовне только строго определенных интерфейсов.*

Понятие «***полиморфизм***» *может трактоваться как способность* объекта *принадлежать более чем одному классу*. Введение этого понятия отражает необходимость смотреть на *объекты* под разными углами зрения, выделять при построении абстракций разные аспекты сущностей моделируемой предметной области, не нарушая при этом целостности *объекта*. (Строго говоря, существуют и другие виды *полиморфизма*, такие как перегрузка и параметрический *полиморфизм*, но нас они сейчас не интересуют.)

***Наследование** означает построение новых классов на основе существующих с возможностью добавления или переопределения данных и методов*. Наследование является важным инструментом борьбы с размножением сущностей без необходимости. Общая информация не

дублируется, указывается только то, что меняется. При этом *класс*-потомок помнит о своих «предках».

Очень важно и то, что *наследование* и *полиморфизм* в совокупности наделяют объектно-ориентированную систему способностью к относительно безболезненной эволюции. Средства информационной безопасности приходится постоянно модифицировать и обновлять, и если нельзя сделать так, чтобы это было экономически выгодно, ИБ из инструмента защиты превращается в обузу.

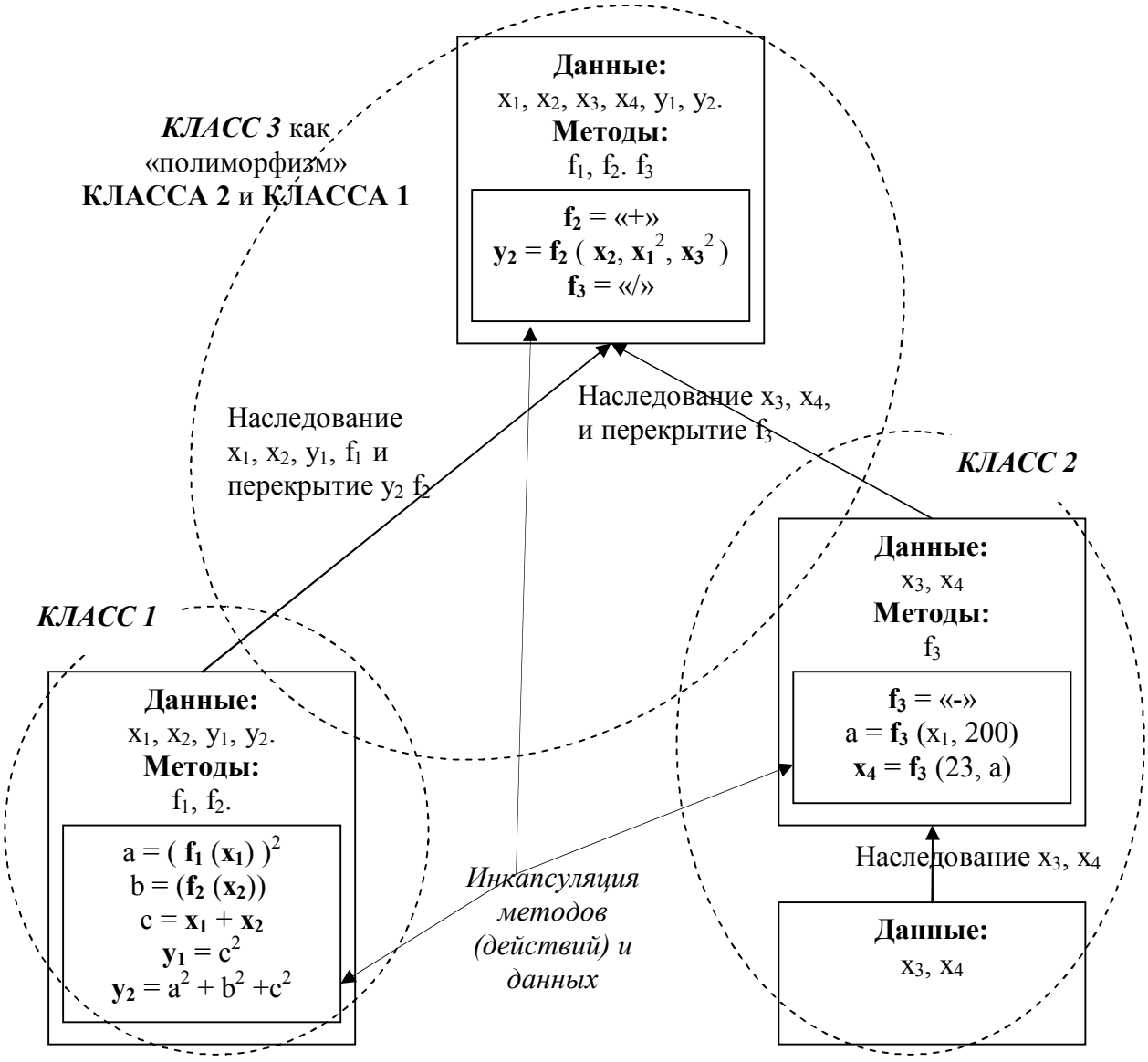


Рис. 3.1 - Объектно-ориентированный подход к вопросам ИБ

Объекты реального мира обладают, как правило, несколькими относительно независимыми характеристиками. Применительно к объектной модели будем называть такие характеристики ***гранями***.

Основными **гранями** *ИБ является:*

- *доступностью,*
- *целостностью,*
- *конфиденциальностью.*

Понятие *грани* позволяет более естественно, чем *полиморфизм*, смотреть на *объекты* с разных точек зрения и строить разноплановые абстракции.

Понятие ***уровня детализации*** *важно не только для визуализации* объектов*, но и для систематического рассмотрения сложных систем, представленных в иерархическом виде.* Само по себе оно очень простое: если очередной уровень иерархии рассматривается с *уровнем детализации* $n>0$, то следующий - с *уровнем* (n-1). *Объект* с *уровнем детализации* 0 считается атомарным.

Понятие *уровня детализации* показа позволяет рассматривать иерархии с потенциально бесконечной высотой, варьировать детализацию как *объектов* в целом, так и их *граней*.

Весьма распространенной конкретизацией *объектно-ориентированного подхода* являются ***компонентные объектные среды***, к числу которых принадлежит, например, JavaBeans. Здесь появляется два новых важных понятия: *компонент* и *контейнер*.

Компонент *можно определить как многократно используемый* ***объект***, *допускающий обработку в графическом инструментальном окружении и сохранение в долговременной памяти.*

Контейнеры *могут включать в себя множество* ***компонентов***, *образуя общий контекст взаимодействия с другими компонентами и с окружением. Контейнеры могут выступать в роли компонентов других контейнеров.*

***Компонентные объектные среды обладают всеми достоинствами, присущими объектно-ориентированному подходу*:**

- *инкапсуляция* объектных *компонентов* скрывает сложность реализации, делая видимым только предоставляемый вовне интерфейс;
- *наследование* позволяет развивать созданные ранее *компоненты*, не нарушая целостность объектной оболочки;
- *полиморфизм* по сути дает возможность группировать *объекты*, характеристики которых с некоторой точки зрения можно считать сходными.

Понятия же *компонента* и *контейнера* необходимы нам потому, что с их помощью можно естественным образом представить защищаемую ИС и сами защитные средства. В частности, *контейнер* может определять границы контролируемой зоны (задавать так называемый «периметр безопасности»).

3.3 Применение объектно-ориентированного подхода к рассмотрению защищаемых систем

Применим *объектно-ориентированный подход* к вопросам информационной безопасности.

Проблема обеспечения информационной безопасности - комплексная, защищать приходится *сложные системы*, и сами защитные средства тоже сложны, поэтому нам понадобятся все введенные понятия. Начнем с понятия *грани*.

Фактически три *грани* уже были введены: это доступность, целостность и конфиденциальность. Их можно рассматривать относительно независимо, и считается, что если все они обеспечены, то обеспечена и ИБ в целом (то есть субъектам информационных отношений не будет нанесен неприемлемый ущерб).

Таким образом, мы структурировали нашу цель. Теперь нужно структурировать средства ее достижения. Введем следующие *грани*:

- законодательные меры обеспечения информационной безопасности;
- административные меры (приказы и другие действия руководства организаций, связанных с защищаемыми информационными системами);
- процедурные меры (меры безопасности, ориентированные на людей);
- программно-технические меры.

Отметим, что, в принципе, их можно рассматривать и как результат варьирования *уровня детализации* (по этой причине мы будем употреблять словосочетания «законодательный уровень», «процедурный уровень» и т.п.). Законы и нормативные акты ориентированы на всех субъектов информационных отношений независимо от их организационной принадлежности (это могут быть как юридические, так и физические лица) в пределах страны (международные конвенции имеют даже более широкую область действия), административные меры - на всех субъектов в пределах организации, процедурные – на отдельных людей (или небольшие категории субъектов), программно-технические – на оборудование и программное обеспечение.

При такой трактовке в переходе с уровня на уровень можно усмотреть применение *наследования* (каждый следующий уровень не отменяет, а дополняет предыдущий), а также *полиморфизма* (субъекты выступают сразу в нескольких ипостасях - например, как инициаторы административных мер и как обычные пользователи, обязанные этим мерам подчиняться).

Очевидно, для всех выделенных, относительно независимых *граней* действует принцип *инкапсуляции* (это и значит, что *грани* «относительно независимы»). Более того, эти две совокупности *граней* можно назвать *ортогональными*, поскольку для фиксированной *грани* в одной совокупности (например, доступности) *грани* в другой совокупности должны пробегать все

множество возможных значений (нужно рассмотреть законодательные, административные, процедурные и программно-технические меры). *Ортогональных совокупностей* не должно быть много; думается, двух совокупностей с числом элементов, соответственно, 3 и 4 уже достаточно, так как они дают 12 комбинаций.

Продемонстрируем теперь, как можно рассматривать защищаемую ИС, варьируя *уровень детализации*.

Пусть интересы субъектов информационных отношений концентрируются вокруг ИС некой организации, располагающей двумя территориально разнесенными производственными площадками, на каждой из которых есть серверы, обслуживающие своих и внешних пользователей, а также пользователи, нуждающиеся во внутренних и внешних сервисах. Одна из площадок оборудована внешним подключением (то есть имеет выход в Internet).

При взгляде с нулевым *уровнем детализации* мы увидим лишь то, что у организации есть информационная система (см. рис. 3.2).

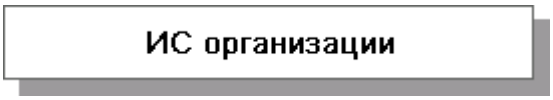


Рис. 3.2 - ИС при рассмотрении с уровнем детализации 0.

Уже здесь необходимо учесть законы, применимые к организациям, располагающим информационными системами. Возможно, какую-либо информацию нельзя хранить и обрабатывать на компьютерах, если ИС не была аттестована на соответствие определенным требованиям. На административном уровне могут быть декларированы цели, ради которых создавалась ИС, общие правила закупок, внедрения новых *компонентов*, эксплуатации и т.п. На процедурном уровне нужно определить требования к физической безопасности ИС и пути их выполнения, правила противопожарной безопасности и т.п. На программно-техническом уровне могут быть определены предпочтительные аппаратно-программные платформы и т.п.

По каким критериям проводить *декомпозицию* ИС – в значительной степени дело вкуса. Будем считать, что на первом *уровне детализации* делаются видимыми сервисы и пользователи, точнее, разделение на клиентскую и серверную часть (рис. 3.3).

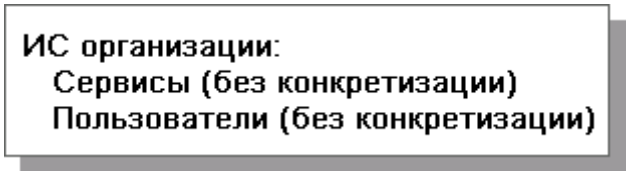


Рис. 3.3 - ИС при рассмотрении с уровнем детализации 1.

На этом уровне следует сформулировать требования к сервисам (к самому их наличию, к доступности, целостности и конфиденциальности предоставляемых информационных услуг), изложить способы выполнения этих требований, определить общие правила поведения пользователей, необходимый уровень их предварительной подготовки, методы контроля их поведения, порядок поощрения и наказания и т.п. Могут быть сформулированы требования и предпочтения по отношению к серверным и клиентским платформам.

На втором *уровне детализации* мы увидим следующее (см. рис. 3.4).

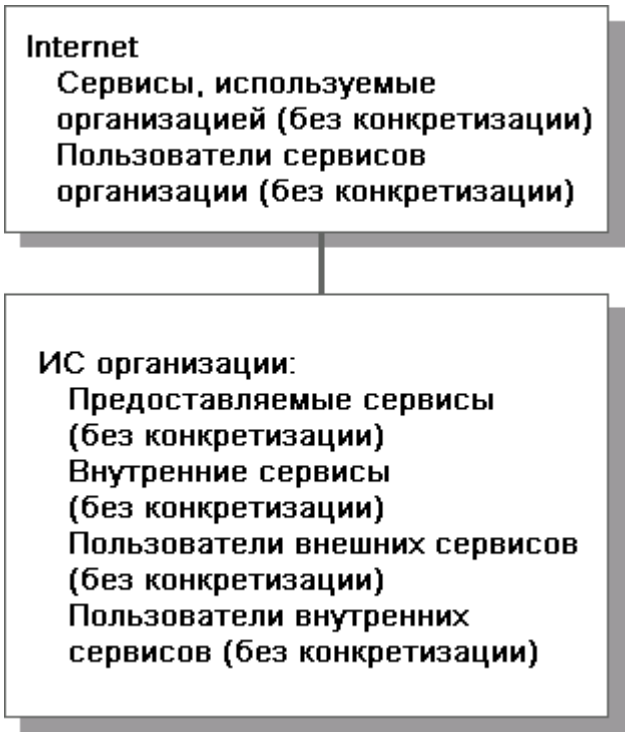


Рис. 3.4 - ИС при рассмотрении с уровнем детализации 2.

На этом уровне нас все еще не интересует внутренняя структура ИС организации, равно как и детали Internet. Констатируется только существование связи между этими сетями, наличие в них пользователей, а также предоставляемых и внутренних сервисов. Что это за сервисы, пока неважно.

Находясь на *уровне детализации* 2, мы должны учитывать законы, применимые к организациям, ИС которых снабжены внешними подключениями. Речь идет о допустимости такого подключения, о его защите, об ответственности пользователей, обращающихся к внешним сервисам, и об ответственности организаций, открывающих свои сервисы для внешнего доступа. Конкретизация аналогичной направленности, с учетом наличия внешнего подключения, должна быть выполнена на административном, процедурном и программно-техническом уровнях.

Обратим внимание на то, что *контейнер* (в смысле *компонентной объектной среды*) «ИС организации» задает границы контролируемой зоны,

в пределах которых организация проводит определенную политику. Internet живет по другим правилам, которые организация должна принимать, как данность.

Увеличивая *уровень детализации*, можно разглядеть две разнесенные производственные площадки и каналы связи между ними, распределение сервисов и пользователей по этим площадкам и средства обеспечения безопасности внутренних коммуникаций, специфику отдельных сервисов, разные категории пользователей и т.п. Мы, однако, на этом остановимся.

3.4 Недостатки традиционного подхода к информационной безопасности с объектной точки зрения

Исходя из основных положений *объектно-ориентированного подхода*, следует в первую очередь признать устаревшим традиционное деление на активные и пассивные сущности (*субъекты* и *объекты* в привычной для дообъектной ИБ терминологии).

Подобное деление устарело, по крайней мере, по двум причинам.

- Во-первых, ***в объектном подходе пассивных объектов нет.*** Можно считать, что все *объекты* активны одновременно и при необходимости вызывают *методы* друг друга. Как реализованы эти *методы* (и, в частности, как организован доступ к переменным и их значениям) - внутреннее дело вызываемого *объекта*; детали реализации скрыты, инкапсулированы. Вызывающему *объекту* доступен только предоставляемый интерфейс.
- Во-вторых, ***нельзя сказать, что какие-то программы (методы) выполняются от имени пользователя***. *Реализации* объектов *сложны, так что последние нельзя рассматривать всего лишь как инструменты выполнения воли пользователей.* Скорее можно считать, что пользователь прямо или (как правило) косвенно, на свой страх и риск, «просит» некоторый *объект* об определенной информационной услуге. Когда активизируется вызываемый *метод*, *объект* действует скорее от имени (во всяком случае, по воле) своего создателя, чем от имени вызвавшего его пользователя. Можно считать, что *объекты* обладают достаточной «свободой воли», чтобы выполнять действия, о которых пользователь не только не просил, но даже не догадывается об их возможности. Особенно это справедливо в сетевой среде и для программного обеспечения (ПО), полученного через Internet, но может оказаться верным и для коммерческого ПО, закупленного по всем правилам у солидной фирмы.

Для иллюстрации приведем следующий гипотетический пример. Банк, ИС которого имеет соединение с Internet, приобрел за рубежом автоматизированную банковскую систему (АБС). Только спустя некоторое время в банке решили, что внешнее соединение нуждается в защите, и

установили межсетевой экран. Изучение регистрационной информации экрана показало, что время от времени за рубеж отправляются IP-пакеты, содержащие какие-то непонятные данные (наверное, зашифрованные, решили в банке). Стали разбираться, куда же пакеты направляются, и оказалось, что идут они в фирму, разработавшую АБС. Возникло подозрение, что в АБС встроена закладка, чтобы получать информацию о деятельности банка. Связались с фирмой; там очень удивились, поначалу все отрицали, но в конце концов выяснили, что один из программистов не убрал из поставленного в банк варианта отладочную выдачу, которая была организована через сеть (как передача IP-пакетов специфического вида, с явно заданным IP-адресом рабочего места этого программиста). Таким образом, никакого злого умысла не было, однако некоторое время информация о платежах свободно гуляла по сетям.

Отметим, что при определении допустимости доступа важно не только (и не столько), кто обратился к *объекту*, но и то, какова **семантика** действия. Без привлечения семантики нельзя определить так называемые **«троянские программы»**, выполняющие, помимо декларированных, некоторые скрытые (обычно негативные) действия.

Следует также признать устаревшим и положение о том, что разграничение доступа направлено на защиту от злоумышленников. Приведенный выше пример показывает, что внутренние ошибки распределенных ИС представляют не меньшую опасность, а гарантировать их отсутствие в *сложных системах* современная технология программирования не позволяет.

В дообъектной ИБ одним из важнейших требований является **безопасность повторного использования** пассивных сущностей (таких, например, как динамически выделяемые области памяти). Очевидно, подобное требование вступает в конфликт с таким фундаментальным принципом, как *инкапсуляция*. *Объект* нельзя очистить внешним образом (заполнить нулями или случайной последовательностью бит), если только он сам не предоставляет соответствующий *метод*. При наличии такого *метода* надежность очистки зависит от корректности его реализации и вызова.

Одним из самых прочных стереотипов среди специалистов по ИБ является трактовка **операционной системы** как доминирующего **средства безопасности**. На разработку защищенных ОС выделяются значительные средства, зачастую в ущерб остальным направлениям защиты и, следовательно, в ущерб реальной безопасности. В современных ИС, выстроенных в многоуровневой архитектуре клиент/сервер, ОС не контролирует *объекты*, с которыми работают пользователи, равно как и действия самих пользователей, которые регистрируются и учитываются прикладными средствами. Основной функцией безопасности ОС становится защита возможностей, предоставляемых привилегированным пользователям, от атак обычных пользователей.

4.ОСНОВНЫЕ ОПРЕДЕЛЕНИЯ И КРИТЕРИИ КЛАССИФИКАЦИИ УГРОЗ

4.1 Основные понятия об угрозах

Угроза *- это потенциальная возможность определенным образом нарушить информационную безопасность.*

Попытка реализации угрозы называется ***атакой****, а тот, кто предпринимает такую попытку -* ***злоумышленником****. Потенциальные злоумышленники называются* ***источниками угрозы****.*

Чаще всего угроза является следствием наличия уязвимых мест в защите информационных систем (таких, например, как возможность доступа посторонних лиц к критически важному оборудованию или ошибки в программном обеспечении).

Промежуток времени от момента, когда появляется возможность использовать слабое место, и до момента, когда пробел ликвидируется, называется ***окном опасности****, ассоциированным с данным уязвимым местом.* Пока существует окно опасности, возможны успешные атаки на ИС.

Если речь идет об ошибках в ПО, то окно опасности «открывается» с появлением средств использования ошибки и ликвидируется при наложении заплат, ее исправляющих.

Для большинства уязвимых мест время существования окна опасности определяются следующими событиями:

1. *должно стать известно о средствах использования пробела в защите;*
2. *должны быть выпущены соответствующие заплаты;*
3. *заплаты должны быть установлены в защищаемой ИС.*

Новые уязвимые места и средства их использования появляются постоянно; это значит, во-первых, что почти всегда существуют окна опасности и, во-вторых, что отслеживание таких окон должно производиться постоянно, а выпуск и наложение заплат - как можно более оперативно.

Некоторые угрозы нельзя считать следствием каких-то ошибок или просчетов; они существуют в силу самой природы современных ИС. Например, угроза отключения электричества или выхода его параметров за допустимые границы существует в силу зависимости аппаратного обеспечения ИС от качественного электропитания.

Подчеркнем, что само понятие «угроза» в разных ситуациях зачастую трактуется по-разному. Например, для подчеркнуто открытой организации угроз конфиденциальности может просто не существовать - вся информация считается общедоступной; однако в большинстве случаев нелегальный доступ представляется серьезной опасностью. Иными словами, угрозы, как и все в ИБ, зависят от интересов субъектов информационных отношений (и от того, какой ущерб является для них неприемлемым).

Угрозы можно классифицировать по нескольким критериям:

- *по аспекту информационной безопасности (доступность, целостность, конфиденциальность), против которого угрозы направлены в первую очередь;*
- *по компонентам информационных систем, на которые угрозы нацелены (данные, программы, аппаратура, поддерживающая инфраструктура);*
- *по способу осуществления (случайные/преднамеренные действия природного/техногенного характера);*
- *по расположению источника угроз (внутри/вне рассматриваемой ИС).*

В качестве основного критерия будем использовать первый (по аспекту ИБ), привлекая при необходимости остальные.

4.2 Наиболее распространенные угрозы доступности

Самыми частыми и самыми опасными (с точки зрения размера ущерба) являются непреднамеренные ошибки штатных пользователей, операторов, системных администраторов и других лиц, обслуживающих информационные системы.

Иногда такие ошибки и являются собственно угрозами (неправильно введенные данные или ошибка в программе, вызвавшая крах системы), иногда они создают уязвимые места, которыми могут воспользоваться злоумышленники (таковы обычно ошибки администрирования). По некоторым данным, до 65% потерь - следствие непреднамеренных ошибок.

Очевидно, самый радикальный способ борьбы с непреднамеренными ошибками - максимальная автоматизация и строгий контроль.

Другие угрозы доступности классифицируем по компонентам ИС, на которые нацелены угрозы:

- *отказ пользователей;*
- *внутренний отказ информационной системы;*
- *отказ поддерживающей инфраструктуры.*

Обычно применительно к пользователям рассматриваются следующие угрозы:

- *нежелание работать с информационной системой* (чаще всего проявляется при необходимости осваивать новые возможности и при расхождении между запросами пользователей и фактическими возможностями и техническими характеристиками);
- *невозможность работать с системой в силу отсутствия соответствующей подготовки* (недостаток общей компьютерной грамотности, неумение интерпретировать диагностические сообщения, неумение работать с документацией и т.п.);

- *невозможность работать с системой в силу отсутствия технической поддержки* (неполнота документации, недостаток справочной информации и т.п.).

Основными источниками внутренних отказов являются:

- *отступление (случайное или умышленное) от установленных правил эксплуатации;*
- *выход системы из штатного режима эксплуатации в силу случайных или преднамеренных действий пользователей или обслуживающего персонала* (превышение расчетного числа запросов, чрезмерный объем обрабатываемой информации и т.п.);
- ошибки при (пере)конфигурировании системы;
- *отказы программного и аппаратного обеспечения;*
- *разрушение данных;*
- *разрушение или повреждение аппаратуры.*

По отношению к поддерживающей инфраструктуре рекомендуется рассматривать следующие угрозы:

- *нарушение работы* (случайное или умышленное) *систем связи, электропитания, водо- и/или теплоснабжения, кондиционирования;*
- разрушение или повреждение помещений;
- *невозможность или нежелание обслуживающего персонала и/или пользователей выполнять свои обязанности* (гражданские беспорядки, аварии на транспорте, террористический акт или его угроза, забастовка и т.п.).

Весьма опасны так называемые «обиженные» сотрудники - нынешние и бывшие. Как правило, они стремятся нанести вред организации-«обидчику», например:

- испортить оборудование;
- встроить логическую бомбу, которая со временем разрушит программы и/или данные;
- удалить данные.

Обиженные сотрудники, даже бывшие, знакомы с порядками в организации и способны нанести немалый ущерб. Необходимо следить за тем, чтобы при увольнении сотрудника его права доступа (логического и физического) к информационным ресурсам аннулировались.

Опасны, разумеется, стихийные бедствия и события, воспринимаемые как стихийные бедствия,- пожары, наводнения, землетрясения, ураганы. По статистике, на долю огня, воды и тому подобных «злоумышленников» (среди которых самый опасный - перебой электропитания) приходится 13% потерь, нанесенных информационным системам.

4.2.1 Примеры угроз доступности

Угрозы доступности могут выглядеть грубо - как повреждение или даже разрушение **оборудования** (в том числе носителей данных). Такое

повреждение может вызываться естественными причинами (чаще всего - грозами). К сожалению, находящиеся в массовом использовании источники бесперебойного питания не защищают от мощных кратковременных импульсов, и случаи выгорания оборудования - не редкость.

В принципе, мощный кратковременный импульс, способный разрушить данные на магнитных носителях, можно сгенерировать и искусственным образом - с помощью так называемых высокоэнергетических радиочастотных пушек. Но, наверное, в наших условиях подобную угрозу следует все же признать надуманной.

Действительно опасны протечки водопровода и отопительной системы. Часто организации, чтобы сэкономить на арендной плате, снимают помещения в домах старой постройки, делают косметический ремонт, но не меняют ветхие трубы. Автору курса довелось быть свидетелем ситуации, когда прорвало трубу с горячей водой, и системный блок компьютера (это была рабочая станция производства Sun Microsystems) оказался заполнен кипятком. Когда кипяток вылили, а компьютер просушили, он возобновил нормальную работу, но лучше таких опытов не ставить...

Летом, в сильную жару, норовят сломаться кондиционеры, установленные в серверных залах, набитых дорогостоящим оборудованием. В результате значительный ущерб наносится и репутации, и финансовому положению организации.

Общеизвестно, что периодически необходимо производить резервное копирование данных. Однако даже если это предложение выполняется, резервные носители зачастую хранят небрежно (к этому мы еще вернемся при обсуждении угроз конфиденциальности), не обеспечивая их защиту от вредного воздействия окружающей среды. И когда требуется восстановить данные, оказывается, что эти самые носители никак не желают читаться.

Перейдем к другим угрозам доступности - программных атаках на доступность.

В качестве средства вывода системы из штатного режима эксплуатации может использоваться агрессивное потребление ресурсов (обычно - полосы пропускания сетей, вычислительных возможностей процессоров или оперативной памяти). По расположению источника угрозы такое **потребление** подразделяется на локальное и удаленное. При просчетах в конфигурации системы локальная программа способна практически монополизировать процессор и/или физическую память, сведя скорость выполнения других программ к нулю.

Простейший пример удаленного потребления ресурсов - атака, получившая наименование «DoS – атака». Она представляет собой попытку переполнить таблицу «полуоткрытых» TCP-соединений сервера (установление соединений начинается, но не заканчивается). Такая атака по меньшей мере затрудняет установление новых соединений со стороны легальных пользователей, то есть сервер выглядит как недоступный.

Удаленное потребление ресурсов в последнее время проявляется в особенно опасной форме - как скоординированные распределенные атаки, когда на сервер с множества разных адресов с максимальной скоростью направляются вполне легальные запросы на соединение и/или обслуживание. Временем начала «моды» на подобные атаки можно считать февраль 2000 года, когда жертвами оказались несколько крупнейших систем электронной коммерции (точнее - владельцы и пользователи систем). Отметим, что если имеет место архитектурный просчет в виде разбалансированности между пропускной способностью сети и производительностью сервера, то защититься от распределенных атак на доступность крайне трудно.

Для выведения систем из штатного режима эксплуатации могут использоваться уязвимые места в виде программных и аппаратных ошибок. Например, программа «Teardrop» удаленно «подвешивает» компьютеры, эксплуатируя ошибку в сборке фрагментированных IP-пакетов.

4.2.2 Вредоносное программное обеспечение

Одним из опаснейших способов проведения атак является внедрение в атакуемые системы вредоносного программного обеспечения.

Выделяют следующие грани вредоносного ПО:

- *вредоносная функция;*
- *способ распространения;*
- *внешнее представление.*

Часть, осуществляющую разрушительную функцию, будем называть **«бомбой»**. Вообще говоря, спектр вредоносных функций неограничен, поскольку «бомба», как и любая другая программа, может обладать сколь угодно сложной логикой, но ***обычно «бомбы» предназначаются для:***

1. *внедрения другого вредоносного ПО;*
2. *получения контроля над атакуемой системой;*
3. *агрессивного потребления ресурсов;*
4. *изменения или разрушения программ и/или данных.*

По механизму распространения различают:

- ***вирусы*** *- код, обладающий способностью к распространению (возможно, с изменениями) путем внедрения в другие программы;*
- ***«черви»*** *- код, способный самостоятельно, то есть без внедрения в другие программы, вызывать распространение своих копий по ИС и их выполнение (для активизации вируса требуется запуск зараженной программы).*

Вирусы обычно распространяются локально, в пределах узла сети; для передачи по сети им требуется внешняя помощь, такая как пересылка зараженного файла. «Черви», напротив, ориентированы в первую очередь на распространение по сети.

Иногда само распространение вредоносного ПО вызывает агрессивное потребление ресурсов и, следовательно, является вредоносной функцией. Например, «черви» «съедают» полосу пропускания сети и ресурсы почтовых систем. По этой причине для атак на доступность они не нуждаются во встраивании специальных «бомб».

Вредоносный код, который выглядит как функционально полезная программа, называется ***троянским вирусом***. Например, обычная программа, будучи пораженной вирусом, становится троянской; порой троянские программы изготавливают вручную и подсовывают доверчивым пользователям в какой-либо привлекательной упаковке.

Отметим, что данные нами определения и приведенная классификация вредоносного ПО отличаются от общепринятых. Например, в ГОСТ Р 51275-99 «Защита информации. Объект информатизации. Факторы, воздействующие на информацию. Общие положения» содержится следующее определение:

*«**Программный вирус** - это исполняемый или интерпретируемый программный код, обладающий свойством несанкционированного распространения и самовоспроизведения в автоматизированных системах или телекоммуникационных сетях с целью изменить или уничтожить программное обеспечение и/или данные, хранящиеся в автоматизированных системах».*

На наш взгляд, подобное определение не совсем удачно, поскольку в нем смешаны функциональные и транспортные аспекты.

Окно опасности для вредоносного ПО появляется с выпуском новой разновидности «бомб», вирусов и/или «червей» и перестает существовать с обновлением базы данных антивирусных программ и наложением других необходимых заплат.

По традиции из всего вредоносного ПО наибольшее внимание общественности приходится на долю вирусов. Однако до марта 1999 года с полным правом можно было утверждать, что «несмотря на экспоненциальный рост числа известных вирусов, аналогичного роста количества инцидентов, вызванных ими, не зарегистрировано. Соблюдение несложных правил «компьютерной гигиены» практически сводит риск заражения к нулю. Там, где работают, а не играют, число зараженных компьютеров составляет лишь доли процента».

В марте 1999 года, с появлением вируса «Melissa», ситуация кардинальным образом изменилась. «Melissa» - это макровирус для файлов MS-Word, распространяющийся посредством электронной почты в присоединенных файлах. Когда такой (зараженный) присоединенный файл открывают, он рассылает свои копии по первым 50 адресам из адресной книги Microsoft Outlook. В результате почтовые серверы подвергаются атаке на доступность.

Вслед за «Melissa» появилась на свет целая серия вирусов, «червей» и их комбинаций: «Explorer.zip» (июнь 1999), «Bubble Boy» (ноябрь 1999),

«ILOVEYOU» (май 2000) и т.д. Не то что бы от них был особенно большой ущерб, но общественный резонанс они вызвали немалый.

Активное содержимое, помимо интерпретируемых компонентов документов и других файлов данных, имеет еще одно популярное обличье - так называемые мобильные агенты. Это программы, которые загружаются на другие компьютеры и там выполняются. Наиболее известные примеры мобильных агентов - Java-апплеты, загружаемые на пользовательский компьютер и интерпретируемые Internet-навигаторами. Оказалось, что разработать для них модель безопасности, оставляющую достаточно возможностей для полезных действий, не так-то просто; еще сложнее реализовать такую модель без ошибок. В августе 1999 года стали известны недочеты в реализации технологий ActiveX и Java в рамках Microsoft Internet Explorer, которые давали возможность размещать на Web-серверах вредоносные апплеты, позволяющие получать полный контроль над системой-визитером.

Для внедрения «бомб» часто используются ошибки типа «переполнение буфера», когда программа, работая с областью памяти, выходит за границы допустимого и записывает в нужные злоумышленнику места определенные данные. Так действовал еще в 1988 году знаменитый «червь Морриса»; в июне 1999 года хакеры нашли способ использовать аналогичный метод по отношению к Microsoft Internet Information Server (IIS), чтобы получить контроль над Web-сервером. Окно опасности охватило сразу около полутора миллионов серверных систем...

Не забыты современными злоумышленниками и испытанные троянские программы. Например, «троянцы» Back Orifice и Netbus позволяют получить контроль над пользовательскими системами с различными вариантами MS-Windows.

Таким образом, действие вредоносного ПО может быть направлено не только против доступности, но и против других основных аспектов информационной безопасности.

4.3 Основные угрозы целостности

На втором месте по размерам ущерба (после непреднамеренных ошибок и упущений) стоят кражи и подлоги. По данным газеты USA Today, еще в 1992 году в результате подобных противоправных действий с использованием персональных компьютеров американским организациям был нанесен общий ущерб в размере 882 миллионов долларов. Можно предположить, что реальный ущерб был намного больше, поскольку многие организации по понятным причинам скрывают такие инциденты; не вызывает сомнений, что в наши дни ущерб от такого рода действий вырос многократно.

В большинстве случаев виновниками оказывались штатные сотрудники организаций, отлично знакомые с режимом работы и мерами защиты. Это

еще раз подтверждает опасность внутренних угроз, хотя говорят и пишут о них значительно меньше, чем о внешних.

Целостность подразделяется на

- *статическую;*
- *динамическую.*

С целью нарушения статической целостности злоумышленник (как правило, штатный сотрудник) может:

- *ввести неверные данные;*
- *изменить данные.*

Показательный случай нарушения целостности имел место в 1996 году. Служащая Oracle (личный секретарь вице-президента) предъявила судебный иск, обвиняя президента корпорации в незаконном увольнении после того, как она отвергла его ухаживания. В доказательство своей правоты женщина привела электронное письмо, якобы отправленное ее начальником президенту. Содержание письма для нас сейчас не важно; важно время отправки. Дело в том, что вице-президент предъявил, в свою очередь, файл с регистрационной информацией компании сотовой связи, из которого явствовало, что в указанное время он разговаривал по мобильному телефону, находясь вдалеке от своего рабочего места. Таким образом, в суде состоялось противостояние «файл против файла». Очевидно, один из них был фальсифицирован или изменен, то есть была нарушена его целостность. Суд решил, что подделали электронное письмо (секретарша знала пароль вице-президента, поскольку ей было поручено его менять), и иск был отвергнут...

(Теоретически возможно, что оба фигурировавших на суде файла были подлинными, корректными с точки зрения целостности, а письмо отправили пакетными средствами, однако, на наш взгляд, это было бы очень странное для вице-президента действие.)

Еще один урок: ***угрозой целостности*** *является не только фальсификация или изменение данных, но и отказ от совершенных действий.* Если нет средств обеспечить «неотказуемость», компьютерные данные не могут рассматриваться в качестве доказательства.

Потенциально уязвимы с точки зрения нарушения ***целостности*** *не только* ***данные****, но и* ***программы****. Внедрение рассмотренного выше вредоносного ПО - пример подобного нарушения.*

Угрозами динамической целостности являются

- *нарушение атомарности транзакций;*
- *переупорядочение, кража, дублирование данных;*
- *внесение дополнительных сообщений (сетевых пакетов и т.п.).* Соответствующие действия в сетевой среде называются активным прослушиванием.

4.4 Основные угрозы конфиденциальности

Конфиденциальную информацию можно разделить на:

- *предметную;*
- *служебную.*

Служебная информация *(например, пароли пользователей) не относится к определенной предметной области, в информационной системе она играет техническую роль, но ее раскрытие особенно опасно, поскольку оно чревато получением несанкционированного доступа ко всей информации, в том числе предметной.*

Даже если информация хранится в компьютере или предназначена для компьютерного использования, угрозы ее конфиденциальности могут носить некомпьютерный и вообще нетехнический характер.

Если для доступа к различным системам используются многоразовые пароли или иная конфиденциальная информация, то наверняка эти данные будут храниться не только в голове, но и в записной книжке, которые пользователь часто оставляет на рабочем столе, а то и попросту теряет. Невозможно помнить много разных паролей и как результат применение несложных схем чередования или использование двух-трех легко запоминающихся (и столь же легко угадываемым) паролей. Описанный класс уязвимых мест можно назвать ***размещением конфиденциальных данных в среде, где им не обеспечена (зачастую - и не может быть обеспечена) необходимая защита***. Угроза же состоит в том, что кто-то не откажется узнать секреты, которые сами просятся в руки.

Помимо паролей, хранящихся в записных книжках пользователей, в этот класс попадает ***передача конфиденциальных данных в открытом виде*** (в разговоре, в письме, по сети), которая делает возможным перехват данных. Для атаки могут использоваться разные технические средства (подслушивание или прослушивание разговоров, пассивное прослушивание сети и т.п.), но идея одна - осуществить доступ к данным в тот момент, когда они наименее защищены.

Весьма опасной угрозой являются... ***выставки***, на которые многие организации, недолго думая, отправляют оборудование из производственной сети, со всеми хранящимися на них данными. Остаются прежними пароли, при удаленном доступе они продолжают передаваться в открытом виде. Это плохо даже в пределах защищенной сети организации; в объединенной сети выставки - это слишком суровое испытание честности всех участников.

Еще один пример изменения, о котором часто забывают, ***- хранение данных на резервных носителях.*** Для защиты данных на основных носителях применяются развитые системы управления доступом; копии же нередко просто лежат в шкафах и получить доступ к ним могут многие.

Перехват данных - очень серьезная угроза, и если конфиденциальность действительно является критичной, а данные передаются по многим каналам, их защита может оказаться весьма сложной и дорогостоящей. Технические средства перехвата хорошо проработаны, доступны, просты в эксплуатации, а установить их, например на кабельную сеть, может кто угодно, так что эту угрозу нужно принимать во внимание по отношению не только к внешним, но и к внутренним коммуникациям.

Кражи оборудования являются угрозой не только для резервных носителей, но и для компьютеров, особенно портативных. Часто ноутбуки оставляют без присмотра на работе или в автомобиле, иногда просто теряют.

Опасной нетехнической угрозой конфиденциальности являются ***методы морально-психологического воздействия, такие как маскарад*** - выполнение действий под видом лица, обладающего полномочиями для доступа к данным.

К неприятным угрозам, от которых трудно защищаться, можно отнести ***злоупотребление полномочиями.*** На многих типах систем привилегированный пользователь (например системный администратор) способен прочитать любой (незашифрованный) файл, получить доступ к почте любого пользователя и т.д. Другой пример - ***нанесение ущерба при сервисном обслуживании***. Обычно сервисный инженер получает неограниченный доступ к оборудованию и имеет возможность действовать в обход программных защитных механизмов.

Таковы основные угрозы, которые наносят наибольший ущерб субъектам информационных отношений.

ЧАСТЬ 2. УРОВНИ ОБЕСПЕЧЕНИЯ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

5. ЗАКОНОДАТЕЛЬНЫЙ УРОВЕНЬ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

В деле обеспечения информационной безопасности успех может принести только комплексный подход. ***Для защиты интересов субъектов информационных отношений необходимо сочетать меры следующих уровней:***

- *законодательного;*
- *административного (приказы и другие действия руководства организаций, связанных с защищаемыми информационными системами);*
- *процедурного (меры безопасности, ориентированные на людей);*
- *программно-технического.*

5.1 Понятие о законодательном уровне информационной безопасности

Законодательный уровень является важнейшим для обеспечения информационной безопасности. Большинство людей не совершают противоправных действий не потому, что это технически невозможно, а потому, что это осуждается и/или наказывается обществом, потому, что так поступать не принято.

На законодательном уровне различают две группы мер:

- *меры, направленные на создание и поддержание в обществе негативного* (в том числе с применением наказаний) *отношения к нарушениям и нарушителям информационной безопасности* (назовем их мерами ограничительной направленности);
- *направляющие и координирующие меры, способствующие повышению образованности общества в области информационной безопасности, помогающие в разработке и распространении средств обеспечения информационной безопасности* (меры созидательной направленности).

Самое важное (и, вероятно, самое трудное) на законодательном уровне - создать механизм, позволяющий согласовать процесс разработки законов с реалиями и прогрессом информационных технологий. Законы не могут опережать жизнь, но важно, чтобы отставание не было слишком большим, так как на практике, помимо прочих отрицательных моментов, это ведет к снижению информационной безопасности.

5.2 Обзор российского законодательства в области информационной безопасности

5.2.1 Правовые акты общего назначения, затрагивающие вопросы информационной безопасности

Основным законом Российской Федерации является Конституция, принятая 12 декабря 1993 года.

В соответствии со ***статьей 24 Конституции****, органы государственной власти и органы местного самоуправления, их должностные лица обязаны обеспечить каждому возможность ознакомления с документами и материалами, непосредственно затрагивающими его права и свободы, если иное не предусмотрено законом.*

Статья 41 Конституции *гарантирует право на знание фактов и обстоятельств, создающих угрозу для жизни и здоровья людей.* ***Статья 42 Конституции*** *- право на знание достоверной информации о состоянии окружающей среды*.

В принципе, право на информацию может реализовываться средствами бумажных технологий, но в современных условиях наиболее практичным и удобным для граждан является создание соответствующими законодательными, исполнительными и судебными органами информационных серверов и поддержание доступности и целостности представленных на них сведений, то есть обеспечение их (серверов) информационной безопасности.

Статья 23 Конституции *гарантирует право на личную и семейную тайну, на тайну переписки, телефонных переговоров, почтовых, телеграфных и иных сообщений, статья 29 - право свободно искать, получать, передавать, производить и распространять информацию любым законным способом.* Современная интерпретация этих положений включает обеспечение конфиденциальности данных, в том числе в процессе их передачи по компьютерным сетям, а также доступ к средствам защиты информации.

В Гражданском кодексе Российской Федерации (в своем изложении мы опираемся на редакцию от 15 мая 2001 года) фигурируют такие понятия, как банковская, коммерческая и служебная тайна. *Согласно* ***статье 139 ГК РФ****, информация составляет служебную или коммерческую тайну в случае, когда информация имеет действительную или потенциальную коммерческую ценность в силу неизвестности ее третьим лицам, к ней нет свободного доступа на законном основании, и обладатель информации принимает меры к охране ее конфиденциальности.* Это подразумевает, как минимум, компетентность в вопросах ИБ и наличие доступных (и законных) средств обеспечения конфиденциальности.

Весьма шепетильным в плане информационной безопасности является Уголовный кодекс Российской Федерации (редакция от 14 марта 2002 года).

Глава 28 УК РФ - «Преступления в сфере компьютерной информации» *- содержит три статьи:*

- *статья 272. Неправомерный доступ к компьютерной информации;*
- *статья 273. Создание, использование и распространение вредоносных программ для ЭВМ;*
- *статья 274. Нарушение правил эксплуатации ЭВМ, системы ЭВМ или их сети.*

Первая имеет дело с посягательствами на конфиденциальность, вторая - с вредоносным ПО, третья - с нарушениями доступности и целостности, повлекшими за собой уничтожение, блокирование или модификацию охраняемой законом информации ЭВМ. Включение в сферу действия УК РФ вопросов доступности информационных сервисов представляется нам очень своевременным.

Статья 138 УК РФ*, защищая конфиденциальность персональных данных, предусматривает наказание за нарушение тайны переписки, телефонных переговоров, почтовых, телеграфных или иных сообщений. Аналогичную роль для банковской и коммерческой тайны играет статья 183 УК РФ.*

Интересы государства в плане обеспечения конфиденциальности информации нашли наиболее полное выражение в ***Законе «О государственной тайне»*** (с изменениями и дополнениями от 6 октября 1997 года). В нем ***гостайна*** *определена как защищаемые государством сведения в области его военной, внешнеполитической, экономической, разведывательной, контрразведывательной и оперативно-розыскной деятельности, распространение которых может нанести ущерб безопасности Российской Федерации.* Там же дается определение ***средств защиты информации*** *- это технические, криптографические, программные и другие средства, предназначенные для защиты сведений, составляющих государственную тайну; средства, в которых они реализованы, а также средства контроля эффективности защиты информации.* Подчеркнем важность последней части определения.

5.2.2 Закон «Об информации, информатизации и защите информации»

Основополагающим среди российских законов, посвященных вопросам информационной безопасности, следует считать ***закон «Об информации, информатизации и защите информации» от 20 февраля 1995 года № 24-ФЗ*** (принят Государственной Думой 25 января 1995 года). В нем даются основные определения и намечаются направления развития законодательства в данной области.

Процитируем некоторые из этих определений:

- ***информация*** *- сведения о лицах, предметах, фактах, событиях, явлениях и процессах независимо от формы их представления;*

- **документированная информация (документ)** - зафиксированная на материальном носителе информация с реквизитами, позволяющими ее идентифицировать;
- **информационные процессы** - процессы сбора, обработки, накопления, хранения, поиска и распространения информации;
- ***информационная система*** - *организационно упорядоченная совокупность документов (массивов документов) и информационных технологий, в том числе с использованием средств вычислительной техники и связи, реализующих информационные процессы;*
- ***информационные ресурсы*** - *отдельные документы и отдельные массивы документов, документы и массивы документов в информационных системах (библиотеках, архивах, фондах, банках данных, других информационных системах);*
- **информация о гражданах (персональные данные)** - сведения о фактах, событиях и обстоятельствах жизни гражданина, позволяющие идентифицировать его личность;
- ***конфиденциальная информация*** - *документированная информация, доступ к которой ограничивается в соответствии с законодательством Российской Федерации;*
- ***пользователь (потребитель) информации*** - *субъект, обращающийся к информационной системе или посреднику за получением необходимой ему информации и пользующийся ею.*

Обратим внимание на гибкость определения конфиденциальной информации, которая не сводится к сведениям, составляющим государственную тайну, а также на понятие персональных данных, закладывающее основу защиты последних.

Закон выделяет следующие цели защиты информации:

- *предотвращение утечки, хищения, утраты, искажения, подделки информации;*
- *предотвращение угроз безопасности личности, общества, государства;*
- *предотвращение несанкционированных действий по уничтожению, модификации, искажению, копированию, блокированию информации;*
- *предотвращение других форм незаконного вмешательства в информационные ресурсы и информационные системы, обеспечение правового режима документированной информации как объекта собственности;*
- *защита конституционных прав граждан на сохранение личной тайны и конфиденциальности персональных данных, имеющихся в информационных системах;*

- *сохранение государственной тайны, конфиденциальности документированной информации в соответствии с законодательством;*
- *обеспечение прав субъектов в информационных процессах и при разработке, производстве и применении информационных систем, технологий и средств их обеспечения.*

Отметим, что Закон на первое место ставит сохранение конфиденциальности информации. Целостность представлена также достаточно полно, хотя и на втором месте. О доступности («предотвращение несанкционированных действий по ... блокированию информации») сказано довольно мало.

Продолжим цитирование:

«Защите подлежит любая документированная информация, неправомерное обращение с которой может нанести ущерб ее собственнику, владельцу, пользователю и иному лицу».

По сути, это положение констатирует, что защита информации направлена на обеспечение интересов субъектов информационных отношений. Далее.

«Режим защиты информации устанавливается:

- *в отношении сведений, отнесенных к государственной тайне, - уполномоченными органами на основании Закона Российской Федерации «О государственной тайне»;*
- *в отношении конфиденциальной документированной информации - собственником информационных ресурсов или уполномоченным лицом на основании настоящего Федерального закона;*
- *в отношении персональных данных - федеральным законом.»*

Здесь явно выделены три вида защищаемой информации, ко второму из которых принадлежит, в частности, коммерческая информация. Поскольку защите подлежит только документированная информация, необходимым условием является фиксация коммерческой информации на материальном носителе и снабжение ее реквизитами. Отметим, что в данном месте Закона речь идет только о конфиденциальности; остальные аспекты ИБ забыты.

Обратим внимание, что защиту государственной тайны и персональных данных берет на себя государство; за другую конфиденциальную информацию отвечают ее собственники.

Как же защищать информацию? В качестве основного закон предлагает для этой цели мощные универсальные средства: лицензирование и сертификацию. Процитируем **статью 19**.

1. Информационные системы, базы и банки данных, предназначенные для информационного обслуживания граждан и организаций, подлежат сертификации в порядке, установленном Законом Российской Федерации «О сертификации продукции и услуг».
2. Информационные системы органов государственной власти Российской Федерации и органов государственной власти субъектов

Российской Федерации, других государственных органов, организаций, которые обрабатывают документированную информацию с ограниченным доступом, а также средства защиты этих систем подлежат обязательной сертификации. Порядок сертификации определяется законодательством Российской Федерации.

3. Организации, выполняющие работы в области проектирования, производства средств защиты информации и обработки персональных данных, получают лицензии на этот вид деятельности. Порядок лицензирования определяется законодательством Российской Федерации.
4. Интересы потребителя информации при использовании импортной продукции в информационных системах защищаются таможенными органами Российской Федерации на основе международной системы сертификации.

Здесь трудно удержаться от риторического вопроса: а есть ли в России информационные системы без импортной продукции? Получается, что на защите интересов потребителей стоит в данном случае только таможня...

И еще несколько пунктов, теперь из **статьи 22**:

2. Владелец документов, массива документов, информационных систем обеспечивает уровень защиты информации в соответствии с законодательством Российской Федерации.
3. Риск, связанный с использованием несертифицированных информационных систем и средств их обеспечения, лежит на собственнике (владельце) этих систем и средств. Риск, связанный с использованием информации, полученной из несертифицированной системы, лежит на потребителе информации.
4. Собственник документов, массива документов, информационных систем может обращаться в организации, осуществляющие сертификацию средств защиты информационных систем и информационных ресурсов, для проведения анализа достаточности мер защиты его ресурсов и систем и получения консультаций.
5. Владелец документов, массива документов, информационных систем обязан оповещать собственника информационных ресурсов и (или) информационных систем о всех фактах нарушения режима защиты информации.

Из пункта 5 следует, что должны обнаруживаться все (успешные) атаки на ИС. Вспомним в этой связи один из результатов опроса (см. лекцию 1): около трети респондентов-американцев не знали, были ли взломаны их ИС за последние 12 месяцев. По нашему законодательству их можно было бы привлечь к ответственности...

Далее, **статья 23 «Защита прав субъектов в сфере информационных процессов и информатизации»** содержит следующий пункт:

2. Защита прав субъектов в указанной сфере осуществляется судом, арбитражным судом, третейским судом с учетом специфики правонарушений и нанесенного ущерба. Очень важными являются пункты статьи 5, касающиеся юридической силы электронного документа и электронной цифровой подписи:
3. Юридическая сила документа, хранимого, обрабатываемого и передаваемого с помощью автоматизированных информационных и телекоммуникационных систем, может подтверждаться электронной цифровой подписью. Юридическая сила электронной цифровой подписи признается при наличии в автоматизированной информационной системе программно-технических средств, обеспечивающих идентификацию подписи, и соблюдении установленного режима их использования.
4. Право удостоверять идентичность электронной цифровой подписи осуществляется на основании лицензии. Порядок выдачи лицензий определяется законодательством Российской Федерации.

Таким образом, Закон предлагает действенное средство контроля целостности и решения проблемы «неотказуемости» (невозможности отказаться от собственной подписи).

Таковы важнейшие, на наш взгляд, положения Закона «Об информации, информатизации и защите информации». На следующей странице будут рассмотрены другие законы РФ в области информационной безопасности.

5.2.3 Другие законы и нормативные акты

Следуя логике Закона «Об информации, информатизации и защите информации», продолжим наш обзор ***Законом «О лицензировании отдельных видов деятельности» от 8 августа 2001 года номер 128-ФЗ*** (Принят Государственной Думой 13 июля 2001 года). Начнем с основных определений.

Лицензия - *специальное разрешение на осуществление конкретного вида деятельности при обязательном соблюдении лицензионных требований и условий, выданное лицензирующим органом юридическому лицу или индивидуальному предпринимателю.*

Лицензируемый вид деятельности - вид деятельности, на осуществление которого на территории Российской Федерации требуется получение лицензии в соответствии с настоящим Федеральным законом.

Лицензирование - мероприятия, связанные с предоставлением лицензий, переоформлением документов, подтверждающих наличие лицензий, приостановлением и возобновлением действия лицензий, аннулированием лицензий и контролем лицензирующих органов за

соблюдением лицензиатами при осуществлении лицензируемых видов деятельности соответствующих лицензионных требований и условий.

Лицензирующие органы *- федеральные органы исполнительной власти, органы исполнительной власти субъектов Российской Федерации, осуществляющие лицензирование в соответствии с настоящим Федеральным законом.*

Лицензиат - юридическое лицо или индивидуальный предприниматель, имеющие лицензию на осуществление конкретного вида деятельности.»

Статья 17 Закона «О лицензировании отдельных видов деятельности» устанавливает перечень видов деятельности, на осуществление которых требуются лицензии. Нас будут интересовать следующие виды:

- распространение шифровальных (криптографических) средств;
- техническое обслуживание шифровальных (криптографических) средств;
- предоставление услуг в области шифрования информации;
- разработка и производство шифровальных (криптографических) средств, защищенных с использованием шифровальных (криптографических) средств информационных систем, телекоммуникационных систем;
- выдача сертификатов ключей электронных цифровых подписей, регистрация владельцев электронных цифровых подписей, оказание услуг, связанных с использованием электронных цифровых подписей и подтверждением подлинности электронных цифровых подписей;
- выявление электронных устройств, предназначенных для негласного получения информации, в помещениях и технических средствах (за исключением случая, если указанная деятельность осуществляется для обеспечения собственных нужд юридического лица или индивидуального предпринимателя);
- разработка и (или) производство средств защиты конфиденциальной информации;
- техническая защита конфиденциальной информации;
- разработка, производство, реализация и приобретение в целях продажи специальных технических средств, предназначенных для негласного получения информации, индивидуальными предпринимателями и юридическими лицами, осуществляющими предпринимательскую деятельность.

Необходимо учитывать, что, согласно статье 1, действие данного Закона не распространяется на следующие виды деятельности:

- деятельность, связанная с защитой государственной тайны;
- деятельность в области связи;
- образовательная деятельность.

Основными лицензирующими органами в области защиты информации являются

- ***Федеральное агентство правительственной связи и информации (ФАПСИ)*** *ведает всем, что связано с криптографией,*
- ***Гостехкомиссия*** *лицензирует деятельность по защите конфиденциальной информации. Эти же организации возглавляют работы по сертификации средств соответствующей направленности.*

Кроме того, ввоз и вывоз средств криптографической защиты информации (шифровальной техники) и нормативно-технической документации к ней может осуществляться исключительно на основании лицензии Министерства внешних экономических связей Российской Федерации, выдаваемой на основании решения ФАПСИ. Все эти вопросы регламентированы соответствующими указами Президента и постановлениями Правительства РФ, которые мы здесь перечислять не будем.

В эпоху глобальных коммуникаций важную роль играет ***Закон «Об участии в международном информационном обмене» от 4 июля 1996 года номер 85-ФЗ*** (принят Государственной Думой 5 июня 1996 года). В нем, как и в Законе «Об информации...», основным защитным средством являются лицензии и сертификаты.

10 января 2002 года Президентом был подписан очень важный ***закон «Об электронной цифровой подписи» номер 1-ФЗ*** (принят Государственной Думой 13 декабря 2001 года), развивающий и конкретизирующий приведенные выше положения закона «Об информации...». Его роль поясняется в **статье 1**.

1. Целью настоящего Федерального закона является обеспечение правовых условий использования электронной цифровой подписи в электронных документах, при соблюдении которых электронная цифровая подпись в электронном документе признается равнозначной собственноручной подписи в документе на бумажном носителе.
2. Действие настоящего Федерального закона распространяется на отношения, возникающие при совершении гражданско-правовых сделок и в других предусмотренных законодательством Российской Федерации случаях. Действие настоящего Федерального закона не распространяется на отношения, возникающие при использовании иных аналогов собственноручной подписи.

Закон вводит следующие основные понятия:

Электронный документ - документ, в котором информация представлена в электронно-цифровой форме.

Электронная цифровая подпись - реквизит электронного документа, предназначенный для защиты данного электронного документа от подделки, полученный в результате криптографического преобразования информации с

использованием закрытого ключа электронной цифровой подписи и позволяющий идентифицировать владельца сертификата ключа подписи, а также установить отсутствие искажения информации в электронном документе.

Владелец сертификата ключа подписи - физическое лицо, на имя которого удостоверяющим центром выдан сертификат ключа подписи и которое владеет соответствующим закрытым ключом электронной цифровой подписи, позволяющим с помощью средств электронной цифровой подписи создавать свою электронную цифровую подпись в электронных документах (подписывать электронные документы).

Средства электронной цифровой подписи - аппаратные и (или) программные средства, обеспечивающие реализацию хотя бы одной из следующих функций: создание электронной цифровой подписи в электронном документе с использованием закрытого ключа электронной цифровой подписи, подтверждение с использованием открытого ключа электронной цифровой подписи подлинности электронной цифровой подписи в электронном документе, создание закрытых и открытых ключей электронных цифровых подписей.

Сертификат средств электронной цифровой подписи - документ на бумажном носителе, выданный в соответствии с правилами системы сертификации для подтверждения соответствия средств электронной цифровой подписи установленным требованиям.

Закрытый ключ электронной цифровой подписи - уникальная последовательность символов, известная владельцу сертификата ключа подписи и предназначенная для создания в электронных документах электронной цифровой подписи с использованием средств электронной цифровой подписи.

Открытый ключ электронной цифровой подписи - уникальная последовательность символов, соответствующая закрытому ключу электронной цифровой подписи, доступная любому пользователю информационной системы и предназначенная для подтверждения с использованием средств электронной цифровой подписи подлинности электронной цифровой подписи в электронном документе.

Сертификат ключа подписи - документ на бумажном носителе или электронный документ с электронной цифровой подписью уполномоченного лица удостоверяющего центра, которые включают в себя открытый ключ электронной цифровой подписи и выдаются удостоверяющим центром участнику информационной системы для подтверждения подлинности электронной цифровой подписи и идентификации владельца сертификата ключа подписи.

Подтверждение подлинности электронной цифровой подписи в электронном документе - положительный результат проверки соответствующим сертифицированным средством электронной цифровой подписи с использованием сертификата ключа подписи принадлежности

электронной цифровой подписи в электронном документе владельцу сертификата ключа подписи и отсутствия искажений в подписанном данной электронной цифровой подписью электронном документе.

Пользователь сертификата ключа подписи - физическое лицо, использующее полученные в удостоверяющем центре сведения о сертификате ключа подписи для проверки принадлежности электронной цифровой подписи владельцу сертификата ключа подписи.

Информационная система общего пользования - информационная система, которая открыта для использования всеми физическими и юридическими лицами и в услугах которой этим лицам не может быть отказано.

Корпоративная информационная система - информационная система, участниками которой может быть ограниченный круг лиц, определенный ее владельцем или соглашением участников этой информационной системы.

Пересказать такие определения своими словами невозможно... Обратим внимание на неоднозначное использование термина «сертификат», которое, впрочем, не должно привести к путанице. Кроме того, данное здесь определение электронного документа слабее, чем в Законе «Об информации...», поскольку нет упоминания реквизитов.

Согласно Закону, электронная цифровая подпись в электронном документе равнозначна собственноручной подписи в документе на бумажном носителе при одновременном соблюдении следующих условий:

- сертификат ключа подписи, относящийся к этой электронной цифровой подписи, не утратил силу (действует) на момент проверки или на момент подписания электронного документа при наличии доказательств, определяющих момент подписания;
- подтверждена подлинность электронной цифровой подписи в электронном документе;
- электронная цифровая подпись используется в соответствии со сведениями, указанными в сертификате ключа подписи.

Закон определяет сведения, которые должен содержать сертификат ключа подписи:

- уникальный регистрационный номер сертификата ключа подписи, даты начала и окончания срока действия сертификата ключа подписи, находящегося в реестре удостоверяющего центра;
- фамилия, имя и отчество владельца сертификата ключа подписи или псевдоним владельца. В случае использования псевдонима запись об этом вносится удостоверяющим центром в сертификат ключа подписи;
- открытый ключ электронной цифровой подписи;
- наименование средств электронной цифровой подписи, с которыми используется данный открытый ключ электронной цифровой подписи;

- наименование и местонахождение удостоверяющего центра, выдавшего сертификат ключа подписи;
- сведения об отношениях, при осуществлении которых электронный документ с электронной цифровой подписью будет иметь юридическое значение.

На этом заканчиваем обзор законов РФ, относящихся к информационной безопасности.

5.3 Обзор зарубежного законодательства в области информационной безопасности

Очертим некоторые законы нескольких стран (в первую очередь - США), поскольку только в США таких законодательных актов около 500.

Ключевую роль играет ***американский «Закон об информационной безопасности» (Computer Security Act of 1987, Public Law 100-235 (H.R. 145), January 8, 1988***). Его цель - реализация минимально достаточных действий по обеспечению безопасности информации в федеральных компьютерных системах, без ограничений всего спектра возможных действий.

Характерно, что уже в начале Закона называется конкретный исполнитель - Национальный институт стандартов и технологий (НИСТ), отвечающий за выпуск стандартов и руководств, направленных на защиту от уничтожения и несанкционированного доступа к информации, а также от краж и подлогов, выполняемых с помощью компьютеров. Таким образом, имеется в виду как регламентация действий специалистов, так и повышение информированности всего общества.

В 1997 году появилось продолжение описанного закона - **законопроект «О совершенствовании информационной безопасности»** (Computer Security Enhancement Act of 1997, H.R. 1903), направленный на усиление роли Национального института стандартов и технологий и упрощение операций с криптосредствами.

В законопроекте констатируется, что частный сектор готов предоставить криптосредства для обеспечения конфиденциальности и целостности (в том числе аутентичности) данных, что разработка и использование шифровальных технологий должны происходить на основании требований рынка, а не распоряжений правительства. Кроме того, здесь отмечается, что за пределами США имеются сопоставимые и общедоступные криптографические технологии, и это следует учитывать при выработке экспортных ограничений, чтобы не снижать конкурентоспособность американских производителей аппаратного и программного обеспечения.

Для защиты федеральных ИС рекомендуется более широко применять технологические решения, основанные на разработках частного сектора.

Кроме того, предлагается оценить возможности общедоступных зарубежных разработок.

За четыре года (1997-2001 гг.) на законодательном и других уровнях информационной безопасности США было сделано многое. Смягчены экспортные ограничения на криптосредства (в январе 2000 г.). Сформирована инфраструктура с открытыми ключами. Разработано большое число стандартов (например, новый стандарт электронной цифровой подписи - FIPS 186-2, январь 2000 г.). Все это позволило не заострять более внимания на криптографии как таковой, а сосредоточиться на одном из ее важнейших приложений - аутентификации, рассматривая ее по отработанной на криптосредствах методике. Очевидно, что, независимо от судьбы законопроекта, в США будет сформирована национальная инфраструктура электронной аутентификации. В данном случае законотворческая деятельность идет в ногу с прогрессом информационных технологий.

Конечно, в законодательстве США имеются в достаточном количестве и положения ограничительной направленности, и директивы, защищающие интересы таких ведомств, как Министерство обороны, АНБ, ФБР, ЦРУ, но мы не будем на них останавливаться. Желающие могут прочитать раздел «Законодательная база в области защиты информации» в превосходной статье О. Беззубцева и А. Ковалева «О лицензировании и сертификации в области защиты информации» (Jet Info, 1997, 4).

В законодательстве ФРГ выделим весьма развернутый (44 раздела) ***Закон о защите данных (Federal Data Protection Act of December 20, 1990 (BGBl.I 1990 S.2954), amended by law of September 14, 1994 (BGBl. I S. 2325)***). Он целиком посвящен защите персональных данных.

Как, вероятно, и во всех других законах аналогичной направленности, в данном случае устанавливается приоритет интересов национальной безопасности над сохранением тайны частной жизни. В остальном права личности защищены весьма тщательно. Например, если сотрудник фирмы обрабатывает персональные данные в интересах частных компаний, он дает подписку о неразглашении, которая действует и после перехода на другую работу. Государственные учреждения, хранящие и обрабатывающие персональные данные, несут ответственность за нарушение тайны частной жизни «субъекта данных», как говорится в Законе. В материальном выражении ответственность ограничена верхним пределом в 250 тысяч немецких марок.

Из законодательства Великобритании упомянем семейство так называемых ***добровольных стандартов BS 7799***, помогающих организациям на практике сформировать программы безопасности. В последующих лекциях мы еще вернемся к рассмотрению этих стандартов; здесь же отметим, что они действительно работают, несмотря на «добровольность» (или благодаря ей?).

В современном мире глобальных сетей законодательная база должна быть согласована с международной практикой. В этом плане поучителен

пример Аргентины. В конце марта 1996 года компетентными органами Аргентины был арестован Хулио Цезар Ардита, 21 года, житель Буэнос-Айреса, системный оператор электронной доски объявлений «Крик», известный в компьютерном подполье под псевдонимом «El Griton». Ему вменялись в вину систематические вторжения в компьютерные системы ВМС США, НАСА, многих крупнейших американских университетов, а также в компьютерные системы Бразилии, Чили, Кореи, Мексики и Тайваня. Однако, несмотря на тесное сотрудничество компетентных органов Аргентины и США, Ардита был отпущен без официального предъявления обвинений, поскольку по аргентинскому законодательству вторжение в компьютерные системы не считается преступлением. Кроме того, в силу принципа «двойной криминальности», действующего в международных правовых отношениях, Аргентина не может выдать хакера американским властям. Дело Ардита показывает, каким может быть будущее международных компьютерных вторжений при отсутствии всеобщих или хотя бы двусторонних соглашений о борьбе с компьютерной преступностью.

5.4 О текущем состоянии российского законодательства в области информационной безопасности

Как уже отмечалось, ***самое важное (и, вероятно, самое трудное) на законодательном уровне - создать механизм, позволяющий согласовать процесс разработки законов с реалиями и прогрессом информационных технологий.*** Пока такого механизма нет и, увы, не предвидится. Сейчас бессмысленно задаваться вопросом, чего не хватает российскому законодательству в области ИБ, это все равно, что интересоваться у пунктирного отрезка, чего тому не хватает, чтобы покрыть всю плоскость. Даже чисто количественное сопоставление с законодательством США показывает, что наша законодательная база явно неполна.

Справедливости ради необходимо отметить, что ограничительная составляющая в российском законодательстве представлена существенно лучше, чем координирующая и направляющая. *Глава 28 Уголовного кодекса РФ достаточно полно охватывает основные аспекты информационной безопасности, однако обеспечить реализацию соответствующих статей пока еще сложно.*

Положения базового Закона «Об информации, информатизации и защите информации» носят весьма общий характер, а основное содержание статей, посвященных информационной безопасности, сводится к необходимости использовать исключительно сертифицированные средства, что, в общем, правильно, но далеко не достаточно. Характерно, что Закон разъясняет вопросы ответственности в случае использования несертифицированных средств, но что делать, если нарушение ИБ произошло в системе, построенной строго по правилам? Кто возместит

ущерб субъектам информационных отношений? Поучителен в этом отношении рассмотренный выше закон ФРГ о защите данных.

Законодательством определены органы, ведающие лицензированием и сертификацией. (Отметим в этой связи, что Россия - одна из немногих стран (в список еще входят Вьетнам, Китай, Пакистан), сохранивших жесткий государственный контроль за производством и распространением внутри страны средств обеспечения ИБ, в особенности продуктов криптографических технологий.) Но кто координирует, финансирует и направляет проведение исследований в области ИБ, разработку отечественных средств защиты, адаптацию зарубежных продуктов? Законодательством США определена главная ответственная организация - НИСТ, которая исправно выполняет свою роль. В Великобритании имеются содержательные добровольные стандарты ИБ, помогающие организациям всех размеров и форм собственности. У нас пока ничего такого нет.

В области информационной безопасности законы реально преломляются и работают через нормативные документы, подготовленные соответствующими ведомствами. В этой связи очень важны Руководящие документы Гостехкомиссии России, определяющие требования к классам защищенности средств вычислительной техники и автоматизированных систем. Особенно выделим утвержденный в июле 1997 года Руководящий документ по межсетевым экранам, вводящий в официальную сферу один из самых современных классов защитных средств.

В современном мире глобальных сетей нормативно-правовая база должна быть согласована с международной практикой. Особое внимание следует обратить на то, что желательно привести российские стандарты и сертификационные нормативы в соответствие с международным уровнем информационных технологий вообще и информационной безопасности в частности. Есть целый ряд оснований для того, чтобы это сделать. Одно из них - необходимость защищенного взаимодействия с зарубежными организациями и зарубежными филиалами российских компаний. Второе (более существенное) - доминирование аппаратно-программных продуктов зарубежного производства.

Проблема сертификации аппаратно-программных продуктов зарубежного производства действительно сложна, однако, как показывает опыт европейских стран, решить ее можно. Сложившаяся в Европе система сертификации по требованиям информационной безопасности позволила оценить операционные системы, системы управления базами данных и другие разработки американских компаний. Вхождение России в эту систему и участие российских специалистов в сертификационных испытаниях в состоянии снять имеющееся противоречие между независимостью в области информационных технологий и информационной безопасностью без какого-либо ущерба для национальной безопасности.

Подводя итог, можно наметить следующие основные направления деятельности на законодательном уровне:

- *разработка новых законов с учетом интересов всех категорий субъектов информационных отношений;*
- *обеспечение баланса созидательных и ограничительных (в первую очередь преследующих цель наказать виновных) законов;*
- *интеграция в мировое правовое пространство;*
- *учет современного состояния информационных технологий.*

6. СТАНДАРТЫ И СПЕЦИФИКАЦИИ В ОБЛАСТИ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

Стандарты и спецификации бывают двух разных видов:

- ***оценочных стандартов**, направленных на классификацию информационных систем и средств защиты по требованиям безопасности;*
- ***технических спецификаций**, регламентирующих различные аспекты реализации средств защиты.*

6.1 Оценочные стандарты и технические спецификации. «Оранжевая книга» как оценочный стандарт

6.1.1 Основные понятия

Оценочные стандарты выделяют важнейшие, с точки зрения ИБ, аспекты ИС, играя роль архитектурных спецификаций. Другие технические спецификации определяют, как строить ИС предписанной архитектуры.

Исторически первым оценочным стандартом, получившим широкое распространение и оказавшим огромное влияние на базу стандартизации ИБ во многих странах, стал стандарт Министерства обороны США «Критерии оценки доверенных компьютерных систем». Данный труд, называемый чаще всего по цвету обложки «Оранжевой книгой», был впервые опубликован в августе 1983 года. «Оранжевая книга» поясняет понятие безопасной системы, которая «управляет, с помощью соответствующих средств, доступом к информации, так что только должным образом авторизованные лица или процессы, действующие от их имени, получают право читать, записывать, создавать и удалять информацию».

Очевидно, однако, что абсолютно безопасных систем не существует, это абстракция. Оценивается лишь степень доверия, которое можно оказать той или иной системе.

Степень доверия оценивается по двум основным критериям.

- ***Политика безопасности** - набор законов, правил и норм поведения, определяющих, как организация обрабатывает, защищает и распространяет информацию. Политика безопасности - это активный аспект защиты, включающий в себя анализ возможных угроз и выбор мер противодействия.* В частности, правила определяют, в каких случаях пользователь может оперировать конкретными наборами данных. Чем выше степень доверия системе, тем строже и многообразнее должна быть политика безопасности.
- ***Уровень гарантированности** - мера доверия, которая может быть оказана архитектуре и реализации ИС. Это пассивный аспект защиты.* Уровень гарантированности показывает, насколько

корректны механизмы, отвечающие за реализацию политики безопасности.

Концепция доверенной вычислительной базы является центральной при оценке степени доверия безопасности.

Доверенная вычислительная база *- это совокупность защитных механизмов ИС (включая аппаратное и программное обеспечение), отвечающих за проведение в жизнь политики безопасности.* Качество вычислительной базы определяется исключительно ее реализацией и корректностью исходных данных, которые вводит системный администратор.

Основное назначение доверенной вычислительной базы *- выполнять функции монитора обращений, то есть контролировать допустимость выполнения субъектами (активными сущностями ИС, действующими от имени пользователей) определенных операций над объектами (пассивными сущностями).*

Монитор проверяет каждое обращение пользователя к программам или данным на предмет согласованности с набором действий, допустимых для пользователя.

Монитор обращений должен обладать тремя качествами:

- ***Изолированность.*** Необходимо предупредить возможность отслеживания работы монитора.
- ***Полнота.*** Монитор должен вызываться при каждом обращении, не должно быть способов обойти его.
- ***Верифицируемость.*** Монитор должен быть компактным, чтобы его можно было проанализировать и протестировать, будучи уверенным в полноте тестирования.

Реализация монитора обращений называется ***ядром безопасности****. Ядро безопасности - это основа, на которой строятся все защитные механизмы.* Помимо перечисленных выше свойств монитора обращений, ядро должно гарантировать собственную неизменность.

Границу доверенной вычислительной базы называют ***периметром безопасности****.* Как уже указывалось, компоненты, лежащие вне периметра безопасности, вообще говоря, могут не быть доверенными. С развитием распределенных систем понятию «периметр безопасности» все чаще придают другой смысл, имея в виду границу владений определенной организации. То, что находится внутри владений, считается доверенным, а то, что вне, - нет.

6.1.2 Механизмы безопасности

Согласно «Оранжевой книге», политика безопасности должна обязательно включать в себя следующие элементы:

- *произвольное управление доступом;*
- *безопасность повторного использования объектов;*
- *метки безопасности;*
- *принудительное управление доступом.*

Произвольное управление доступом (называемое иногда дискреционным) - это метод разграничения доступа к объектам, основанный на учете личности субъекта или группы, в которую субъект входит. Произвольность управления состоит в том, что некоторое лицо (обычно владелец объекта) может по своему усмотрению предоставлять другим субъектам или отбирать у них права доступа к объекту.

Безопасность повторного использования объектов - важное дополнение средств управления доступом, предохраняющее от случайного или преднамеренного извлечения конфиденциальной информации из «мусора». Безопасность повторного использования должна гарантироваться для областей оперативной памяти (в частности, для буферов с образами экрана, расшифрованными паролями и т.п.), для дисковых блоков и магнитных носителей в целом.

Для реализации **принудительного управления доступом** с субъектами и объектами ассоциируются **метки безопасности**. Метка субъекта описывает его благонадежность, метка объекта - степень конфиденциальности содержащейся в нем информации.

Согласно «Оранжевой книге», метки безопасности состоят из двух частей:

- уровня секретности;
- списка категорий.

Уровни секретности образуют упорядоченное множество, категории - неупорядоченное. Назначение последних - описать предметную область, к которой относятся данные.

Принудительное (или мандатное) управление доступом основано на сопоставлении меток безопасности субъекта и объекта.

Субъект может читать информацию из объекта, если уровень секретности субъекта не ниже, чем у объекта, а все категории, перечисленные в метке безопасности объекта, присутствуют в метке субъекта. В таком случае говорят, что метка субъекта доминирует над меткой объекта. Смысл сформулированного правила понятен - читать можно только то, что положено.

Субъект может записывать информацию в объект, если метка безопасности объекта доминирует над меткой субъекта. В частности, «конфиденциальный» субъект может записывать данные в секретные файлы,

но не может - в несекретные (разумеется, должны также выполняться ограничения на набор категорий).

Описанный способ управления доступом называется **принудительным**, поскольку он не зависит от воли субъектов (даже системных администраторов). После того, как зафиксированы метки безопасности субъектов и объектов, оказываются зафиксированными и права доступа.

Если понимать политику безопасности узко, как правила разграничения доступа, то механизм подотчетности является дополнением подобной политики. Цель подотчетности - в каждый момент времени знать, кто работает в системе и что делает. ***Средства подотчетности делятся на три категории:***

- *идентификация и аутентификация;*
- *предоставление доверенного пути;*
- *анализ регистрационной информации.*

Обычный способ идентификации - ввод имени пользователя при входе в систему. Стандартное средство проверки подлинности (аутентификации) пользователя - пароль.

Доверенный путь связывает пользователя непосредственно с доверенной вычислительной базой, минуя другие, потенциально опасные компоненты ИС. Цель предоставления доверенного пути - дать пользователю возможность убедиться в подлинности обслуживающей его системы.

Анализ регистрационной информации (аудит) имеет дело с действиями (событиями), так или иначе затрагивающими безопасность системы.

Переходя к пассивным аспектам защиты, укажем, что в «Оранжевой книге» ***рассматривается два вида гарантированности****:*

1. *Операционная гарантированность относится к архитектурным и реализационным аспектам системы.* Операционная гарантированность - это способ убедиться в том, что архитектура системы и ее реализация действительно реализуют избранную политику безопасности. Операционная гарантированность включает в себя проверку следующих элементов:
   - архитектура системы;
   - целостность системы;
   - проверка тайных каналов передачи информации;
   - доверенное администрирование;
   - доверенное восстановление после сбоев.
2. *технологическая гарантированность - к методам построения и сопровождения.* Технологическая гарантированность охватывает весь жизненный цикл ИС, то есть периоды проектирования, реализации, тестирования, продажи и сопровождения. Все перечисленные действия

должны выполняться в соответствии с жесткими стандартами, чтобы исключить утечку информации и нелегальные «закладки».

6.1.3 Классы безопасности

В «Оранжевой книге» определяется четыре уровня доверия - D, C, B и A. Уровень D предназначен для систем, признанных неудовлетворительными. По мере перехода от уровня C к A к системам предъявляются все более жесткие требования. Уровни C и B подразделяются на классы (C1, C2, B1, B2, B3) с постепенным возрастанием степени доверия.

Классификацию, введенную в «Оранжевой книге» можно сформулировать так:

- *уровень C - произвольное управление доступом;*
- *уровень B - принудительное управление доступом;*
- *уровень A - верифицируемая безопасность.*

6.2 Информационная безопасность распределенных систем. Рекомендации X.800

6.2.1 Сетевые сервисы безопасности

Переходим к рассмотрению технической спецификации X.800, появившейся немногим позднее «Оранжевой книги», но весьма полно и глубоко трактующей вопросы информационной безопасности распределенных систем. Рекомендации X.800 - документ довольно обширный. Мы остановимся на специфических сетевых функциях (сервисах) безопасности, а также на необходимых для их реализации защитных механизмах.

Выделяют следующие сервисы безопасности и исполняемые ими роли.

Аутентификация. *Данный сервис обеспечивает проверку подлинности партнеров по общению и проверку подлинности источника данных.* Она служит для предотвращения таких угроз, как маскарад и повтор предыдущего сеанса связи. Аутентификация бывает:

- односторонней (обычно клиент доказывает свою подлинность серверу),
- двусторонней (взаимной).

Управление доступом*. Обеспечивает защиту от несанкционированного использования ресурсов, доступных по сети.*

Конфиденциальность данных*. Обеспечивает защиту от несанкционированного получения информации.* Отдельно упомянем **конфиденциальность трафика** (это защита информации, которую можно получить, анализируя сетевые потоки данных).

Целостность данных *подразделяется на подвиды в зависимости от того, какой тип общения используют партнеры - с установлением*

соединения или без него, защищаются ли все данные или только отдельные поля, обеспечивается ли восстановление в случае нарушения целостности.

Неотказуемость *(невозможность отказаться от совершенных действий) обеспечивает два вида услуг:*

- *неотказуемость с подтверждением подлинности источника данных*
- *неотказуемость с подтверждением доставки.*

В таблице 5.1 указаны уровни **эталонной семиуровневой модели** OSI, на которых могут быть реализованы функции безопасности. Отметим, что прикладные процессы, в принципе, могут взять на себя поддержку всех защитных сервисов.

Таблица 5.1 - Распределение функций безопасности по уровням эталонной семиуровневой модели OSI

Функции безопасности	**Уровень**						
	1	**2**	**3**	**4**	**5**	**6**	**7**
Аутентификация	-	-	+	+	-	-	+
Управление доступом	-	-	+	+	-	-	+
Конфиденциальность соединения	+	+	+	+	-	+	+
Конфиденциальность вне соединения	-	+	+	+	-	+	+
Избирательная конфиденциальность	-	-	-	-	-	+	+
Конфиденциальность трафика	+	-	+	-	-	-	+
Целостность с восстановлением	-	-	-	+	-	-	+
Целостность без восстановления	-	-	+	+	-	-	+
Избирательная целостность	-	-	-	-	-	-	+
Целостность вне соединения	-	-	+	+	-	-	+
Неотказуемость	-	-	-	-	-	-	+

6.2.2 Сетевые механизмы безопасности

Для реализации сервисов (функций) безопасности могут использоваться следующие механизмы и их комбинации:

- ***шифрование****;*
- ***электронная цифровая подпись****;*
- ***механизмы управления доступом.*** Могут располагаться на любой из участвующих в общении сторон или в промежуточной точке;
- ***механизмы контроля целостности данных.*** В рекомендациях X.800 различаются два аспекта целостности: целостность отдельного сообщения или поля информации и целостность потока сообщений или полей информации.

- ***механизмы аутентификации.*** Согласно рекомендациям X.800, аутентификация может достигаться за счет использования паролей, , криптографических методов, устройств измерения и анализа биометрических характеристик;
- ***механизмы дополнения трафика***;
- ***механизмы управления маршрутизацией***. Маршруты могут выбираться статически или динамически. Оконечная система, зафиксировав неоднократные атаки на определенном маршруте, может отказаться от его использования. На выбор маршрута способна повлиять метка безопасности, ассоциированная с передаваемыми данными;
- ***механизмы нотаризации***. Служат для заверения таких коммуникационных характеристик, как целостность, время, личности отправителя и получателей. Заверение обеспечивается надежной третьей стороной, обладающей достаточной информацией. Обычно нотаризация опирается на механизм электронной подписи.

В таблице 5.2 сведены сервисы (функции) и механизмы безопасности. Таблица 5.2 показывает, какие механизмы (по отдельности или в комбинации с другими) могут использоваться для реализации той или иной функции.

Таблица 5.2. Взаимосвязь функций и механизмов безопасности

Функции	**Механизмы**							
	Шифрование	**Электронная подпись**	**Управление доступом**	**Целостность**	**Аутентификация**	**Дополнение трафика**	**Управление маршрутизацией**	**Нотаризация**
Аутентификация партнеров	+	+	-	-	+	-	-	-
Аутентификация источника	+	+	-	-	-	-	-	-
Управление доступом	-	-	+	-	-	-	-	-
Конфиденциальность	+	-	+	-	-	-	+	-
Избирательная конфиденциальность	+	-	-	-	-	-	-	-
Конфиденциальность трафика	+	-	-	-	-	+	+	-
Целостность соединения	+	-	-	+	-	-	-	-
Целостность вне соединения	+	+	-	+	-	-	-	-
Неотказуемость	-	+	-	+	-	-	-	+

«+» механизм пригоден для реализации данной функции безопасности;
«-» механизм не предназначен для реализации данной функции безопасности.

6.2.3 Администрирование средств безопасности

Администрирование средств безопасности *включает в себя распространение информации, необходимой для работы сервисов и механизмов безопасности, а также сбор и анализ информации об их функционировании. Концептуальной основой администрирования является информационная база управления безопасностью.* Эта база может не существовать как единое (распределенное) хранилище, но каждая из оконечных систем должна располагать информацией, необходимой для реализации избранной политики безопасности.

Согласно рекомендациям X.800, усилия администрирование средств безопасности должны распределяться по трем направлениям:

- ***администрирование информационной системы в целом****;* обеспечение актуальности политики безопасности, взаимодействие с другими административными службами, **реагирование** на происходящие события, **аудит** и **безопасное восстановление.**
- ***администрирование сервисов безопасности****;* включает в себя определение защищаемых объектов, выработку правил подбора механизмов безопасности (при наличии альтернатив), комбинирование механизмов для реализации сервисов, взаимодействие с другими администраторами для обеспечения согласованной работы.
- ***администрирование механизмов безопасности****;* определяются перечнем задействованных механизмов:
  - **управление ключами (генерация и распределение)**;
  - **управление шифрованием**;
  - **администрирование управления доступом** (распределение информации, необходимой для управления - паролей, списков доступа и т.п.);
  - **управление аутентификацией** (распределение информации, необходимой для аутентификации - паролей, ключей и т.п.);
  - **управление маршрутизацией** (выделение доверенных путей);
  - управление нотаризацией (распространение информации о нотариальных службах, администрирование этих служб).

6.3 Стандарт ISO/IEC 15408 «Общие критерии оценки безопасности информационных технологий»

6.3.1 Основные понятия

Этот международный стандарт стал итогом почти десятилетней работы специалистов нескольких стран, он вобрал в себя опыт существовавших к тому времени документов национального и межнационального масштаба. По

историческим причинам данный стандарт часто называют «Общими критериями» (или даже ОК). Мы также будем использовать это сокращение.

«Общие критерии» на самом деле являются метастандартом, определяющим инструменты оценки безопасности ИС и порядок их использования. В отличие от «Оранжевой книги», «Общие критерии» не содержат предопределенных «классов безопасности». Такие классы можно строить, исходя из **требований безопасности**, существующих для конкретной организации и/или конкретной информационной системы.

С программистской точки зрения «Общие критерии» можно считать набором библиотек, помогающих писать содержательные «программы» - ***задания по безопасности****, типовые* ***профили защиты*** *и т.п.*

Как и «Оранжевая книга», «Общие критерии» содержат два основных вида требований безопасности:

- **функциональные**, соответствующие активному аспекту защиты, предъявляемые к функциям безопасности и реализующим их механизмам;
- **требования доверия**, соответствующие пассивному аспекту, предъявляемые к технологии и процессу разработки и эксплуатации.

Требования безопасности предъявляются, а их выполнение проверяется для определенного **объекта оценки** - аппаратно-программного продукта или информационной системы.

Очень важно, что *безопасность в «Общих критериях» рассматривается не статично, а в привязке к жизненному циклу объекта оценки. Выделяются следующие этапы:*

- *определение назначения, условий применения, целей и требований безопасности;*
- *проектирование и разработка;*
- *испытания, оценка и сертификация;*
- *внедрение и эксплуатация.*

В «Общих критериях» ***объект оценки*** *рассматривается в контексте* ***среды безопасности****, которая характеризуется определенными условиями и угрозами.*

В свою очередь, угрозы характеризуются следующими параметрами:

1. *источник угрозы;*
2. *метод воздействия;*
3. *уязвимые места, которые могут быть использованы;*
4. *ресурсы (активы), которые могут пострадать.*

С точки зрения технологии программирования в «Общих критериях» использован устаревший библиотечный (не объектный) подход. Чтобы, тем не менее, структурировать пространство требований, в «Общих критериях» введена иерархия **класс-семейство-компонент-элемент**.

Классы определяют наиболее общую, «предметную» группировку требований (например, функциональные требования подотчетности).

Семейства в пределах класса различаются по строгости и другим нюансам требований.

Компонент - минимальный набор требований, фигурирующий как целое.

Элемент - неделимое требование.

Как указывалось выше, с помощью библиотек могут формироваться два вида нормативных документов:

- ***Профиль защиты (ПЗ)*** *представляет собой типовой набор требований, которым должны удовлетворять продукты и/или системы определенного класса* (например, операционные системы на компьютерах в правительственных организациях).
- ***Задание по безопасности*** *содержит совокупность требований к конкретной разработке, выполнение которых обеспечивает достижение поставленных целей безопасности*.

В «Общих критериях» нет готовых классов защиты. Сформировать классификацию в терминах «Общих критериев» - значит определить несколько иерархически упорядоченных (содержащих усиливающиеся требования) профилей защиты, в максимально возможной степени использующих стандартные функциональные требования и требования ***доверия безопасности***.

Базовый профиль защиты должен включать требования к основным (обязательным в любом случае) возможностям.

Производные профили получаются из базового путем добавления необходимых пакетов расширения, то есть подобно тому, как создаются производные классы в объектно-ориентированных языках программирования.

6.3.2 Функциональные требования

Функциональные требования сгруппированы на основе выполняемой ими роли или обслуживаемой цели безопасности. Всего в «Общих критериях» представлено 11 функциональных классов, 66 семейств, 135 компонентов. Это, конечно, значительно больше, чем число аналогичных сущностей в «Оранжевой книге».

Перечислим классы функциональных требований «Обших критериев»:

1. ***идентификация и аутентификация****;*
2. ***защита данных пользователя***;
3. ***защита функций безопасности*** (требования относятся к целостности и контролю данных сервисов безопасности и реализующих их механизмов);
4. ***управление безопасностью*** (требования этого класса относятся к управлению атрибутами и параметрами безопасности);

5. ***аудит безопасности*** (выявление, регистрация, хранение, анализ данных, затрагивающих безопасность объекта оценки, реагирование на возможное нарушение безопасности);
6. ***доступ к объекту оценки***;
7. ***приватность*** (защита пользователя от раскрытия и несанкционированного использования его идентификационных данных);
8. ***использование ресурсов*** (требования к доступности информации);
9. ***криптографическая поддержка*** (управление ключами);
10. ***связь*** (аутентификация сторон, участвующих в обмене данными);
11. ***доверенный маршрут/канал*** (для связи с сервисами безопасности).

6.3.3 Требования доверия безопасности

Установление доверия безопасности, согласно «Общим критериям», основывается на активном исследовании объекта оценки.

Форма представления требований доверия, в принципе, та же, что и для функциональных требований. Специфика состоит в том, что каждый элемент требований доверия принадлежит одному из трех типов:

- *действия **разработчиков**;*
- *представление и содержание **свидетельств**;*
- *действия **оценщиков**.*

Всего в «Общих критериях» 10 классов, 44 семейства, 93 компонента требований доверия безопасности.

Применительно к требованиям доверия в «Общих критериях» введены так называемые оценочные уровни доверия (их семь), содержащие осмысленные комбинации компонентов.

1. **Оценочный уровень доверия 1 (начальный)** предусматривает анализ **функциональной спецификации**, спецификации интерфейсов, эксплуатационной документации, а также независимое тестирование. Уровень применим, когда угрозы не рассматриваются как серьезные.
2. **Оценочный уровень доверия 2**, в дополнение к первому уровню, предусматривает наличие **проекта верхнего уровня** объекта оценки, выборочное независимое тестирование, анализ стойкости функций безопасности, поиск разработчиком явных уязвимых мест.
3. На **3 уровне** ведется контроль среды разработки и управление конфигурацией объекта оценки.
4. На **уровне 4** добавляются полная спецификация интерфейсов, **проекты нижнего уровня**, анализ подмножества реализации, применение неформальной **модели** политики безопасности, независимый анализ уязвимых мест, автоматизация управления конфигурацией. Вероятно, это самый высокий уровень, которого

можно достичь при существующей технологии программирования и приемлемых затратах.

5. **Уровень 5**, в дополнение к предыдущим, предусматривает применение формальной модели политики безопасности, полуформальных функциональной спецификации и проекта верхнего уровня с **демонстрацией соответствия** между ними.
6. На **уровне 6** реализация должна быть представлена в структурированном виде. Анализ соответствия распространяется на проект нижнего уровня.
7. Оценочный **уровень 7** (самый высокий) предусматривает формальную верификацию проекта объекта оценки. Он применим к ситуациям чрезвычайно высокого риска.

6.4 Руководящие документы Гостехкомиссии России

Гостехкомиссия России ведет весьма активную нормотворческую деятельность, выпуская Руководящие документы (РД), играющие роль национальных оценочных стандартов в области информационной безопасности. В качестве стратегического направления Гостехкомиссия России выбрала ориентацию на «Общие критерии», что можно только приветствовать.

Рассмотрим два важных, хотя и не новых, руководящих документа:

1. *Классификацию **автоматизированных систем (АС)** по уровню **защищенности от несанкционированного доступа (НСД)***
2. *Классификацию **межсетевых экранов (МЭ)**.*

Согласно первому из них, устанавливается девять классов защищенности АС от НСД к информации.

Требования ко всем девяти классам защищенности АС сведены в таблицу 5.3.

Переходя к рассмотрению второго РД Гостехкомиссии России - Классификации межсетевых экранов. Данный РД важен не столько содержанием, сколько самим фактом своего существования.

*Основным критерием классификации МЭ служит протокольный уровень (в соответствии с эталонной семиуровневой моделью), на котором осуществляется **фильтрация информации**.* Это понятно: чем выше уровень, тем больше информации на нем доступно и, следовательно, тем более тонкую и надежную фильтрацию можно реализовать.

Таблица 5.3 - Требования к защищенности автоматизированных систем

Подсистемы и требования	**Классы**								
	3Б	**3А**	**2Б**	**2А**	**1Д**	**1Г**	**1В**	**1Б**	**1А**
1. Подсистема управления доступом 1.1. Идентификация, проверка подлинности и контроль доступа субъектов: в систему;	+	+	+	+	+	+	+	+	+
к терминалам, ЭВМ, узлам сети ЭВМ, каналам связи, внешним устройствам ЭВМ;	-	-	-	+	-	+	+	+	+
к программам;	-	-	-	+	-	+	+	+	+
к томам, каталогам, файлам, записям, полям записей.	-	-	-	+	-	+	+	+	+
1.2. Управление потоками информации	-	-	-	+	-	-	+	+	+
2. Подсистема регистрации и учета 2.1. Регистрация и учет: входа/выхода субъектов доступа в/из системы (узла сети);	+	+	+	+	+	+	+	+	+
выдачи печатных (графических) выходных документов;	-	+	-	+	-	+	+	+	+
запуска/завершения программ и процессов (заданий, задач);	-	-	-	+	-	+	+	+	+
доступа программ субъектов доступа к терминалам, ЭВМ, узлам сети ЭВМ, каналам связи, внешним устройствам ЭВМ, программам, томам, каталогам, файлам, записям, полям записей;	-	-	-	+	-	+	+	+	+
изменения полномочий субъектов доступа;	-	-	-	-	-	-	+	+	+
создаваемых защищаемых объектов доступа.	-	-	-	+	-	-	+	+	+
2.2. Учет носителей информации.	+	+	+	+	+	+	+	+	+
2.3. Очистка (обнуление, обезличивание) освобождаемых областей оперативной памяти ЭВМ и внешних накопителей.	-	+	-	+	-	+	+	+	+
2.4. Сигнализация попыток нарушения защиты.	-	-	-	-	-	-	+	+	+

Таблица 5.3 - Требования к защищенности автоматизированных систем (продолжение)

Подсистемы и требования	Классы								
	3Б	**3А**	**2Б**	**2А**	**1Д**	**1Г**	**1В**	**1Б**	**1А**
3. Криптографическая подсистема 3.1. Шифрование конфиденциальной информации.	-	-	-	+	-	-	-	+	+
3.2. Шифрование информации, принадлежащей различным субъектам доступа (группам субъектов) на разных ключах.	-	-	-	-	-	-	-	-	+
3.3. Использование аттестованных (сертифицированных) криптографических средств.	-	-	-	+	-	-	-	+	+
4. Подсистема обеспечения целостности 4.1. Обеспечение целостности программных средств и обрабатываемой информации.	+	+	+	+	+	+	+	+	+
4.2. Физическая охрана средств вычислительной техники и носителей информации.	+	+	+	+	+	+	+	+	+
4.3. Наличие администратора (службы защиты) информации в АС.	-	-	-	+	-	-	+	+	+
4.4. Периодическое тестирование СЗИ НСД.	+	+	+	+	+	+	+	+	+
4.5. Наличие средств восстановления СЗИ НСД.	+	+	+	+	+	+	+	+	+
4.6. Использование сертифицированных средств защиты.	-	+	-	+	-	-	+	+	+

«-» нет требований к данному классу;
«+» есть требования к данному классу;
«СЗИ НСД» система защиты информации от несанкционированного доступа

7. АДМИНИСТРАТИВНЫЙ УРОВЕНЬ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

7.1 Основные понятия административного уровня информационной безопасности

К административному уровню информационной безопасности относятся действия общего характера, предпринимаемые руководством организации.

Главная цель мер административного уровня *- сформировать* ***программу*** *работ в области информационной безопасности и обеспечить ее выполнение, выделяя необходимые ресурсы и контролируя состояние дел.*

Основой программы *является* ***политика безопасности***, отражающая подход организации к защите своих информационных активов. Руководство каждой организации должно осознать необходимость поддержания режима безопасности и выделения на эти цели значительных ресурсов.

Политика безопасности строится на основе анализа рисков, которые признаются реальными для информационной системы организации. Когда риски проанализированы и стратегия защиты определена, составляется программа обеспечения информационной безопасности.

«Политика безопасности» (является не совсем точным переводом английского словосочетания «security policy»), имеет в виду не отдельные правила или их наборы, а стратегию организации в области информационной безопасности.

Политика безопасности *- совокупность документированных решений, принимаемых руководством организации и направленных на защиту информации и ассоциированных с ней ресурсов.*

Чтобы рассматривать ИС предметно, с использованием актуальных данных, следует составить карту информационной системы. Эта карта, разумеется, должна быть изготовлена в объектно-ориентированном стиле, с возможностью варьировать не только уровень детализации, но и видимые грани объектов. Техническим средством составления, сопровождения и визуализации подобных карт может служить свободно распространяемый каркас какой-либо системы управления.

7.2 Политика безопасности

Политика безопасности определяет архитектуру системы защиты и реализуется посредством:

- *административно-организационных мер,*
- *физических и программно-технических средств.*

Для конкретной организации политика безопасности должна носить индивидуальный характер и зависеть от конкретной технологии обработки информации и используемых программных и технических средств.

Политика безопасности определяется способом управления доступом, определяющим порядок доступа к объектам системы.

Различают два основных вида политики безопасности.

- ***Избирательная политика безопасности*** *основана на избирательном способе управления доступом и характеризуется заданным администратором множеством разрешенных отношений доступа* (например, в виде троек «объект, субъект, тип доступа»). Обычно для описания свойств избирательного управления доступом применяют математическую модель на основе матрицы доступа.
- ***Полномочная политика безопасности*** *основана на полномочном (мандатном) способе управления доступом и характеризуется совокупностью правил предоставления доступа, определенных на множестве атрибутов безопасности субъектов и объектов*, например, в зависимости от метки конфиденциальности информации и уровня допуска пользователя. ***Полномочное управление доступом подразумевает, что:***
  - все субъекты и объекты системы однозначно идентифицированы;
  - каждому объекту системы присвоена метка конфиденциальности информации, определяющая ценность содержащейся в нем информации;
  - каждому субъекту системы присвоен определенный уровень допуска, определяющий максимальное значение метки конфиденциальности информации объектов, к которым субъект имеет доступ.

С практической точки зрения политику безопасности целесообразно рассматривать на трех уровнях детализации.

1. ***Верхний уровень***. *К верхнему уровню можно отнести решения, затрагивающие организацию в целом.* Примерный список подобных решений может включать в себя следующие элементы:

- решение сформировать или пересмотреть комплексную программу обеспечения информационной безопасности, назначение ответственных за продвижение программы;
- формулировка целей, которые преследует организация в области информационной безопасности, определение общих направлений в достижении этих целей;
- обеспечение базы для соблюдения законов и правил;

- формулировка административных решений по тем вопросам реализации программы безопасности, которые должны рассматриваться на уровне организации в целом.

На верхний уровень выносится управление защитными ресурсами и координация использования этих ресурсов, выделение специального персонала для защиты критически важных систем и взаимодействие с другими организациями, обеспечивающими или контролирующими режим безопасности.

Политика верхнего уровня имеет дело с тремя аспектами законопослушности и исполнительской дисциплины.

- Во-первых, организация должна соблюдать существующие законы.
- Во-вторых, следует контролировать действия лиц, ответственных за выработку программы безопасности.
- Наконец, необходимо обеспечить определенную степень исполнительности персонала, а для этого нужно выработать систему поощрений и наказаний.

2. *Средний уровень.* *К данному уровню можно отнести вопросы, касающиеся отдельных аспектов информационной безопасности, но важные для различных эксплуатируемых организацией систем.* Политика среднего уровня должна для каждого аспекта освещать следующие темы:

- **Описание аспекта.** Например, если рассмотреть применение пользователями неофициального программного обеспечения, последнее можно определить как ПО, которое не было одобрено и/или закуплено на уровне организации.
- **Область применения.** Следует определить, где, когда, как, по отношению к кому и чему применяется данная политика безопасности. Например, касается ли политика, связанная с использованием неофициального программного обеспечения, организаций-субподрядчиков? Затрагивает ли она сотрудников, пользующихся портативными и домашними компьютерами и вынужденных переносить информацию на производственные машины?
- **Позиция организации по данному аспекту.** Продолжая пример с неофициальным программным обеспечением, можно представить себе позиции полного запрета, выработки процедуры приемки подобного ПО и т.п. Позиция может быть сформулирована и в гораздо более общем виде, как набор целей, которые преследует организация в данном аспекте.
- **Роли и обязанности.** Например, если для использования неофициального программного обеспечения сотрудникам требуется разрешение руководства, должно быть известно, у кого и как его можно получить. Если неофициальное программное обеспечение

использовать нельзя, следует знать, кто следит за выполнением данного правила.

- **Законопослушность.** Политика должна содержать общее описание запрещенных действий и наказаний за них.
- **Точки контакта.** Должно быть известно, куда следует обращаться за разъяснениями, помощью и дополнительной информацией. Обычно «точкой контакта» служит определенное должностное лицо, а не конкретный человек, занимающий в данный момент данный пост.

3. *Политика безопасности **нижнего уровня** относится к конкретным информационным сервисам. Она включает в себя два аспекта - цели и правила их достижения, поэтому ее порой трудно отделить от вопросов реализации.* В отличие от двух верхних уровней, рассматриваемая политика должна быть определена более подробно. Есть много вещей, специфичных для отдельных видов услуг, которые нельзя единым образом регламентировать в рамках всей организации. В то же время, эти вещи настолько важны для обеспечения режима безопасности, что относящиеся к ним решения должны приниматься на управленческом, а не техническом уровне.

Приведем несколько примеров вопросов, на которые следует дать ответ в политике безопасности нижнего уровня:

- кто имеет право доступа к объектам, поддерживаемым сервисом?
- при каких условиях можно читать и модифицировать данные?
- как организован удаленный доступ к сервису?

При формулировке целей политики нижнего уровня можно исходить из соображений целостности, доступности и конфиденциальности, но нельзя на этом останавливаться. Ее цели должны быть более конкретными. Например, если речь идет о системе расчета заработной платы, можно поставить цель, чтобы только сотрудникам отдела кадров и бухгалтерии позволялось вводить и модифицировать информацию. В более общем случае цели должны связывать между собой объекты сервиса и действия с ними.

7.3 Программа безопасности

После того, как сформулирована политика безопасности, можно приступать к составлению программы ее реализации и собственно к реализации.

Чтобы понять и реализовать какую-либо программу, ее нужно структурировать по уровням, обычно в соответствии со структурой организации. ***В простейшем и самом распространенном случае достаточно двух уровней –***

1. ***верхнего***, или центрального, который охватывает всю организацию,

2. ***нижнего***, или служебного, который относится к отдельным услугам или группам однородных сервисов.

Программу верхнего уровня возглавляет лицо, отвечающее за информационную безопасность организации. У этой программы следующие главные цели:

- управление рисками (оценка рисков, выбор эффективных средств защиты);
- координация деятельности в области информационной безопасности, пополнение и распределение ресурсов;
- стратегическое планирование;
- контроль деятельности в области информационной безопасности.

*В рамках **программы верхнего уровня** принимаются стратегические решения по обеспечению безопасности, оцениваются технологические новинки.* Информационные технологии развиваются очень быстро, и необходимо иметь четкую политику отслеживания и внедрения новых средств.

*Цель **программы нижнего уровня** - обеспечить надежную и экономичную защиту конкретного сервиса или группы однородных сервисов. На этом уровне решается, какие следует использовать механизмы защиты; закупаются и устанавливаются технические средства; выполняется повседневное администрирование; отслеживается состояние слабых мест и т.п.* Обычно за программу нижнего уровня отвечают администраторы сервисов.

7.4 Синхронизация программы безопасности с жизненным циклом систем

Если синхронизировать программу безопасности нижнего уровня с жизненным циклом защищаемого сервиса, можно добиться большего эффекта с меньшими затратами. ***В жизненном цикле информационного сервиса можно выделить следующие этапы:***

1. ***Инициация.*** *На данном этапе выявляется необходимость в приобретении нового сервиса, документируется его предполагаемое назначение.*
2. ***Закупка.*** *На данном этапе составляются спецификации, прорабатываются варианты приобретения, выполняется собственно закупка.*
3. ***Установка.*** *Сервис устанавливается, конфигурируется, тестируется и вводится в эксплуатацию.*
4. ***Эксплуатация.*** *На данном этапе сервис не только работает и администрируется, но и подвергается модификациям.*
5. ***Выведение из эксплуатации.*** *Происходит переход на новый сервис.*

Рассмотрим действия, выполняемые на каждом из этапов, более подробно.

На этапе инициации оформляется понимание того, что необходимо приобрести новый или значительно модернизировать существующий сервис; определяется, какими характеристиками и какой функциональностью он должен обладать; оцениваются финансовые и иные ограничения.

Требуется сформулировать ответы на следующие вопросы:

1. какого рода информация предназначается для обслуживания новым сервисом?
2. каковы возможные последствия нарушения конфиденциальности, целостности и доступности этой информации?
3. каковы угрозы, по отношению к которым сервис и информация будут наиболее уязвимы?
4. есть ли какие-либо особенности нового сервиса (например, территориальная распределенность компонентов), требующие принятия специальных процедурных мер?
5. каковы характеристики персонала, имеющие отношение к безопасности (квалификация, благонадежность)?
6. каковы законодательные положения и внутренние правила, которым должен соответствовать новый сервис?

Этап закупки - один из самых сложных. Нужно окончательно сформулировать требования к защитным средствам нового сервиса, к компании, которая может претендовать на роль поставщика, и к квалификации, которой должен обладать персонал, использующий или обслуживающий закупаемый продукт. Все эти сведения оформляются в виде спецификации, куда входят не только аппаратура и программы, но и документация, обслуживание, обучение персонала. Разумеется, особое внимание должно уделяться вопросам совместимости нового сервиса с существующей конфигурацией. Нередко средства безопасности являются необязательными компонентами коммерческих продуктов, и нужно проследить, чтобы соответствующие пункты не выпали из спецификации.

Установка. Когда продукт закуплен, его необходимо установить. Несмотря на кажущуюся простоту, установка является очень ответственным делом:

- Во-первых, новый продукт следует сконфигурировать. Как правило, коммерческие продукты поставляются с отключенными средствами безопасности; их необходимо включить и должным образом настроить.
- Во-вторых, новый сервис нуждается в процедурных регуляторах. Следует позаботиться о чистоте и охране помещения, о документах, регламентирующих использование сервиса, о подготовке планов на случай экстренных ситуаций, об организации обучения пользователей и т.п.

- Тестирование - проводится после принятия перечисленных мер необходимо провести. Его полнота и комплексность могут служить гарантией безопасности эксплуатации в штатном режиме.

Период эксплуатации - самый длительный и сложный. С психологической точки зрения наибольшую опасность в это время представляют незначительные изменения в конфигурации сервиса, в поведении пользователей и администраторов. Если безопасность не поддерживать, она ослабевает. Для борьбы с эффектом медленных изменений приходится прибегать к периодическим проверкам безопасности сервиса. Разумеется, после значительных модификаций подобные проверки являются обязательными.

При **выведении из эксплуатации** затрагиваются аппаратно-программные компоненты сервиса и обрабатываемые им данные. Аппаратура продается, утилизируется или выбрасывается. Только в специфических случаях необходимо заботиться о физическом разрушении аппаратных компонентов, хранящих конфиденциальную информацию. Программы, вероятно, просто стираются, если иное не предусмотрено лицензионным соглашением.

При выведении данных из эксплуатации их обычно переносят на другую систему, архивируют, выбрасывают или уничтожают. Если архивирование производится с намерением впоследствии прочитать данные в другом месте, следует позаботиться об аппаратно-программной совместимости средств чтения и записи.

7.5 Понятие об управлении рисками

Управление рисками рассматривается нами на административном уровне ИБ, поскольку только руководство организации способно выделить необходимые ресурсы, инициировать и контролировать выполнение соответствующих программ.

Использование информационных систем связано с определенной совокупностью рисков. Когда возможный ущерб неприемлемо велик, необходимо принять экономически оправданные меры защиты. Периодическая (пере)оценка рисков необходима для контроля эффективности деятельности в области безопасности и для учета изменений обстановки.

Уровень риска *является количественной функцией вероятности реализации определенной угрозы (использующей некоторые уязвимые места), а также величины возможного ущерба.*

Суть мероприятий по управлению рисками *состоит в том, чтобы оценить их размер, выработать эффективные и экономичные меры снижения рисков, а затем убедиться, что риски заключены в приемлемые*

рамки (и остаются таковыми). Следовательно, управление рисками включает в себя два вида деятельности, которые чередуются циклически:

1. (пере)оценка (измерение) рисков;
2. выбор эффективных и экономичных защитных средств (нейтрализация рисков).

По отношению к выявленным рискам возможны следующие действия:

- ликвидация риска (например, за счет устранения причины);
- уменьшение риска (например, за счет использования дополнительных защитных средств);
- принятие риска (и выработка плана действия в соответствующих условиях);
- переадресация риска (например, путем заключения страхового соглашения).

Процесс управления рисками можно разделить на следующие этапы:

1. *Выбор анализируемых объектов и уровня детализации их рассмотрения.* Для небольшой организации допустимо рассматривать всю информационную инфраструктуру; однако если организация крупная, всеобъемлющая оценка может потребовать неприемлемых затрат времени и сил. В таком случае следует сосредоточиться на наиболее важных сервисах, заранее соглашаясь с приближенностью итоговой оценки.
2. *Выбор методологии оценки рисков.* Целью оценки является получение ответа на два вопроса: приемлемы ли существующие риски, и если нет, то какие защитные средства стоит использовать. Значит, оценка должна быть количественной, допускающей сопоставление с заранее выбранными границами допустимости и расходами на реализацию новых регуляторов безопасности.
3. *Идентификация активов.* При идентификации активов, то есть тех ресурсов и ценностей, которые организация пытается защитить, следует, конечно, учитывать не только компоненты информационной системы, но и поддерживающую инфраструктуру, персонал, а также нематериальные ценности, такие как репутация организации.
4. *Анализ угроз и их последствий, выявление уязвимых мест в защите.* Первый шаг в анализе угроз - их идентификация. Целесообразно выявлять не только сами угрозы, но и **источники** их возникновения - это поможет в выборе дополнительных средств защиты. После идентификации угрозы необходимо оценить вероятность ее осуществления. Кроме вероятности осуществления, важен размер потенциального ущерба. Оценивая размер ущерба, необходимо иметь в виду не только непосредственные расходы на замену оборудования или восстановление информации, но и более

отдаленные, такие как подрыв репутации, ослабление позиций на рынке и т.п.

5. *Оценка рисков.* После того, как накоплены исходные данные и оценена степень неопределенности, можно переходить к обработке информации, то есть собственно к оценке рисков. Вполне допустимо применить такой простой метод, как умножение вероятности осуществления угрозы на предполагаемый ущерб.
6. *Выбор защитных мер.* Как правило, для ликвидации или нейтрализации уязвимого места, сделавшего угрозу реальной, существует несколько механизмов безопасности, различных по эффективности и стоимости. Оценивая стоимость мер защиты, приходится, разумеется, учитывать не только прямые расходы на закупку оборудования и/или программ, но и расходы на внедрение новинки и, в частности, обучение и переподготовку персонала. Важным обстоятельством является совместимость нового средства со сложившейся организационной и аппаратно-программной структурой, с традициями организации.
7. *Реализация и проверка выбранных мер.* Когда намеченные меры приняты, необходимо проверить их действенность, то есть убедиться, что остаточные риски стали приемлемыми. Если это на самом деле так, значит, можно спокойно намечать дату ближайшей переоценки.
8. *Оценка остаточного риска.* Если остаточные риски не соответствуют заданным необходимо проанализировать допущенные ошибки и провести повторный сеанс управления рисками немедленно.

8. ПРОЦЕДУРНЫЙ УРОВЕНЬ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

8.1. Основные классы мер процедурного уровня

Рассмотрим меры безопасности, которые ориентированы на людей, а не на технические средства. Именно люди формируют режим информационной безопасности, и они же оказываются главной угрозой, поэтому «человеческий фактор» заслуживает особого внимания.

Следует осознать ту степень зависимости от компьютерной обработки данных, в которую попало современное общество. Акцент следует делать не на военной или криминальной стороне дела, а на гражданских аспектах, связанных с поддержанием нормального функционирования аппаратного и программного обеспечения, то есть концентрироваться на вопросах доступности и целостности данных.

На процедурном уровне можно выделить следующие классы мер:

- *управление персоналом;*
- *физическая защита;*
- *поддержание работоспособности;*
- *реагирование на нарушения режима безопасности;*
- *планирование восстановительных работ.*

Рассмотрим меры процедурного уровня более подробно.

8.2 Управление персоналом

Управление персоналом начинается с приема нового сотрудника на работу и даже раньше - с составления описания должности. Уже на данном этапе желательно подключить к работе специалиста по информационной безопасности для определения компьютерных привилегий, ассоциируемых с должностью.

Существует два общих принципа, которые следует иметь в виду при управлении персоналом:

- ***Принцип разделения обязанностей*** *предписывает так распределять роли и ответственность, чтобы один человек не мог нарушить критически важный для организации процесс.*
- ***Принцип минимизации привилегий*** *предписывает выделять пользователям только те права доступа, которые необходимы им для выполнения служебных обязанностей.* Назначение этого принципа очевидно - уменьшить ущерб от случайных или умышленных некорректных действий.

Когда кандидат определен, он, вероятно, должен пройти обучение, по крайней мере, его следует подробно ознакомить со служебными

обязанностями, а также с нормами и процедурами информационной безопасности. Желательно, чтобы меры безопасности были им усвоены до вступления в должность и до заведения его системного счета с входным именем, паролем и привилегиями.

С момента заведения системного счета начинается его администрирование, а также протоколирование и анализ действий пользователя. Постепенно изменяется окружение, в котором работает пользователь, его служебные обязанности и т.п. Все это требует соответствующего изменения привилегий.

Ликвидация системного счета пользователя, особенно в случае конфликта между сотрудником и организацией, должна производиться максимально оперативно (в идеале - одновременно с извещением о наказании или увольнении). Возможно и физическое ограничение доступа к рабочему месту. Разумеется, если сотрудник увольняется, у него нужно принять все его компьютерное хозяйство и, в частности, криптографические ключи, если использовались средства шифрования.

Проблема обучения *- одна из основных с точки зрения информационной безопасности. Если сотрудник не знаком с политикой безопасности своей организации, он не может стремиться к достижению сформулированных в ней целей. Не зная мер безопасности, он не сможет их соблюдать.* Напротив, если сотрудник знает, что его действия протоколируются, он, возможно, воздержится от нарушений.

8.3 Физическая защита

Безопасность информационной системы зависит от окружения, в котором она функционирует. Необходимо принять меры для защиты зданий и прилегающей территории, поддерживающей инфраструктуры, вычислительной техники, носителей данных.

Основной принцип физической защиты, соблюдение которого следует постоянно контролировать, формулируется как «непрерывность защиты в пространстве и времени».

Выделяют следующие основные направления обеспечения физической защиты:

- *физическое управление доступом;*
- *противопожарные меры;*
- *защита поддерживающей инфраструктуры;*
- *защита от перехвата данных;*
- *защита мобильных систем.*

8.4 Поддержание работоспособности

Недооценка факторов безопасности в повседневной работе - ахиллесова пята многих организаций. Дорогие средства безопасности теряют

смысл, если они плохо документированы, конфликтуют с другим программным обеспечением, а пароль системного администратора не менялся с момента установки. Нечаянные ошибки системных администраторов и пользователей грозят повреждением аппаратуры, разрушением программ и данных; в лучшем случае они создают бреши в защите, которые делают возможной реализацию угроз.

Можно выделить следующие направления повседневной деятельности направленных на поддержание работоспособности:

- *поддержка пользователей;*
- *поддержка программного обеспечения;*
- *конфигурационное управление;*
- *резервное копирование;*
- *управление носителями;*
- *документирование;*
- *регламентные работы.*

8.5 Реагирование на нарушения режима безопасности

Программа безопасности, принятая организацией, должна предусматривать набор оперативных мероприятий, направленных на обнаружение и нейтрализацию нарушений режима информационной безопасности. Важно, чтобы в подобных случаях последовательность действий была спланирована заранее, поскольку меры нужно принимать срочные и скоординированные.

Реакция на нарушения режима безопасности преследует три главные цели:

- *локализация инцидента и уменьшение наносимого вреда;*
- *выявление нарушителя;*
- *предупреждение повторных нарушений.*

В организации должен быть человек, доступный 24 часа в сутки (лично, по телефону, пейджеру или электронной почте), который отвечает за реакцию на нарушения. Все должны знать координаты этого человека и обращаться к нему при первых признаках опасности.

Нередко требование локализации инцидента и уменьшения наносимого вреда вступает в конфликт с желанием выявить нарушителя. В политике безопасности организации приоритеты должны быть расставлены заранее. Поскольку, как показывает практика, выявить злоумышленника очень сложно, на наш взгляд, в первую очередь следует заботиться об уменьшении ущерба.

Чтобы найти нарушителя, нужно заранее выяснить контактные координаты поставщика сетевых услуг и договориться с ним о самой возможности и порядке выполнения соответствующих действий.

Необходимо отслеживать появление новых уязвимых мест и как можно быстрее ликвидировать ассоциированные с ними окна опасности. Кто-то в организации должен курировать этот процесс, принимать краткосрочные меры и корректировать программу безопасности для принятия долгосрочных мер.

8.6 Планирование восстановительных работ

Ни одна организация не застрахована от серьезных аварий, вызванных естественными причинами, действиями злоумышленника, халатностью или некомпетентностью.

Планирование восстановительных работ *позволяет подготовиться к авариям, уменьшить ущерб от них и сохранить способность к функционированию хотя бы в минимальном объеме.*

Отметим, что меры информационной безопасности можно разделить на три группы, в зависимости от того на какой аспект они направлены на:

- предупреждение атак;
- обнаружение атак;
- ликвидацию последствий атак.

Планирование восстановительных работ, очевидно, относится к последней из трех перечисленных групп.

Процесс планирования восстановительных работ можно разделить на следующие этапы:

- *выявление критически важных функций организации, установление приоритетов;*
- *идентификация ресурсов, необходимых для выполнения критически важных функций;*
- *определение перечня возможных аварий;*
- *разработка стратегии восстановительных работ;*
- *подготовка к реализации выбранной стратегии;*
- *проверка стратегии.*

Планируя восстановительные работы, следует отдавать себе отчет в том, что полностью сохранить функционирование организации не всегда возможно. Необходимо выявить критически важные функции, без которых организация теряет свое лицо, и даже среди критичных функций расставить приоритеты, чтобы как можно быстрее и с минимальными затратами возобновить работу после аварии.

Критичные ресурсы обычно относятся к одной из следующих категорий:

- персонал;
- информационная инфраструктура;
- физическая инфраструктура.

При определении перечня возможных аварий нужно попытаться разработать их сценарии. Как будут развиваться события? Каковы могут оказаться масштабы бедствия? Что произойдет с критичными ресурсами? Например, смогут ли сотрудники попасть на работу? Будут ли выведены из строя компьютеры? Возможны ли случаи саботажа? Будет ли работать связь? Пострадает ли здание организации? Можно ли будет найти и прочитать необходимые бумаги?

Стратегия восстановительных работ должна базироваться на наличных ресурсах и быть не слишком накладной для организации. При разработке стратегии целесообразно провести анализ рисков, которым подвергаются критичные функции, и попытаться выбрать наиболее экономичное решение.

Стратегия должна предусматривать не только работу по временной схеме, но и возвращение к нормальному функционированию.

Подготовка к реализации выбранной стратегии состоит в выработке плана действий в экстренных ситуациях и по их окончании, а также в обеспечении некоторой избыточности критичных ресурсов. Избыточность обеспечивается также мерами резервного копирования, хранением копий в нескольких местах, представлением информации в разных видах (на бумаге и в файлах) и т.д.

9. ОСНОВНЫЕ ПРОГРАММНО-ТЕХНИЧЕСКИЕ МЕРЫ ОБЕСПЕЧЕНИЯ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

9.1 Основные понятия программно-технического уровня информационной безопасности

***Программно-технические меры**, то есть меры, направленные на контроль компьютерных сущностей - оборудования, программ и/или данных, образуют последний и самый важный рубеж информационной безопасности*. Напомним, что ущерб наносят в основном действия легальных пользователей, по отношению к которым процедурные регуляторы малоэффективны. Главные враги - некомпетентность и неаккуратность при выполнении служебных обязанностей, и только программно-технические меры способны им противостоять.

Следует, учитывать, что быстрое развитие информационных технологий не только предоставляет обороняющимся новые возможности, но и объективно затрудняет обеспечение надежной защиты, если опираться исключительно на меры программно-технического уровня. Причин тому несколько:

- повышение быстродействия микросхем, развитие архитектур с высокой степенью параллелизма позволяет методом грубой силы преодолевать барьеры (прежде всего криптографические), ранее казавшиеся неприступными;
- развитие сетей и сетевых технологий, увеличение числа связей между информационными системами, рост пропускной способности каналов расширяют круг злоумышленников, имеющих техническую возможность организовывать атаки;
- появление новых информационных сервисов ведет и к образованию новых уязвимых мест как «внутри» сервисов, так и на их стыках;
- конкуренция среди производителей программного обеспечения заставляет сокращать сроки разработки, что приводит к снижению качества тестирования и выпуску продуктов с дефектами защиты;
- навязываемая потребителям парадигма постоянного наращивания мощности аппаратного и программного обеспечения не позволяет долго оставаться в рамках надежных, апробированных конфигураций и, кроме того, вступает в конфликт с бюджетными ограничениями, из-за чего снижается доля ассигнований на безопасность.

Перечисленные соображения лишний раз подчеркивают важность комплексного подхода к информационной безопасности, а также необходимость гибкой позиции при выборе и сопровождении программно-технических регуляторов.

Центральным для программно-технического уровня является понятие ***сервиса безопасности****.*

Следуя объектно-ориентированному подходу, при рассмотрении информационной системы с единичным уровнем детализации мы увидим совокупность предоставляемых ею информационных сервисов.

Основными сервисами обеспечения безопасности являются:

- *идентификация и аутентификация;*
- *управление доступом;*
- *протоколирование и аудит;*
- *шифрование;*
- *контроль целостности;*
- *экранирование;*
- *анализ защищенности;*
- *обеспечение отказоустойчивости;*
- *обеспечение безопасного восстановления;*
- *туннелирование;*
- *управление.*

Считается, что данного набора сервисов, в принципе, достаточно для построения надежной защиты на программно-техническом уровне, правда, при соблюдении целого ряда дополнительных условий (отсутствие уязвимых мест, безопасное администрирование и т.д.).

Для проведения классификации сервисов безопасности и определения их места в общей архитектуре ***меры безопасности можно разделить на следующие виды:***

- *превентивные, препятствующие нарушениям ИБ;*
- *меры обнаружения нарушений;*
- *локализующие, сужающие зону воздействия нарушений;*
- *меры по выявлению нарушителя;*
- *меры восстановления режима безопасности.*

Большинство сервисов безопасности попадает в число превентивных, и это, безусловно, правильно. Аудит и контроль целостности способны помочь в обнаружении нарушений; активный аудит, кроме того, позволяет запрограммировать реакцию на нарушение с целью локализации и/или прослеживания. Направленность сервисов отказоустойчивости и безопасного восстановления очевидна. Наконец, управление играет инфраструктурную роль, обслуживая все аспекты ИС.

9.2 Особенности современных информационных систем, существенные при обеспечении информационной безопасности

Информационная система типичной современной организации является весьма сложным образованием, построенным в многоуровневой архитектуре, которое пользуется многочисленными внешними сервисами и, в свою очередь, предоставляет собственные сервисы вовне.

Особенности современных ИС наиболее существенные с точки зрения обеспечения безопасности:

- ***корпоративная сеть*** *имеет несколько территориально разнесенных частей* (поскольку организация располагается на нескольких производственных площадках), *связи между которыми находятся в ведении внешнего поставщика сетевых услуг, выходя за пределы зоны, контролируемой организацией*;
- *корпоративная сеть имеет одно или несколько* ***подключений к Internet***;
- *на каждой из производственных площадок могут находиться критически важные серверы, в доступе к которым нуждаются сотрудники, работающие на других площадках*, мобильные пользователи и, возможно, сотрудники других организаций;
- *для доступа пользователей могут применяться не только компьютеры, но и потребительские устройства, использующие, в частности, беспроводную связь*;
- *в течение одного сеанса работы пользователю приходится обращаться к нескольким информационным сервисам*, опирающимся на разные аппаратно-программные платформы;
- *к* ***доступности*** *информационных сервисов предъявляются жесткие требования, которые обычно выражаются в необходимости круглосуточного функционирования* с максимальным временем простоя порядка нескольких минут;
- *информационная система представляет собой сеть с* ***активными агентами***, то есть в процессе работы программные компоненты, такие как **апплеты** или **сервлеты**, передаются с одной машины на другую и выполняются в целевой среде, поддерживая связь с удаленными компонентами;
- *не все пользовательские системы контролируются сетевыми и/или системными администраторами организации*;
- *программное обеспечение, не может считаться надежным, в нем могут быть ошибки, создающие проблемы в защите*;
- *конфигурация информационной системы постоянно изменяется на уровнях административных данных, программ и аппаратуры*

(меняется состав пользователей, их привилегии и версии программ, появляются новые сервисы, новая аппаратура и т.п.).

Следует также учитывать еще по крайней мере два момента.

- *Во-первых, для каждого сервиса основные грани ИБ (доступность, целостность, конфиденциальность) трактуются по-своему*. Целостность с точки зрения системы управления базами данных и с точки зрения почтового сервера - вещи принципиально разные.
- *Во-вторых, основная угроза информационной безопасности организаций по-прежнему исходит не от внешних злоумышленников, а от собственных сотрудников.*

9.3 Архитектура системы безопасности

Сервисы безопасности, какими бы мощными они ни были, сами по себе не могут гарантировать надежность программно-технического уровня защиты. Только проверенная архитектура способна сделать эффективным объединение сервисов, обеспечить управляемость информационной системы, ее способность развиваться и противостоять новым угрозам при сохранении таких свойств, как высокая производительность, простота и удобство использования.

Теоретической основой решения проблемы архитектурной безопасности является следующее фундаментальное утверждение.

«Пусть каждый субъект (то есть процесс, действующий от имени какого-либо пользователя) заключен внутри одного компонента и может осуществлять непосредственный доступ к объектам только в пределах этого компонента. Далее пусть каждый компонент содержит свой монитор обращений, отслеживающий все локальные попытки доступа, и все мониторы проводят в жизнь согласованную политику безопасности. Пусть, наконец, коммуникационные каналы, связывающие компоненты, сохраняют конфиденциальность и целостность передаваемой информации. Тогда совокупность всех мониторов образует единый монитор обращений для всей сетевой конфигурации» (см. рис 9.1).

Обратим внимание на три принципа, содержащиеся в приведенном утверждении:

- *необходимость выработки и проведения в жизнь единой политики безопасности;*
- *необходимость обеспечения конфиденциальности и целостности при сетевых взаимодействиях;*
- *необходимость формирования составных сервисов по содержательному принципу, чтобы каждый полученный таким образом компонент обладал полным набором защитных средств и с внешней точки зрения представлял собой единое целое (не должно быть информационных потоков, идущих к незащищенным сервисам).*

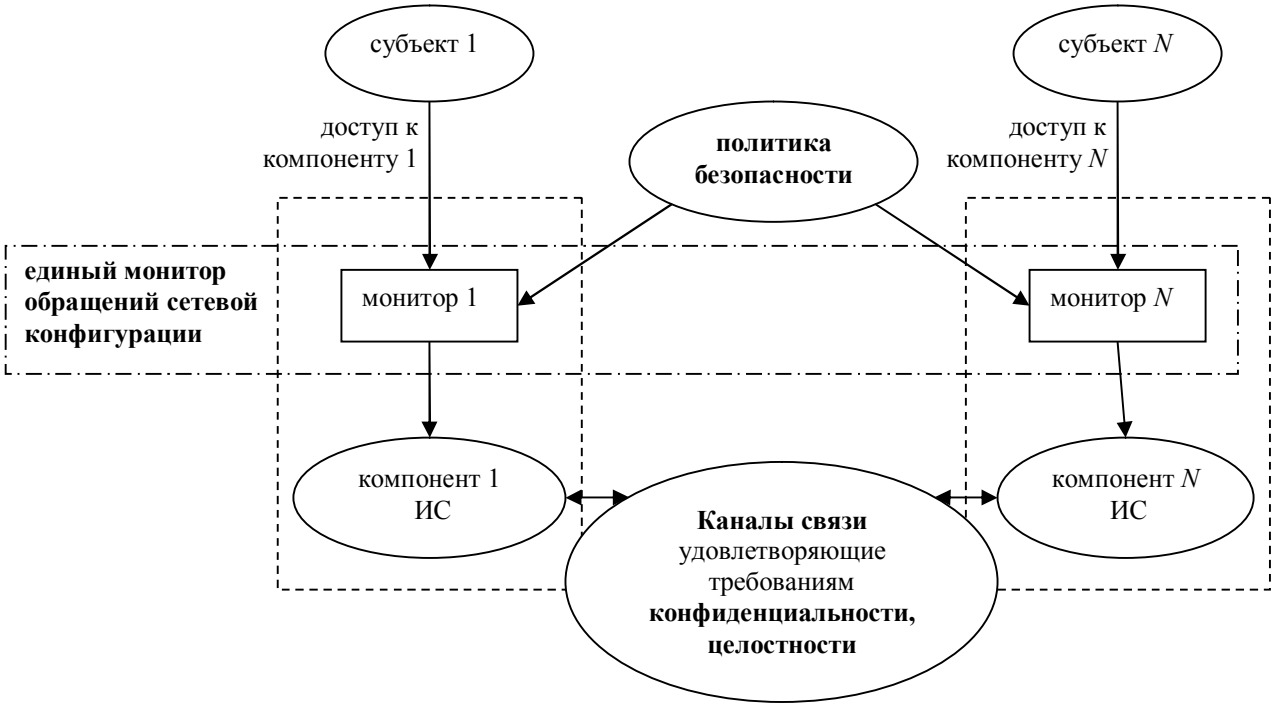


Рис. 9.1 – Архитектура единой системы информационной безопасности

Если какой-либо (составной) сервис не обладает полным набором защитных средств (состав полного набора описан выше), необходимо привлечение дополнительных сервисов, которые мы будем называть экранирующими. Экранирующие сервисы устанавливаются на путях доступа к недостаточно защищенным элементам; в принципе, один такой сервис может экранировать (защищать) сколь угодно большое число элементов.

С практической точки зрения наиболее важными являются следующие принципы архитектурной безопасности:

- ***непрерывность защиты*** *в пространстве и времени, невозможность миновать защитные средства;*
- *следование признанным стандартам, использование апробированных решений;*
- *иерархическая организация ИС с небольшим числом сущностей на каждом уровне;*
- *усиление самого* ***слабого звена****;*
- *невозможность перехода в* ***небезопасное состояние****;*
- *минимизация привилегий;*
- *разделение обязанностей;*
- ***эшелонированность обороны****;*
- *разнообразие защитных средств;*
- *простота и управляемость информационной системы.*

Поясним смысл перечисленных принципов.

Если у злоумышленника или недовольного пользователя появится возможность миновать защитные средства, он, разумеется, так и сделает.

Определенные выше **экранирующие сервисы** должны исключить подобную возможность.

Следование признанным стандартам и использование апробированных решений повышает надежность ИС и уменьшает вероятность попадания в тупиковую ситуацию, когда обеспечение безопасности потребует непомерно больших затрат и принципиальных модификаций.

Иерархическая организация ИС с небольшим числом сущностей на каждом уровне необходима по технологическим соображениям. При нарушении данного принципа система станет неуправляемой и, следовательно, обеспечить ее безопасность будет невозможно.

Надежность любой обороны определяется самым слабым звеном. Злоумышленник не будет бороться против силы, он предпочтет легкую победу над слабостью. (Часто самым слабым звеном оказывается не компьютер или программа, а человек, и тогда проблема обеспечения информационной безопасности приобретает нетехнический характер.)

Принцип невозможности перехода в небезопасное состояние означает, что при любых обстоятельствах, в том числе нештатных, защитное средство либо полностью выполняет свои функции, либо полностью блокирует доступ. Образно говоря, если в крепости механизм подъемного моста ломается, мост оставляют поднятым, препятствуя проходу неприятеля.

Применительно к программно-техническому уровню **принцип минимизации привилегий** предписывает выделять пользователям и администраторам только те права доступа, которые необходимы им для выполнения служебных обязанностей. Этот принцип позволяет уменьшить ущерб от случайных или умышленных некорректных действий пользователей и администраторов.

Принцип разделения обязанностей предполагает такое распределение ролей и ответственности, чтобы один человек не мог нарушить критически важный для организации процесс или создать брешь в защите по заказу злоумышленников. В частности, соблюдение данного принципа особенно важно, чтобы предотвратить злонамеренные или неквалифицированные действия системного администратора.

Принцип эшелонированности обороны предписывает не полагаться на один защитный рубеж, каким бы надежным он ни казался. За средствами физической защиты должны следовать программно-технические средства, за идентификацией и аутентификацией - управление доступом и, как последний рубеж, - протоколирование и аудит. Эшелонированная оборона способна, по крайней мере, задержать злоумышленника, а благодаря наличию такого рубежа, как протоколирование и аудит, его действия не останутся незамеченными. Принцип разнообразия защитных средств предполагает создание различных по своему характеру оборонительных рубежей, чтобы от потенциального злоумышленника требовалось овладение разнообразными и, по возможности, несовместимыми между собой навыками.

Очень важен **принцип простоты и управляемости информационной системы** в целом и защитных средств в особенности. Только для простого защитного средства можно формально или неформально доказать его корректность. Только в простой и управляемой системе можно проверить согласованность конфигурации различных компонентов и осуществлять централизованное администрирование.

Для обеспечения высокой доступности (непрерывности функционирования) необходимо соблюдать следующие принципы архитектурной безопасности:

- *внесение в конфигурацию той или иной формы* ***избыточности*** (резервное оборудование, запасные каналы связи и т.п.);
- *наличие средств обнаружения нештатных ситуаций*;
- *наличие средств* ***реконфигурирования*** *для восстановления,* ***изоляции*** *и/или замены компонентов, отказавших или подвергшихся атаке на доступность*;
- *рассредоточенность сетевого управления, отсутствие* ***единой точки отказа***;
- *выделение подсетей и изоляция групп пользователей друг от друга*. Данная мера, являющаяся обобщением разделения процессов на уровне операционной системы, ограничивает зону поражения при возможных нарушениях информационной безопасности.
- *минимизация объема защитных средств, выносимых на клиентские системы.*
- *реализация сервисов безопасности на сетевом и транспортном уровнях*
- *поддержка механизмов аутентификации, устойчивых к сетевым угрозам.*

ЧАСТЬ 3. КРИПТОГРАФИЧЕСКАЯ ЗАЩИТА ИНФОРМАЦИИ

10. ОСНОВНЫЕ ПРИНЦИПЫ КРИПТОГРАФИЧЕСКОЙ ЗАШИТЫ ИНФОРМАЦИИ

10.1 Понятие криптографии

Криптография *представляет собой совокупность методов преобразования данных, направленных на то, чтобы сделать эти данные бесполезными для противника.* Такие преобразования позволяют решить две главные проблемы защиты данных: ***проблему конфиденциальности*** (путем лишения противника возможности извлечь информацию из канала связи) и ***проблему целостности*** (путем лишения противника возможности изменить сообщение так, чтобы изменился его смысл, или ввести ложную информацию в канал связи).

Обобщенная схема криптографической системы, обеспечивающей шифрование передаваемой информации, показана на рисунке 10.1.

Рис. 10.1 - Обобщенная схема криптосистемы

Отправитель генерирует ***открытый текст*** исходного сообщения *M*, которое должно быть передано законному ***получателю*** по незащищенному каналу. За каналом следит ***перехватчик*** с целью перехватить и раскрыть передаваемое сообщение. Для того чтобы перехватчик не смог узнать содержание сообщения *M*, отправитель шифрует его с помощью обратимого преобразования $E_к$ и получает ***шифртекст*** (или ***криптограмму***) $C = E_K(M)$, который отправляет получателю.

Законный получатель, приняв шифртекст C, расшифровывает его с помощью обратного преобразования $D = E_K^{-1}$ и получает исходное сообщение в виде открытого текста *M*:

$$D_k(C) = E_k^{-1}(E_k(M)) = M$$

Преобразование $E_к$ выбирается из семейства криптографических преобразований, называемых ***криптоалгоритмами****. Параметр, с помощью которого выбирается отдельное используемое преобразование, называется* ***криптографическим ключом K***.

Криптосистема имеет разные варианты реализации: набор инструкций, аппаратные средства, комплекс программ компьютера, которые позволяют

зашифровать открытый текст и расшифровать шифртекст различными способами.

Ф*ормально,* ***криптографическая система*** *- это однопараметрическое семейство* $(E_k)_{k\in\bar{k}}$ *обратимых преобразований вида:*

$$E_k : \bar{M} \to \bar{C}$$

из пространства $\bar{M}$ *сообщений открытого текста в пространство* $\bar{C}$ *шифрованных текстов. Параметр* ***К*** *(ключ) выбирается из конечного множества* $\bar{K}$*, называемого* ***пространством ключей****.*

Шифр *(в соответствии со стандартом ГОСТ 28147-89) - совокупность обратимых преобразований множества открытых данных на множество зашифрованных данных, задаваемых ключом и алгоритмом криптографического преобразования.*

Общая классификация алгоритмов шифрования представлена на рис. 10.2.

Рис. 10.2 - Общая классификация алгоритмов шифрования

Ключ *- это конкретное секретное состояние некоторых параметров алгоритма криптографического преобразования данных, обеспечивающее выбор только одного варианта из всех возможных для данного алгоритма.*

Криптостойкость *- основная характеристика шифра является, которая определяет его стойкость к раскрытию методами криптоанализа. Обычно эта характеристика определяется интервалом времени, необходимым для раскрытия шифра.*

К шифрам, используемым для криптографической защиты информации, предъявляется ряд требований:

1. *достаточная криптостойкость (надежность закрытия данных);*
2. *простота процедур шифрования и расшифрования;*
3. *незначительная избыточность информации за счет шифрования;*
4. *нечувствительность к небольшим ошибкам шифрования и др.*

В той или иной мере этим требованиям отвечает:

- ***Шифрование перестановкой*** *заключается в том, что символы шифруемого текста переставляются по определенному правилу в пределах некоторого блока этого текста.* При достаточной длине блока, в пределах которого осуществляется перестановка, и сложном неповторяющемся порядке перестановки можно достигнуть приемлемой для простых практических приложений стойкости шифра.
- ***Шифрование заменой (подстановкой)*** *заключается в том, что символы шифруемого текста заменяются символами того же или другого алфавита в соответствии с заранее обусловленной схемой замены.*
- ***Шифрование гаммированием*** *заключается в том, что символы шифруемого текста складываются с символами некоторой случайной последовательности, именуемой* ***гаммой-шифром***. Стойкость шифрования определяется в основном длиной (периодом) неповторяющейся части гаммы шифра. Поскольку с помощью ЭВМ можно генерировать практически бесконечную гамму шифра, то данный способ является одним из основных для шифрования информации в автоматизированных системах.
- ***Шифрование аналитическим преобразованием*** *заключается в том, что шифруемый текст преобразуется по некоторому аналитическому правилу (формуле).* Например, можно использовать правило умножения вектора на матрицу, причем умножаемая матрица является ключом шифрования (поэтому ее размер и содержание должны храниться в секрете), а символами умножаемого вектора последовательно служат символы шифруемого текста. Другим примером может служить использование так называемых однонаправленных функций для построения криптосистем с открытым ключом.

10.2 Понятия о симметричных и асимметричных криптосистемах

В общем случае шифрование может быть симметричным или асимметричным относительно преобразования расшифрования, что определяет два класса криптосистем:

- ***симметричные*** *(одноключевые) криптосистемы;*
- ***асимметричные*** *(двухключевые) криптосистемы (с открытым ключом).*

Схема симметричной криптосистемы с одним секретным ключом была показана на рис. 10.1. В ней используются одинаковые секретные ключи в блоке шифрования и блоке расшифрования.

Обобщенная схема асимметричной криптосистемы с двумя разными ключами K_1 и K_2 показана на рис. 10.3. В этой криптосистеме один из ключей является открытым, а другой - секретным.

Рис. 10.3 - Обобщенная схема асимметричной криптосистемы с открытым ключом

В симметричной криптосистеме секретный ключ надо передавать отправителю и получателю по защищенному каналу распространения ключей, например такому, как курьерская служба. На рис. 10.1 этот канал показан «экранированной» линией. *В асимметричной криптосистеме передают по незащищенному каналу только открытый ключ, а секретный ключ сохраняют на месте его генерации.*

10.3 Понятие криптоанализа

Криптоанализ *- это наука о раскрытии исходного текста зашифрованного сообщения без доступа к ключу.* Успешный анализ может раскрыть исходный текст или ключ. Он позволяет также обнаружить слабые места в криптосистеме, что, в конечном счете, ведет к тем же результатам.

Фундаментальное правило криптоанализа (впервые сформулированное голландцем А. Керкхоффом еще в XIX веке) ***заключается в том, что стойкость шифра (криптосистемы) должна определяться только секретностью ключа***. Иными словами, правило Керкхоффа состоит в том, что весь алгоритм шифрования, кроме значения секретного ключа, известен криптоаналитику противника. Это обусловлено тем, что криптосистема, реализующая семейство криптографических преобразований, обычно рассматривается как открытая система. Такой подход отражает очень важный принцип технологии защиты информации: защищенность системы не должна зависеть от секретности чего-либо такого, что невозможно быстро изменить в случае утечки секретной информации.

Другое почти общепринятое допущение в криптоанализе состоит в том, что криптоаналитик имеет в своем распоряжении шифртексты сообщений.

Криптоаналитическая атака *- любая попытка со стороны перехватчика расшифровать шифртекст C для получения открытого текста M или зашифровать свой собственный текст M' для получения правдоподобного шифртекста C', не имея подлинного ключа.*

На рис. 10.4 показан поток информации в криптосистеме в случае активных действий перехватчика. Активный перехватчик не только

считывает все шифртексты, передаваемые по каналу, но может также пытаться изменять их по своему усмотрению.

Рис. 10.4 - Поток информации в криптосистеме при активном перехвате сообщений

Если предпринятые криптоаналитические атаки не достигают поставленной цели и криптоаналитик не может, не имея подлинного ключа, вывести M из C или C' из M', то полагают, что такая криптосистема является ***криптостойкой****.*

Существует четыре основных типа криптоаналитических атак (все они формулируются в предположении, что криптоаналитику известны применяемый алгоритм шифрования и шифртексты сообщений):

- ***Криптоаналитическая атака при наличии только известного шифртекста.*** Криптоаналитик имеет только шифртексты $C_1, C_2, ..., C_i$ нескольких сообщений, причем все они зашифрованы с использованием одного и того же алгоритма шифрования E_K. Работа криптоаналитика заключается в том, чтобы раскрыть исходные тексты $M_1, M_2, ..., M_i$ по возможности большинства сообщений или, еще лучше, вычислить ключ K, использованный для шифрования этих сообщений, с тем, чтобы расшифровать и другие сообщения, зашифрованные этим ключом.
- ***Криптоаналитическая атака при наличии известного открытого текста.*** Криптоаналитик имеет доступ не только к шифртекстам $C_1, C_2, ..., C_i$ нескольких сообщений, но также к открытым текстам $M_1, M_2, ..., M_i$ этих сообщений. Его работа заключается в нахождении ключа K, используемого при шифровании этих сообщений, или алгоритма расшифрования D_K любых новых сообщений, зашифрованных тем же самым ключом.
- ***Криптоаналитическая атака при возможности выбора открытого текста.*** Криптоаналитик не только имеет доступ к шифртекстам $C_1, C_2, ..., C_i$ и связанным с ними открытым текстам нескольких сообщений, но и может по желанию выбирать открытые тексты, которые затем получает в зашифрованном виде. Такой

криптоанализ получается более мощным по сравнению с криптоанализом с известным открытым текстом, потому что криптоаналитик может выбрать для шифрования такие блоки открытого текста, которые дадут больше информации о ключе. Работа криптоаналитика состоит в поиске ключа K, использованного для шифрования сообщений, или алгоритма расшифрования D_K новых сообщений, зашифрованных тем же ключом.

- ***Криптоаналитическая атака с адаптивным выбором открытого текста.*** Это особый вариант атаки с выбором открытого текста. Криптоаналитик может не только выбирать открытый текст, который затем шифруется, но и изменять свой выбор в зависимости от результатов предыдущего шифрования. При криптоанализе с простым выбором открытого текста криптоаналитик обычно может выбирать несколько крупных блоков открытого текста для их шифрования; при криптоанализе с адаптивным выбором открытого текста он имеет возможность выбрать сначала более мелкий пробный блок открытого текста, затем выбрать следующий блок в зависимости от результатов первого выбора, и т.д. Эта атака предоставляет криптоаналитику еще больше возможностей, чем предыдущие типы атак.

Кроме перечисленных основных типов криптоаналитических атак, можно отметить, по крайней мере, еще два типа:

1. ***Криптоаналитическая атака с использованием выбранного шифртекста.*** Криптоаналитик может выбирать для расшифрования различные шифртексты $C_1, C_2, ..., C_i$ и имеет доступ к расшифрованным открытым текстам $M_1, M_2, ..., M_i$. Например, криптоаналитик получил доступ к защищенному от несанкционированного вскрытия блоку, который выполняет автоматическое расшифрование. Работа криптоаналитика заключается в нахождении ключа. Этот тип криптоанализа представляет особый интерес для раскрытия алгоритмов с открытым ключом.
2. ***Криптоаналитическая атака методом полного перебора всех возможных ключей.*** Эта атака предполагает использование криптоаналитиком известного шифртекста и осуществляется посредством полного перебора всех возможных ключей с проверкой, является ли осмысленным получающийся открытый текст. Такой подход требует привлечения предельных вычислительных ресурсов и иногда называется силовой атакой.

10.4 Аппаратно-программные криптографические средства защиты информации

Аппаратно-программные средства, обеспечивающие повышенный уровень защиты, можно разбить на пять основных групп (рис. 10.4).

Рис. 10.5 - Аппаратно-программные средства защиты компьютерной информации

10.4.1 Системы идентификации и аутентификации пользователей

Системы идентификации и аутентификации пользователей *применяются для ограничения доступа случайных и незаконных пользователей к ресурсам компьютерной системы. Общий алгоритм работы этих систем заключается в том, чтобы получить от пользователя информацию, удостоверяющую его личность, проверить ее подлинность и затем предоставить (или не предоставить) этому пользователю возможность работы с системой.*

При построении подобных систем возникает проблема выбора информации, на основе которой осуществляются процедуры идентификации и аутентификации пользователя. Можно выделить следующие типы:

- секретная информация, которой обладает пользователь (пароль, персональный идентификатор, секретный ключ и т.п.); эту информацию пользователь должен запомнить или же могут быть применены специальные средства хранения этой информации;
- физиологические параметры человека (отпечатки пальцев, рисунок радужной оболочки глаза и т.п.) или особенности поведения человека (особенности работы на клавиатуре и т. п.).

Системы идентификации, основанные на первом типе информации, принято считать *традиционными.* Системы идентификации, использующие второй тип информации, называются *биометрическими.*

10.4.2 Системы шифрования дисковых данных

Вторую группу средств, обеспечивающих повышенный уровень защиты, составляют *системы шифрования дисковых данных.* ***Основная задача, решаемая такими системами, состоит в защите от несанкционированного использования данных, расположенных на магнитных носителях.***

Обеспечение конфиденциальности данных, располагаемых на магнитных носителях, осуществляется путем их шифрования с использованием симметричных алгоритмов шифрования. Основным классификационным признаком для комплексов шифрования служит уровень их встраивания в компьютерную систему.

Работа прикладных программ с дисковыми накопителями состоит из двух этапов

1. ***Логический этап*** соответствует уровню взаимодействия прикладной программы с операционной системой (например, вызов сервисных функций чтения/записи данных). На этом уровне основным объектом является файл.
2. ***Физический этап*** соответствует уровню взаимодействия операционной системы и аппаратуры. В качестве объектов этого уровня выступают структуры физической организации данных - сектора диска.

В результате системы шифрования данных могут осуществлять криптографические преобразования данных на уровне файлов (защищаются отдельные файлы) и на уровне дисков (защищаются диски целиком).

Другим классификационным признаком систем шифрования дисковых данных является способ их функционирования. По способу функционирования системы шифрования дисковых данных делят на два класса:

1. **системы «прозрачного» шифрования**, в которых криптографические преобразования осуществляются в режиме реального времени, незаметно для пользователя. Например, пользователь записывает подготовленный в текстовом редакторе документ на защищаемый диск, а система защиты в процессе записи выполняет его шифрование.
2. **системы, специально вызываемые для осуществления шифрования**. Как правило, это утилиты, которые необходимо специально вызывать для выполнения шифрования. К ним относятся, например, архиваторы со встроенными средствами парольной защиты.

10.4.3 Системы шифрования данных

К третьей группе средств, обеспечивающих повышенный уровень защиты, относятся *системы шифрования данных, передаваемых по компьютерным сетям.*

Различают два основных способа шифрования:

1. *канальное шифрование*
2. *оконечное (абонентское) шифрование.*

В случае ***канального шифрования*** *защищается вся передаваемая по каналу связи информация, включая служебную.* Соответствующие процедуры шифрования реализуются с помощью протокола канального уровня семиуровневой эталонной модели взаимодействия открытых систем OSI (Open System Interconnection).

Способ *канального шифрования* обладает следующим достоинством - встраивание процедур шифрования на канальный уровень позволяет использовать аппаратные средства, что способствует повышению производительности системы.

Однако, у данного подхода имеются существенные недостатки:

- шифрованию на данном уровне подлежит вся информация, включая служебные данные транспортных протоколов; это осложняет механизм маршрутизации сетевых пакетов и требует расшифрования данных в устройствах промежуточной коммутации (шлюзах, ретрансляторах и т.п.);
- шифрование служебной информации, неизбежное на данном уровне, может привести к появлению статистических закономерностей в шифрованных данных; это влияет на надежность защиты и накладывает ограничения на использование криптографических алгоритмов.

Оконечное (абонентское) шифрование *позволяет обеспечить конфиденциальность данных, передаваемых между двумя прикладными объектами (абонентами).* Оконечное шифрование реализуется с помощью протокола прикладного или представительного уровня эталонной модели OSI. В этом случае защищенным оказывается только содержание сообщения, вся служебная информация остается открытой. Данный способ позволяет избежать проблем, связанных с шифрованием служебной информации, но при этом возникают другие проблемы. В частности, злоумышленник, имеющий доступ к каналам связи компьютерной сети, получает возможность анализировать информацию о структуре обмена сообщениями, например об отправителе и получателе, о времени и условиях передачи данных, а также об объеме передаваемых данных.

10.4.4 Системы аутентификации электронных данных

При обмене электронными данными по сетям связи возникает проблема ***аутентификации*** - *т.е. установление подлинности автора документа и проверка отсутствия изменений в полученном документе.*

Для аутентификации электронных данных применяют:

- код аутентификации сообщения (имитовставку);
- электронную цифровую подпись.

При формировании кода аутентификации сообщения и электронной цифровой подписи используются разные типы систем шифрования.

Код аутентификации сообщения формируют с помощью симметричных систем шифрования данных. В частности, симметричный алгоритм шифрования данных DES при работе в режиме сцепления блоков шифра CBC позволяет сформировать с помощью секретного ключа и начального вектора IV код аутентификации сообщения MAC (Message Authentication Code). Проверка целостности принятого сообщения осуществляется путем проверки кода MAC получателем сообщения.

Аналогичные возможности предоставляет отечественный стандарт симметричного шифрования данных ГОСТ 28147-89. В этом алгоритме предусмотрен режим выработки имитовставки, обеспечивающий *имитозащиту,* т.е. защиту системы шифрованной связи от навязывания ложных данных.

Имитовставка вырабатывается из открытых данных посредством специального преобразования шифрования с использованием секретного ключа и передается по каналу связи в конце зашифрованных данных. Имитовставка проверяется получателем сообщения, владеющим секретным ключом, путем повторения процедуры, выполненной ранее отправителем, над полученными открытыми данными.

Электронная цифровая подпись **(ЭЦП)** представляет собой относительно небольшое количество дополнительной аутентифи-цирующей цифровой информации, передаваемой вместе с подписываемым текстом. Для реализации ЭЦП используются принципы асимметричного шифрования. Система ЭЦП включает процедуру формирования цифровой подписи отправителем с использованием секретного ключа отправителя и процедуру проверки подписи получателем с использованием открытого ключа отправителя.

10.4.5 Средства управления ключевой информацией

Пятую группу средств, обеспечивающих повышенный уровень защиты, *образуют средства управления ключевой информацией.*

Под ключевой информацией понимается *совокупность всех используемых в компьютерной системе или сети криптографических*

ключей. Безопасность любого криптографического алгоритма определяется используемыми криптографическими ключами. В случае ненадежного управления ключами злоумышленник может завладеть ключевой информацией и получить полный доступ ко всей информации в компьютерной системе или сети.

Основным классификационным признаком средств управления ключевой информацией является вид функции управления ключами. Различают следующие основные виды функций управления ключами:

1. генерация ключей;
2. хранение ключей;
3. распределение ключей.

Способы *генерации ключей* различаются для симметричных и асимметричных криптосистем.

Для генерации ключей симметричных криптосистем используются аппаратные и программные средства генерации случайных чисел.

Генерация ключей для асимметричных криптосистем представляет существенно более сложную задачу в связи с необходимостью получения ключей с определенными математическими свойствами.

Функция *хранения ключей* предполагает организацию безопасного хранения, учета и удаления ключей. Для обеспечения безопасного хранения и передачи ключей применяют их шифрование с помощью других ключей. Такой подход приводит к *концепции иерархии ключей.* В иерархию ключей обычно входят главный ключ (мастер-ключ), ключ шифрования ключей и ключ шифрования данных. Следует отметить, что генерация и хранение мастер-ключей являются критическими вопросами криптографической защиты.

Распределение ключей является самым ответственным процессом в управлении ключами. Этот процесс должен гарантировать скрытность распределяемых ключей, а также оперативность и точность их распределения.

Различают два основных способа распределения ключей между пользователями компьютерной сети:

1. применение одного или нескольких центров распределения ключей;
2. прямой обмен сеансовыми ключами между пользователями.

11. АСИММЕТРИЧНЫЕ КРИПТОСИСТЕМЫ

11.1 Концепция криптосистемы с открытым ключом

Эффективными системами криптографической защиты данных являются асимметричные криптосистемы, называемые также криптосистемами с открытым ключом. В таких системах для зашифрования данных используется один ключ, а для расшифрования - другой ключ (отсюда и название-асимметричные). Первый ключ является открытым и может быть опубликован для использования всеми пользователями системы, которые зашифровывают данные. Расшифрование данных с помощью открытого ключа невозможно.

Для расшифрования данных получатель зашифрованной информации использует второй ключ, который является *секретным.* Разумеется, ключ расшифрования не может быть определен из ключа зашифрования.

Обобщенная схема асимметричной криптосистемы с открытым ключом показана на рис. 11.1.

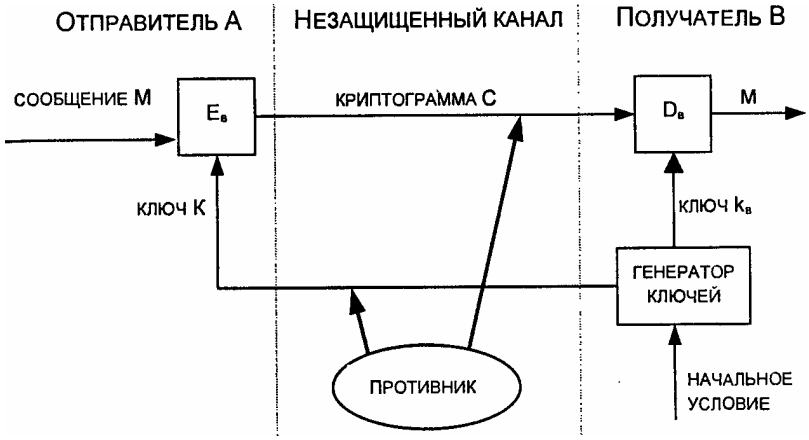


Рис. 11.1 - Обобщенная схема асимметричной криптосистемы

В этой асимметричной криптосистеме применяют два различных ключа:

- *$К_в$ - открытый ключ отправителя А;*
- *$к_в$ - секретный ключ получателя В.*

Генератор ключей целесообразно располагать на стороне получателя *В* (чтобы не пересылать секретный ключ $к_в$ по незащищенному каналу). Значения ключей $К_в$ и $к_в$ зависят от начального состояния генератора ключей.

Раскрытие секретного ключа $к_в$ по известному открытому ключу $К_в$ должно быть вычислительно неразрешимой задачей.

Характерные особенности асимметричных криптосистем:

1. *Открытый ключ $К_в$ и криптограмма С могут быть отправлены по незащищенным каналам, т.е. противнику известны $К_в$ и С.*

2. Алгоритмы шифрования и расшифрования являются открытыми:

$$E_в : M \rightarrow C,$$

$$E_в^{-1} : C \rightarrow M.$$

Защита информации в асимметричной криптосистеме основана на секретности ключа $к_в$.

У. Диффи и М. Хеллман сформулировали требования, выполнение которых обеспечивает безопасность асимметричной криптосистемы:

1. Вычисление пары ключей ($К_в$, $k_в$) получателем B на основе начального условия должно быть простым.
2. Отправитель A, зная открытый ключ $К_в$ и сообщение M, может легко вычислить криптограмму
$$C = E_{Кв}(M) .$$
3. Получатель B, используя секретный ключ $k_в$ и криптограмму C, может легко восстановить исходное сообщение
$$M = E_{кв}^{-1}(C) = E_{кв}^{-1}(C) = E_{кв}^{-1}[\, E_{Кв}(M)\,].$$
4. Противник, зная открытый ключ $К_в$, при попытке вычислить секретный ключ $k_в$ наталкивается на непреодолимую вычислительную проблему. Противник, зная пару ($К_в$, C), при попытке вычислить исходное сообщение M наталкивается на непреодолимую вычислительную проблему.

11.2 Однонаправленные функции

Концепция асимметричных криптографических систем с открытым ключом основана на применении однонаправленных функций.

Неформально однонаправленную функцию можно определить следующим образом. Пусть $\boldsymbol{X}$ и $\boldsymbol{Y}$ - некоторые произвольные множества. Функция

$$\boldsymbol{f : X \rightarrow Y}$$

является однонаправленной, если для всех $\boldsymbol{x} \in \boldsymbol{X}$ можно легко вычислить функцию

$$\boldsymbol{y = f(x)}, \text{ где } \boldsymbol{y} \in \boldsymbol{Y}.$$

И в то же время для большинства $y \in Y$ достаточно сложно получить значение $\boldsymbol{x} \in \boldsymbol{X}$, такое, что $\boldsymbol{f^{-1}(x) = y}$ (при этом полагают, что существует по крайней мере одно такое значение $\boldsymbol{x}$).

Основным критерием отнесения функции $\boldsymbol{f}$ к классу однонаправленных функций является отсутствие эффективных алгоритмов обратного преобразования $\boldsymbol{Y \rightarrow X}$.

В качестве первого примера однонаправленной функции рассмотрим целочисленное умножение. Прямая задача - вычисление произведения двух очень больших целых чисел P и Q, т.е. нахождение значения

$$N = P \cdot Q,$$

является относительно несложной задачей для ЭВМ.

Обратная задача-разложение на множители большого целого числа, т.е. нахождение делителей P и Q большого целого числа $N = P \cdot Q$, является практически неразрешимой задачей при достаточно больших значениях **N**. По современным оценкам теории чисел при целом $N \approx 2^{664}$ и $P \approx Q$ для разложения числа N потребуется около 10^{23} операций, т.е. задача практически неразрешима на современных ЭВМ.

Следующий характерный пример однонаправленной функции - это модульная экспонента с фиксированными основанием и модулем. Пусть A и N-целые числа, такие, что $1 \le A < N$. Определим множество Z_N:

$$Z_N = \{0,1, 2, \dots , N-1\}.$$

Тогда модульная экспонента с основанием A по модулю N представляет собой функцию

$$f_{A.N} : Z_N \rightarrow Z_N,$$

$$f_{A.N}(x) = A^x \bmod N,$$

где X- целое число, $1 < x < N-1$; операция $i \bmod j$ - остаток от целочисленного деления i на j.

Существуют эффективные алгоритмы, позволяющие достаточно быстро вычислить значения функции $f_{A,N}(x)$.

Если $y = A^x$, то естественно записать $x = \log_A(y)$.

Поэтому задачу обращения функции $f_{A,N}(x)$ называют задачей нахождения дискретного логарифма или задачей дискретного логарифмирования.

Задача дискретного логарифмирования формулируется следующим образом. Для известных целых A, N, y найти целое число x, такое, что

$$A^x \bmod N = y.$$

Алгоритм вычисления дискретного логарифма за приемлемое время пока не найден. Поэтому модульная экспонента считается однонаправленной функцией.

По современным оценкам теории чисел при целых числах $A \approx 2^{664}$ и $N \approx 2^{664}$ решение задачи дискретного логарифмирования (нахождение показателя степени x для известного y) потребует около 10^{26} операций, т.е.

эта задача имеет в 10^3 раз большую вычислительную сложность, чем задача разложения на множители. При увеличении длины чисел разница в оценках сложности задач возрастает.

Следует отметить, что пока не удалось доказать, что не существует эффективного алгоритма вычисления дискретного логарифма за приемлемое время. Исходя из этого, модульная экспонента отнесена к однонаправленным функциям условно, что, однако, не мешает с успехом применять ее на практике.

Вторым важным классом функций, используемых при построении криптосистем с открытым ключом, являются так называемые однонаправленные функции с «потайным ходом» (с лазейкой). Дадим неформальное определение такой функции. Функция

$$f : X \to Y$$

относится к классу однонаправленных функций с **«потайным ходом»** в том случае, если она является однонаправленной и, кроме того, возможно эффективное вычисление обратной функции, если известен «потайной ход» (секретное число, строка или другая информация, ассоциирующаяся с данной функцией).

В качестве примера однонаправленной функции с «потайным ходом» можно указать используемую в криптосистеме RSA модульную экспоненту с фиксированными модулем и показателем степени. Переменное основание модульной экспоненты используется для указания числового значения сообщения *M* либо криптограммы *C*.

11.3 Криптосистема шифрования данных RSA

Алгоритм RSA предложили в 1978 г. три автора: Р. Райвест (Rivest), А. Шамир (Shamir) и А. Адлеман (Adleman). Алгоритм получил свое название по первым буквам фамилий его авторов. Алгоритм RSA стал первым полноценным алгоритмом с открытым ключом, который может работать как в режиме шифрования данных, так и в режиме электронной цифровой подписи.

Надежность алгоритма основывается на трудности факторизации больших чисел и трудности вычисления дискретных логарифмов.

Введем следующие понятия:

1. Простое число - делится только на 1 и на само себя;
2. Взаимно простым- не имеют ни одного общего делителя, кроме 1;
3. Результат операции $i \bmod j$ - остаток от целочисленного деления i на j.

В криптосистеме RSA открытый ключ $K_в$, секретный ключ $к_в$, сообщение *M* и криптограмма C принадлежат множеству целых чисел

$$Z_N = \{0, 1, 2, \ldots, N\text{-}1\}, \text{ где } N \text{ - модуль: } N = P \cdot Q.$$

Здесь P и Q - случайные большие простые числа. Для обеспечения максимальной безопасности выбирают P и Q равной длины и хранят в секрете.

Множество Z_N с операциями сложения и умножения по модулю N образует арифметику по модулю N.

Открытый ключ $K_в$ выбирают случайным образом так, чтобы выполнялись условия:

$$1 < K_B \le \varphi(N),\ НОД(K_B, \varphi(N)) = 1,\ \varphi(N) = (P-1)(Q-1),$$

где: $\varphi(N)$ - функция Эйлера; **НОД** - наибольший общий делитель.

Функция Эйлера $\varphi(N)$ указывает количество положительных целых чисел в интервале от 1 до **N**, которые взаимно просты с N.

Второе из указанных выше условий означает, что открытый ключ $K_в$ и функция Эйлера $\varphi(N)$ должны быть взаимно простыми.

Далее, вычисляют секретный ключ $k_в$, такой, что:

$$(k_B \cdot K_B) \bmod \varphi(N) = 1$$

или

$$k_B = \left(\frac{1}{K_B}\right) \bmod \left((P-1)(Q-1)\right).$$

Это можно осуществить, так как получатель B знает пару простых чисел (P, Q) и может легко найти $\varphi(N)$. Заметим, что $k_в$ и N должны быть взаимно простыми.

Открытый ключ $K_в$ используют для шифрования данных, а секретный ключ $k_в$ - для расшифрования.

Преобразование шифрования определяет криптограмму **C** через пару (открытый ключ $K_в$, сообщение M) в соответствии со следующей формулой:

$$C = E_{K_B}(M) = \left(M^{K_B}\right) \bmod N.$$

В качестве алгоритма быстрого вычисления значения C используют ряд последовательных возведений в квадрат целого M и умножений на M с приведением по модулю N.

Обращение функции $C = M^{K_B} \bmod N$, т.е. определение значения M по известным значениям C, $K_в$ и N, практически не осуществимо при $N \approx 2^{512}$.

Однако обратную задачу, т.е. задачу расшифрования криптограммы $\boldsymbol{C}$, можно решить, используя пару (секретный ключ $\boldsymbol{k_в}$, криптограмма $\boldsymbol{C}$) по следующей формуле:

$$M = E_{k_B}^{-1}(C) = \left(C^{k_B}\right) \bmod (N).$$

Таким образом, получатель B, который создает криптосистему, защищает два параметра:

- секретный ключ $\boldsymbol{k_в}$
- пару чисел $\boldsymbol{(P, Q)}$, произведение которых дает значение модуля N.

С другой стороны, получатель B открывает значение модуля N и открытый ключ $\boldsymbol{K_в}$.

Противнику известны лишь значения $\boldsymbol{K_в}$ и $\boldsymbol{N}$. Если бы он смог разложить число $\boldsymbol{N}$ на множители $\boldsymbol{P}$ и $\boldsymbol{Q}$, то он узнал бы «потайной ход» - тройку чисел $\boldsymbol{\{P, Q, K_в\}}$, вычислил значение функции Эйлера

$$\boldsymbol{\varphi(N) = (P-1)(Q-1)}$$

и определил значение секретного ключа $\boldsymbol{k_в}$. Однако, как уже отмечалось, разложение очень большого $\boldsymbol{N}$ на множители вычислительно не осуществимо (при условии, что длины выбранных $\boldsymbol{P}$ и $\boldsymbol{Q}$ составляют не менее 100 десятичных знаков).

11.3.1 Процедуры шифрования и расшифрования в криптосистеме RSA

Предположим, что пользователь A хочет передать пользователю B сообщение в зашифрованном виде, используя криптосистему RSA. В таком случае пользователь A выступает в роли отправителя сообщения, а пользователь B - в роли получателя. Как отмечалось выше, криптосистему RSA должен сформировать получатель сообщения, т.е. пользователь B. Рассмотрим последовательность действий пользователя B и пользователя A.

1. Пользователь B выбирает два произвольных больших простых числа $\boldsymbol{P}$ и $\boldsymbol{Q}$.
2. Пользователь B вычисляет значение модуля $\boldsymbol{N = P \cdot Q}$.
3. Пользователь B вычисляет функцию Эйлера: $\boldsymbol{\varphi(N) = (P-1)(Q-1)}$ и выбирает случайным образом значение большого случайного числа, которое назовем $\boldsymbol{k_в}$. Это число должно быть взаимно простым с функцией Эйлера (т.е. результатом умножения $\boldsymbol{\varphi(N) = (P-1)(Q-1)}$)
4. Определяется такое число $\boldsymbol{K_B}$, для которого является истинным следующее соотношения

$$\boldsymbol{(K_B \cdot k_в) \bmod (\varphi(N)) = 1}$$

и

$$\boldsymbol{1 < K_B \leq \varphi(N)}.$$

5. Пара чисел ($\boldsymbol{N}$, $\boldsymbol{K_в}$) является открытым ключом и может быть передана по незащищенному каналу. А пара чисел ($\boldsymbol{N}$, $\boldsymbol{k_в}$) – закрытым ключом, он держится в секрете и используется для дешифрации.

   Если пользователь *A* хочет передать пользователю *B* сообщение $\boldsymbol{M}$, он выполняет следующие шаги.

6. Пользователь *A* разбивает исходный открытый текст $\boldsymbol{M}$ на блоки (только до $\boldsymbol{N}$**-1**), каждый из которых может быть представлен в виде числа

$$\boldsymbol{M_i = 0, 1, 2, \ldots, N\text{-}1}.$$

7. Пользователь *A* шифрует текст, представленный в виде последовательности чисел $\boldsymbol{M_i}$ по формуле

$$C_i = \left(M_i^{K_B}\right) \bmod \left(N\right)$$

   и отправляет криптограмму

$$\boldsymbol{C_0, C_1, \ldots C_{N\text{-}1}}$$

8. Чтобы расшифровать эти данные, используя секретный ключ ($\boldsymbol{N}$, $\boldsymbol{k_в}$), необходимо выполнить следующие вычисления:

$$M_i = \left(C_i^{k_B}\right) \bmod \left(N\right)$$

В результате будет получена последовательность чисел $\boldsymbol{M_i}$, которые представляют собой исходное сообщение $\boldsymbol{M}$. Чтобы алгоритм RSA имел практическую ценность, необходимо иметь возможность без существенных затрат генерировать большие простые числа, уметь оперативно вычислять значения ключей $\boldsymbol{K_в}$ и $\boldsymbol{k_в}$.

11.3.2 Пример использования алгоритма RSA

Зашифруем и расшифруем сообщение «CAB» по алгоритму RSA. Для простоты будут использованы маленькие числа. На практике применяются очень большие числа (см. следующий раздел).

1. Выберем $P = 3$ и $Q = 11$.

2. Определим $N = P \cdot Q = 3 \cdot 11 = 33$.

3. Найдем $\boldsymbol{\varphi(N) = (P\text{-}1)(Q\text{-}1)} = (3\text{-}1)(11\text{-}1) = 20$.

Выбираем случайным образом значение числа $\boldsymbol{k_в}$. Это число должно быть взаимно простым с функцией Эйлера (т.е. у $\boldsymbol{\varphi(N)} = 20$ и $\boldsymbol{k_в}$ не должно быть общих делителей кроме 1). Пусть $\boldsymbol{k_в = 3}$.

4. Выберем число $\boldsymbol{K_B}$ по следующей формуле:

$$(\boldsymbol{k_B \cdot K_B}) \bmod 20 = 1,$$

т. е. произведение $\boldsymbol{k_B \cdot K_B}$ при целочисленном делении на $\boldsymbol{\varphi(N)} = 20$ должно в остатке давать 1.

Пусть $\boldsymbol{K_B}$ **= 7** т.к.: 7·3=21 и (21 mod 20) = 1

21|20
20|1
1

5. Представим шифруемое сообщение как последовательность чисел в диапазоне от 0 до 32 (кончается на ***N*-1**). Буква ***A*** =1, ***B***=2, ***C***=3.

6. Зашифруем сообщение, используя открытый ключ ($\boldsymbol{K_B}$=7, ***N***=33):

$C_1 = 3^7 \bmod 33 = 2187 \bmod 33 = 9$;

2187|33
2178|66
9

$C_2 = 1^7 \bmod 33 = 1 \bmod 33 = 1$;

1|33
0|0
1

$C_3 = 2^7 \bmod 33 = 128 \bmod 33 = 29$;

128|33
99|3
29

Т.е. криптограмма представляет собой *C* = 09 01 29

7. Расшифруем эти данные, используя закрытый ключ (***N***=33, $\boldsymbol{k_в}$=3).

$M_1=9^3 \bmod 33 =729 \bmod 33 = 3$ (*C*);

729|33
726|22
3

$M_2=1^3 \bmod 33 =1 \bmod 33 = 1$ (*A*);

1|33
0|0
1

$M_3=29^3 \bmod 33 = 24389 \bmod 33 = 2$ (*B*);

24389|33
24387|739
2

Данные расшифрованы.

11.3.3 Безопасность и быстродействие криптосистемы RSA

Безопасность алгоритма RSA базируется на трудности решения задачи факторизации больших чисел, являющихся произведениями двух больших

простых чисел. Действительно, криптостойкость алгоритма RSA определяется тем, что после формирования секретного ключа $k_в$ и открытого ключа $K_в$ «стираются» значения простых чисел P и Q, и тогда исключительно трудно определить секретный ключ $k_в$ по открытому ключу $K_в$, поскольку для этого необходимо решить задачу нахождения делителей P и Q модуля N.

Разложение величины N на простые множители P и Q позволяет вычислить функцию $\varphi(N) = (P-T)\,(Q-T)$ и затем определить секретное значение k_B используя уравнение

$$(K_B \cdot k_в) \bmod (\varphi(N)) = 1.$$

Другим возможным способом криптоанализа алгоритма RSA является непосредственное вычисление или подбор значения функции $\varphi(N) = (P-T)\,(Q-T)$. Если установлено значение $\varphi(N)$, то сомножители P и Q вычисляются достаточно просто. В самом деле, пусть

$$x = P + Q = N + 1 - \varphi(N),$$
$$y = (P - Q)^2 = (P + Q)^2 - 4 * N.$$

Зная $\varphi(N)$, можно определить x и затем y; зная x и y, можно определить числа P и Q из следующих соотношений:

$$P = 1/2\,(x + \sqrt{y}), \quad Q = 1/2\,(x - \sqrt{y}).$$

Однако эта атака не проще задачи факторизации модуля N.

Задача факторизации является трудно разрешимой задачей для больших значений модуля N.

Сначала авторы алгоритма RSA предлагали для вычисления модуля N выбирать простые числа P и Q случайным образом, по 50 десятичных разрядов каждое. Считалось, что такие большие числа N очень трудно разложить на простые множители. Один из авторов алгоритма RSA, Р. Райвест, полагал, что разложение на простые множители числа из почти 130 десятичных цифр, приведенного в их публикации, потребует более 40 квадриллионов лет машинного времени. Однако этот прогноз не оправдался из-за сравнительно быстрого прогресса компьютеров и их вычислительной мощности, а также улучшения алгоритмов факторизации.

Один из наиболее быстрых алгоритмов, известных в настоящее время, алгоритм NFS (Number Field Sieve) может выполнить факторизацию большого числа N (с числом десятичных разрядов больше 120) за число шагов, оцениваемых величиной $e^{2(\ln n)^{1/3}(\ln(\ln n))^{2/3}}$.

В 1994 г. было факторизовано число со 129 десятичными цифрами. Это удалось осуществить математикам А. Ленстра и М. Манасси посредством организации распределенных вычислений на 1600 компьютерах, объединенных сетью, в течение восьми месяцев. По мнению А. Ленстра и М. Манасси, их работа компрометирует криптосистемы RSA и создает большую угрозу их дальнейшим применениям. Теперь разработчикам криптоалгоритмов с открытым ключом на базе RSA приходится избегать

применения чисел длиной менее 200 десятичных разрядов. Самые последние публикации предлагают применять для этого числа длиной не менее 250-300 десятичных разрядов.

Оценка безопасных длин ключей асимметричных криптосистем на ближайшие 20 лет исходя из прогноза развития компьютеров и их вычислительной мощности, а также возможного совершенствования алгоритмов факторизации приведена в таблице 11.1 даны (для трех групп пользователей - индивидуальных пользователей, корпораций и государственных организаций), в соответствии с различием требований к их информационной безопасности. Конечно, данные оценки следует рассматривать как сугубо приблизительные, как возможную тенденцию изменений безопасных длин ключей асимметричных криптосистем со временем.

Таблица 11.1 - Оценки длин ключей для асимметричных криптосистем, бит

Год	**Отдельные пользователи**	**Корпорации**	**Государственные организации**
1995	768	1280	1536
2000	1024	1280	1536
2005	1280	1536	2048
2010	1280	1536	2048
2015	1536	2048	2048

11.4 Аутентификация данных и электронная цифровая подпись

При обмене электронными документами по сети связи существенно снижаются затраты на обработку и хранение документов, убыстряется их поиск. Но при этом возникает проблема аутентификации автора документа и самого документа, т.е. установления подлинности автора и отсутствия изменений в полученном документе. В обычной (бумажной) информатике эти проблемы решаются за счет того, что информация в документе и рукописная подпись автора жестко связаны с физическим носителем (бумагой). В электронных документах на машинных носителях такой связи нет.

Целью аутентификации электронных документов является их защита от возможных видов злоумышленных действий, к которым относятся:

- ***активный перехват*** - нарушитель, подключившийся к сети, перехватывает документы (файлы) и изменяет их;
- ***маскарад*** - абонент *C* посылает документ абоненту *B* от имени абонента *A*;
- ***ренегатство*** - абонент *A* заявляет, что не посылал сообщения абоненту *B*, хотя на самом деле послал;

- ***подмена*** - абонент *B* изменяет или формирует новый документ и заявляет, что получил его от абонента *A*;
- ***повтор*** - абонент *C* повторяет ранее переданный документ, который абонент *A* посылал абоненту *B*.

Эти виды злоумышленных действий могут нанести существенный ущерб банковским и коммерческим структурам, государственным предприятиям и организациям, частным лицам, применяющим в своей деятельности компьютерные информационные технологии.

При обработке документов в электронной форме совершенно непригодны традиционные способы установления подлинности по рукописной подписи и оттиску печати на бумажном документе. Принципиально новым решением является электронная цифровая подпись (ЭЦП).

Электронная цифровая подпись используется для аутентификации текстов, передаваемых по телекоммуникационным каналам. Функционально она аналогична обычной рукописной подписи и обладает ее основными достоинствами:

1. удостоверяет, что подписанный текст исходит от лица, поставившего подпись;
2. не дает самому этому лицу возможности отказаться от обязательств, связанных с подписанным текстом;
3. гарантирует целостность подписанного текста.

Цифровая подпись представляет собой относительно небольшое количество дополнительной цифровой информации, передаваемой вместе с подписываемым текстом.

Система ЭЦП включает две процедуры:

1. процедуру постановки подписи;

2. процедуру проверки подписи.

В процедуре постановки подписи используется секретный ключ отправителя сообщения, в процедуре проверки подписи - открытый ключ отправителя.

При формировании ЭЦП отправитель прежде всего вычисляет хэш-функцию $\boldsymbol{h(M)}$ подписываемого текста M. Вычисленное значение хэш-функции $\boldsymbol{h(M)}$ представляет собой один короткий блок информации т, характеризующий весь текст *M* в целом. Затем число m шифруется секретным ключом отправителя. Получаемая при этом пара чисел представляет собой ЭЦП для данного текста *M*.

При проверке ЭЦП получатель сообщения снова вычисляет хэш-функцию $\boldsymbol{m = h(M)}$ принятого по каналу текста *M*, после чего при помощи открытого ключа отправителя проверяет, соответствует ли полученная подпись вычисленному значению m хэш-функции.

Принципиальным моментом в системе ЭЦП является невозможность подделки ЭЦП пользователя без знания его секретного ключа подписывания.

В качестве подписываемого документа может быть использован любой файл. Подписанный файл создается из неподписанного путем добавления в него одной или более электронных подписей.

Каждая подпись содержит следующую информацию:

- дату подписи;
- срок окончания действия ключа данной подписи;
- информацию о лице, подписавшем файл (Ф.И.О.. должность. краткое наименование фирмы);
- идентификатор подписавшего (имя открытого ключа);
- собственно цифровую подпись.

Технология применения системы ЭЦП предполагает наличие сети абонентов, посылающих друг другу подписанные электронные документы.

Для каждого абонента генерируется пара ключей: секретный и открытый. Секретный ключ хранится абонентом в тайне и используется им для формирования ЭЦП. Открытый ключ известен всем другим пользователям и предназначен для проверки ЭЦП получателем подписанного электронного документа. Иначе говоря, открытый ключ является необходимым инструментом, позволяющим проверить подлинность электронного документа и автора подписи. Открытый ключ не позволяет вычислить секретный ключ.

Для генерации пары ключей (секретного и открытого) в алгоритмах ЭЦП, как и в асимметричных системах шифрования, используются разные математические схемы, основанные на применении однонаправленных функций. Эти схемы разделяются на две группы. В основе такого разделения лежат известные сложные вычислительные задачи:

- задача факторизации (разложения на множители) больших целых чисел;
- задача дискретного логарифмирования.

11.5 Алгоритм цифровой подписи RSA

Первой и наиболее известной во всем мире конкретной системой ЭЦП стала система RSA, математическая схема которой была разработана в 1977 г. в Массачуссетском технологическом институте США.

Сначала необходимо вычислить пару ключей (секретный ключ и открытый ключ). Для этого отправитель (автор) электронных документов вычисляет два больших простых числа P и Q, затем находит их произведение

$$N = P \cdot Q$$

и значение функции

$$\varphi(N) = (P-1)(Q-1).$$

Далее отправитель вычисляет число ***E*** из условий:

$$E \le \varphi(N), \quad \text{НОД}\,(E, \varphi(N)) = 1$$

и число ***D*** из условий:

$$D < N, \quad E * D \equiv 1 (\text{mod}\ \varphi(N)).$$

Пара чисел **(*E*, *N*)** является открытым ключом. Эту пару чисел автор передает партнерам по переписке для проверки его цифровых подписей. Число ***D*** сохраняется автором как секретный ключ для подписывания.

Обобщенная схема формирования и проверки цифровой подписи RSA показана на рис. 11.2.

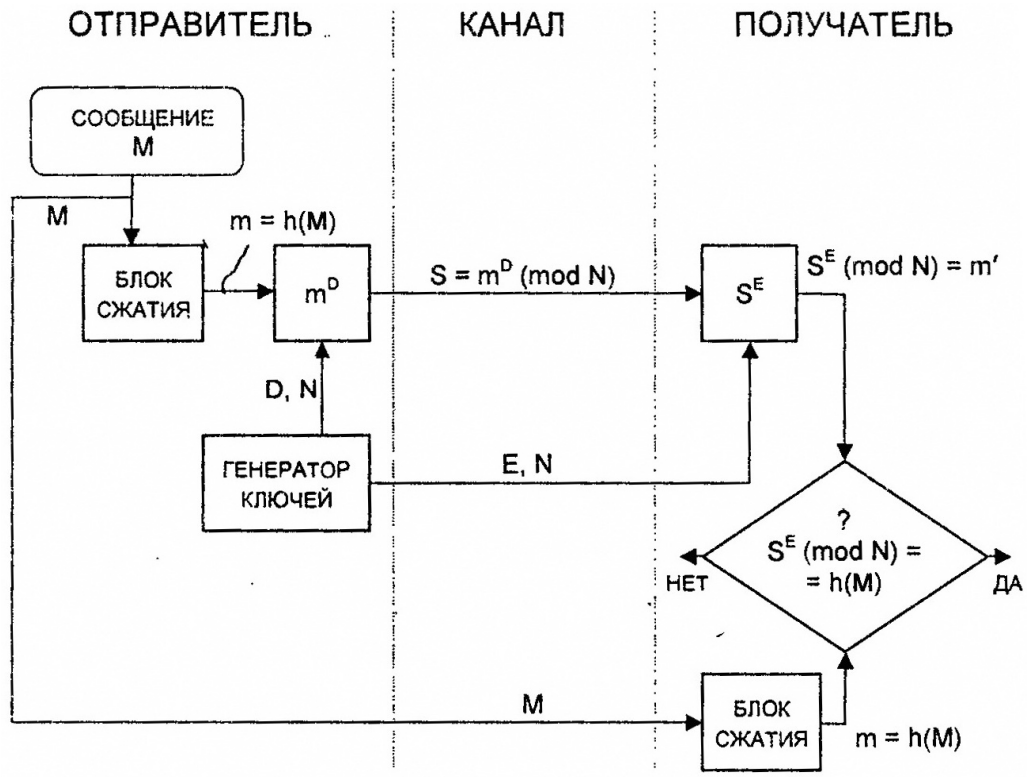


Рис. 11.2 - Обобщенная схема формирования и проверки ЭЦП RSA

Допустим, что отправитель хочет подписать сообщение *M* перед его отправкой. Сначала сообщение *M* (блок информации, файл, таблица) сжимают с помощью хэш-функции $h(\cdot)$ в целое число *m*:

$$\boldsymbol{m = h(M)}.$$

Затем вычисляют цифровую подпись *S* под электронным документом *M*, используя хэш-значение m и секретный ключ ***D***:

$$\boldsymbol{S = m^D \bmod N}.$$

Пара **(*M*, *S*)** передается партнеру-получателю как электронный документ ***M***, подписанный цифровой подписью ***S***, причем подпись ***S*** сформирована обладателем секретного ключа ***D***,

После приема пары **(*M*, *S*)** получатель вычисляет хэш-значение сообщения M двумя разными способами. Прежде всего он восстанавливает

хэш-значение m', применяя криптографическое преобразование подписи S с использованием открытого ключа E:

$$m'=S^E \bmod N.$$

Кроме того, он находит результат хэширования принятого сообщения M с помощью такой же хэш-функции $h(\cdot)$:

$$m = h(M).$$

Если соблюдается равенство вычисленных значений, т.е.

$$SE \bmod N = h(M),$$

то получатель признает пару (M, S) подлинной. Доказано, что только обладатель секретного ключа D может сформировать цифровую подпись S по документу M, а определить секретное число D по открытому числу E не легче, чем разложить модуль N на множители.

Кроме того, можно строго математически доказать, что результат проверки цифровой подписи **S** будет положительным только в том случае, если при вычислении S был использован секретный ключ D, соответствующий открытому ключу E. Поэтому открытый ключ E иногда называют «идентификатором» подписавшего.

Недостатки алгоритма цифровой подписи RSA.

- При вычислении модуля N, ключей E и D для системы цифровой подписи RSA необходимо проверять большое количество дополнительных условий, что сделать практически трудно. Невыполнение любого из этих условий делает возможным фальсификацию цифровой подписи со стороны того, кто обнаружит такое невыполнение. При подписании важных документов нельзя допускать такую возможность даже теоретически.
- Для обеспечения криптостойкости цифровой подписи RSA по отношению к попыткам фальсификации на уровне, например, национального стандарта США на шифрование информации (алгоритм DES), т.е. 10 , необходимо использовать при вычислениях N, D и E целые числа не менее 2^{512} (или около 10^{154}) каждое, что требует больших вычислительных затрат, превышающих на 20...30% вычислительные затраты других алгоритмов цифровой подписи при сохранении того же уровня криптостойкости.
- Цифровая подпись RSA уязвима к так называемой мультипликативной атаке. Иначе говоря, алгоритм цифровой подписи RSA позволяет злоумышленнику без знания секретного ключа D сформировать подписи под теми документами, у которых результат хэширования можно вычислить как произведение результатов хэширования уже подписанных документов.

12. СИММЕТРИЧНЫЕ КРИПТОСИСТЕМЫ

12.1 Понятие о симметричной криптосистеме

Большинство средств защиты информации базируется на использовании криптографических шифров и процедур шифрования – расшифрования осуществляющихся в рамках некоторой криптосистемы. **Характерной особенностью симметричной криптосистемы является применение одного и того же секретного ключа как при шифровании, так и при расшифровании сообщений.**

Как открытый текст, так и шифртекст образуются из букв, входящих в конечное множество символов, называемых алфавитом. Примерами алфавитов являются конечное множество всех заглавных букв, конечное множество всех заглавных и строчных букв и цифр и т. п. В общем виде некоторый алфавит Σ можно представить так:

$$\Sigma = \{a_0, a_1, ..., a_{m-1}\}.$$

Объединяя по определенному правилу буквы из алфавита Σ можно создать новые алфавиты:

- алфавит Σ^2, содержащий m^2 биграмм $a_0a_0, a_0a_1, ..., a_{m-1}a_{m-1}$
- алфавит Σ^3, содержащий m^3 триграмм $a_0a_0a_0, a_0a_0a_1, ..., a_{m-1}a_{m-1}a_{m-1}$.

В общем случае, объединяя по n букв, получаем алфавит Σ^n содержащий m^n n-грамм. Например, английский алфавит

$$\Sigma = \{\text{ABCDEFGH ... WXYZ}\}$$

объемом m=26 букв позволяет сгенерировать посредством операции конкатенации алфавит из 26^2 =676 биграмм:

AA, AB, … ,XZ, ZZ,

алфавит из 26^3=17576 триграмм:

AAA, AAB, … ,XZZ, ZZZ и т.д.

При выполнении криптографических преобразований полезно заменить буквы алфавита целыми числами 0, 1, 2, 3,... . Это позволяет упростить выполнение необходимых алгебраических манипуляций.

Например, можно установить взаимно однозначное соответствие между русским алфавитом $\Sigma = \{\text{АБВГД ... ЮЯ}\}$ и множеством целых $\bar{Z}_{32} = \{0,1,2,3,...,31\}$; между английским алфавитом $\Sigma_{англ} = \{ABCDEF...YZ\}$ и множеством целых $\bar{Z}_{26} = \{0,1,2,3,...,25\}$ (см. табл. 12.1 и 12.2).

В дальнейшем будет обычно использоваться алфавит

$$\bar{Z}_{26} = \{0,1,2,3,...,25\}$$

содержащий m «букв» (в виде чисел).

Замена букв традиционного алфавита числами позволяет более четко сформулировать основные концепции и приемы криптографических преобразований. В то же время в большинстве иллюстраций будет использоваться алфавит естественного языка.

Таблица 12.1 - Соответствие между русским алфавитом и множеством целых $\overline{Z}_{32} = \{0,1,2,3,...,31\}$

Буква	Число	Буква	Число	Буква	Число	Буква	Число
А	0	И	8	Р	16	Ш	24
Б	1	Й	9	С	17	Щ	25
В	2	К	10	Т	18	Ь	26
Г	3	Л	11	У	19	Ы	27
Д	4	М	12	Ф	20	Ъ	28
Е	5	Н	13	Х	21	Э	29
Ж	6	О	14	Ц	22	Ю	30
З	7	П	15	Ч	23	Я	31

Таблица 12.2 - Соответствие между английским алфавитом и множеством целых $\overline{Z}_{26} = \{0,1,2,3,...,25\}$

Буква	Число	Буква	Число	Буква	Число
A	0	J	9	S	18
B	1	K	10	T	19
C	2	L	11	U	20
D	3	M	12	V	21
E	4	N	13	W	22
F	5	O	14	X	23
G	6	P	15	Y	24
H	7	Q	16	Z	25
I	8	R	17		

Текст с n буквами из алфавита Z_m можно рассматривать как n-грамму

$$\overline{x} = (x_0, x_1, x_2, ..., x_{n-1}),$$

где $\overline{x}_i \in \overline{Z}_m \quad 0 \le i < n$, для некоторого целого $n = 1, 2, 3, \ldots$.Через $\overline{Z}_{m,n}$ будем обозначать множество n-грамм, образованных из букв множества $\overline{Z}_m$.

Криптографическое преобразование E представляет собой совокупность преобразований

$$E = \{E^{(n)} : 1 \le n < \infty\}$$

$$E^{(n)} : \overline{Z}_{m,n} \to \overline{Z}_{m,n}$$

Преобразование $E^{(n)}$ определяет, как каждая n-грамма открытого текста $\overline{x} \in \overline{Z}_{m,n}$ заменяется n-граммой шифртекста $\overline{y}$, т. е.

$$\overline{y} = E^{(n)}(\overline{x}),$$

причем

$$\overline{x}, \overline{y} \in \overline{Z}_{m,n},$$

при этом обязательным является требование взаимной однозначности преобразования $E^{(n)}$ на множестве $\overline{Z}_{m,n}$

Криптографическая система может трактоваться как семейство криптографических преобразований

$$\overline{E} = \{E_k : K \in \overline{K}\},$$

помеченных параметром K, называемым ключом.

Множество значений ключа образует ключевое пространство $\overline{K}$ Далее рассматриваются традиционные (классические) методы шифрования, отличающиеся симметричной функцией шифрования. К ним относятся шифры перестановки, шифры простой и сложной замены, а также некоторые их модификации и комбинации. Следует отметить, что комбинации шифров перестановок и шифров замены образуют все многообразие применяемых на практике симметричных шифров.

12.2 Шифры перестановки

При шифровании перестановкой символы шифруемого текста переставляются по определенному правилу в пределах блока этого текста. Шифры перестановки являются самыми простыми и, вероятно, самыми древними шифрами.

12.2.1 Шифрующие таблицы

С начала эпохи Возрождения (конец XIV столетия) начала возрождаться и криптография. Наряду с традиционными применениями криптографии в политике, дипломатии и военном деле появляются и другие задачи - защита интеллектуальной собственности от преследований инквизиции или заимствований злоумышленников. В разработанных шифрах перестановки того времени применяются шифрующие таблицы, которые в сущности задают правила перестановки букв в сообщении.

В качестве ключа в шифрующих таблицах используются:

1. размер таблицы;
2. слово или фраза, задающие перестановку;
3. особенности структуры таблицы.

Одним из самых примитивных табличных шифров перестановки является простая перестановка, для которой ключом служит размер таблицы. Этот метод шифрования сходен с шифром скитала. Например, сообщение

ТЕРМИНАТОР ПРИБЫВАЕТ СЕДЬМОГО В ПОЛНОЧЬ

записывается в таблицу поочередно по столбцам. Результат заполнения таблицы из 5 строк и 7 столбцов показан на рисунке 12.1.

Т	**Н**	**П**	**В**	**Е**	**Г**	**Л**
Е	**А**	**Р**	**А**	**Д**	**О**	**Н**
Р	**Т**	**И**	**Е**	**Ь**	**В**	**О**
М	**О**	**Б**	**Т**	**М**	**П**	**Ч**
И	**Р**	**Ы**	**С**	**О**	**О**	**Ь**

Рис. 12.1 - Заполнение таблицы из 5 строк и 7 столбцов

После заполнения таблицы текстом сообщения по столбцам для формирования шифртекста считывают содержимое таблицы по строкам. Если шифртекст записывать группами по пять букв, получается такое шифрованное сообщение:

ТНПВЕ ГЛЕАР АДОНР ТИЕЬВ ОМОБТ МПЧИР ЫСООЬ

Естественно, отправитель и получатель сообщения должны заранее условиться об общем ключе в виде размера таблицы. Следует заметить, что объединение букв шифртекста в 5-буквенные группы не входит в ключ шифра и осуществляется для удобства записи несмыслового текста. При расшифровании действия выполняют в обратном порядке.

Несколько большей стойкостью к раскрытию обладает метод шифрования, называемый одиночной перестановкой по ключу. Этот метод отличается от предыдущего тем, что столбцы таблицы переставляются по ключевому слову, фразе или набору чисел длиной в строку таблицы.

Применим в качестве ключа, например, слово

ПЕЛИКАН,

а текст сообщения возьмем из предыдущего примера. На рисунке 12.2 показаны две таблицы, заполненные текстом сообщения и ключевым словом, при этом левая таблица соответствует заполнению до перестановки, а правая таблица-заполнению после перестановки.

В верхней строке левой таблицы записан ключ, а номера под буквами ключа определены в соответствии с естественным порядком соответствующих букв ключа в алфавите. Если бы в ключе встретились одинаковые буквы, они бы были понумерованы слева направо. В правой таблице столбцы переставлены в соответствии с упорядоченными номерами букв ключа.

Рис. 12.2 – Шифрование с ключом

При считывании содержимого правой таблицы по строкам и записи шифртекста группами по пять букв получим шифрованное сообщение:

ГНВЕП ЛТООА ДРНЕВ ТЕЬИО РПОТМ БЧМОР СОЫЬИ

Для обеспечения дополнительной скрытности можно повторно зашифровать сообщение, которое уже прошло шифрование. Такой метод шифрования называется *двойной перестановкой.* В случае двойной перестановки столбцов и строк таблицы перестановки определяются отдельно для столбцов и отдельно для строк. Сначала в таблицу записывается текст сообщения, а потом поочередно переставляются столбцы, а затем строки. При расшифровании порядок перестановок должен быть обратным.

Пример выполнения шифрования методом двойной перестановки показан на рисунке 12.3.

Рис. 12.3 - Пример выполнения шифрования методом двойной перестановки

Если считывать шифртекст из правой таблицы построчно блоками по четыре буквы, то получится следующее:

ТЮАЕ ООГМ РЛИП ОЬСВ

Ключом к шифру двойной перестановки служит последовательность номеров столбцов и номеров строк исходной таблицы (в нашем примере последовательности 4132 и 3142 соответственно).

Число вариантов двойной перестановки быстро возрастает при увеличении размера таблицы:

- для таблицы 3x3 - 36 вариантов;
- для таблицы 4x4 - 576 вариантов;
- для таблицы 5x5 - 14400 вариантов.

Однако двойная перестановка не отличается высокой стойкостью и сравнительно просто "взламывается" при любом размере таблицы шифрования.

12.2.2 Система шифрования Цезаря

Шифр Цезаря является частным случаем шифра простой замены (одноалфавитной подстановки). Свое название этот шифр получил по имени римского императора Гая Юлия Цезаря, который использовал этот шифр при переписке с Цицероном (около 50 г. до н.э.).

При шифровании исходного текста каждая буква заменялась на другую букву того же алфавита по следующему правилу. Заменяющая буква определялась путем смещения по алфавиту от исходной буквы на *K* букв. При достижении конца алфавита выполнялся циклический переход к его началу. Цезарь использовал шифр замены при смещении *K*=3. Такой шифр замены можно задать таблицей подстановок, содержащей соответствующие пары букв открытого текста и шифртекста. Совокупность возможных подстановок для *K*= 3 показана на рисунке 12.4.

Одноалфавитные подстановки (K = 3, m = 26)

A → D	J → M	S → V
B → E	K → N	T → W
C → F	L → O	U → X
D → G	M → P	V → Y
E → H	N → Q	W → Z
F → I	O → R	X → A
G → J	P → S	Y → B
H → K	Q → T	Z → C
I → L	R → U	

Рис. 12.4 - Совокупность возможных подстановок для *K*= 3

Например, послание Цезаря:

VENI VIDI VICI

(в переводе на русский означает "Пришел, Увидел, Победил"), направленное его другу Аминтию после победы над понтийским царем Фарнаком, сыном Митридата, выглядело бы в зашифрованном виде так:

YHQL YLGL YLFL

Достоинством системы шифрования Цезаря является простота шифрования и расшифрования.

К недостаткам системы Цезаря следует отнести следующие:

- подстановки, выполняемые в соответствии с системой Цезаря, не маскируют частот появления различных букв исходного открытого текста;
- сохраняется алфавитный порядок в последовательности заменяющих букв; при изменении значения *K* изменяются только начальные позиции такой последовательности;
- число возможных ключей *K* мало;
- шифр Цезаря легко вскрывается на основе анализа частот появления букв в шифртексте.

Криптоаналитическая атака против системы одноалфавитной замены начинается с подсчета частот появления символов: определяется число появлений каждой буквы в шифртексте. Затем полученное распределение частот букв в шифртексте сравнивается с распределением частот букв в алфавите исходных сообщений, например в английском. Буква с наивысшей частотой появления в шифртексте заменяется на букву с наивысшей частотой появления в английском языке и т.д. Вероятность успешного вскрытия системы шифрования повышается с увеличением длины шифртекста.

12.2.3 Аффинная система подстановок Цезаря

В данном преобразовании буква, соответствующая числу *t*, заменяется на букву, соответствующую числовому значению *(at +b)* по модулю *m*.

Следует заметить, что преобразование $E_{a,b}(t)$ является взаимно однозначным отображением на множестве $\overline{Z}_m$ только в том случае, если наибольший общий делитель чисел a и т, обозначаемый как *НОД (a, m)*, равен единице, т.е. a и m должны быть взаимно простыми числами.

Например, пусть $m = 26$, $a = 3$, $b = 5$. Тогда, очевидно, *НОД (3, 26) = 1*, и мы получаем следующее соответствие между числовыми кодами букв:

t	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
3t+5	5	8	11	14	17	20	23	0	3	6	9	12	15	18	21	24	1	4	7	10	13	16	19	22	25	2

Преобразуя числа в буквы английского языка, получаем следующее соответствие для букв открытого текста и шифртекста:

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
F	I	L	O	R	U	X	A	D	G	J	M	P	S	V	Y	B	E	H	K	N	Q	T	W	Z	C

Исходное сообщение **HOPE** преобразуется в шифртекст **AVYR**

Достоинством аффинной системы является удобное управление ключами - ключи шифрования и расшифрования представляются в компактной форме в виде пары чисел *(a, b)*. Недостатки аффинной системы аналогичны недостаткам системы шифрования Цезаря.

12.2.4 Система Цезаря с ключевым словом

Система шифрования Цезаря с ключевым словом является одноалфавитной системой подстановки. Особенностью этой системы является использование ключевого слова для смещения и изменения порядка символов в алфавите подстановки.

Выберем некоторое число k, $0<k<25$, и слово или короткую фразу в качестве *ключевого слова.* Желательно, чтобы все буквы ключевого слова были различными. Пусть выбраны слово **DIPLOMAT** в качестве ключевого слова и число $k = 5$.

Ключевое слово записывается под буквами алфавита, начиная с буквы, числовой код которой совпадает с выбранным числом k:

```
0 1 2 3 4 5          10          15          20          25
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
        D I P L O M A T
```

Оставшиеся буквы алфавита подстановки записываются после ключевого слова в алфавитном порядке:

```
                5
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
V W X Y Z D I P L O M A T B C E F G H J K N Q R S U
```

Теперь мы имеем подстановку для каждой буквы произвольного сообщения.

Исходное сообщение **SEND MORE MONEY**,
шифруется как **HZBY TCGZ TCBZS.**

Следует отметить, что требование о различии всех букв ключевого слова не обязательно. Можно просто записать ключевое слово (или фразу) без повторения одинаковых букв. Например, ключевая фраза **КАК ДЫМ ОТЕЧЕСТВА НАМ СЛАДОК И ПРИЯТЕН** и число $k = 3$ порождают следующую таблицу подстановок:

```
0     3
А Б В Г Д Е Ж З И Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ь Ы Ъ Э Ю Я
Ъ Э Ю К А Д Ы М О Т Е Ч С В Н Л И П Р Я Б Г Ж З Й У Ф Х Ц Ш Щ Ь
```

Несомненным достоинством системы Цезаря с ключевым словом является то, что количество возможных ключевых слов практически неисчерпаемо. Недостатком этой системы является возможность взлома шифртекста на основе анализа частот появления букв.

12.3 Шифры сложной замены

Шифры сложной замены называют многоалфавитными, так как для шифрования каждого символа исходного сообщения применяют свой шифр простой замены. Многоалфавитная подстановка последовательно и циклически меняет используемые алфавиты.

При *r*-алфавитной подстановке символ x_o исходного сообщения заменяется символом y_o из алфавита B_o, символ x_1 - символом y_1 из алфавита B_1, и так далее, символ x_{r-1} заменяется символом y_{r-1} из алфавита B_{r-1}, символ x_r заменяется символом y_r из алфавита B_0 и т.д.

Общая схема многоалфавитной подстановки для случая $r = 4$ показана на рисунке 12.5.

Входной символ:	X_0	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9
Алфавит подстановки:	B_0	B_1	B_2	B_3	B_0	B_1	B_2	B_3	B_0	B_1

Рис. 12.5 - Схема *r*-алфавитной подстановки для случая $r = 4$

Эффект использования многоалфавитной подстановки заключается в том, что обеспечивается маскировка естественной статистики исходного языка, так как конкретный символ из исходного алфавита A может быть преобразован в несколько различных символов шифровальных алфавитов B_j. Степень обеспечиваемой защиты теоретически пропорциональна длине периода r в последовательности используемых алфавитов B_j.

Многоалфавитные шифры замены предложил и ввел в практику. криптографии Леон Батист Альберти, который также был известным архитектором и теоретиком искусства. Его книга "Трактат о шифре", написанная в 1566 г., представляла собой первый в Европе научный труд по криптологии. Кроме шифра многоалфавитной замены, Альберти также подробно описал устройства из вращающихся колес для его реализации. Криптологи всего мира почитают Л. Альберти основоположником криптологии.

12.4 Одноразовая система шифрования

Почти все применяемые на практике шифры характеризуются как условно надежные, поскольку они могут быть в принципе раскрыты при наличии неограниченных вычислительных возможностей. Абсолютно надежные шифры нельзя разрушить даже при использовании неограниченных вычислительных возможностей. Существует единственный такой шифр, применяемый на практике, - одноразовая система шифрования. Характерной особенностью одноразовой системы шифрования является одноразовое использование ключевой последовательности.

Одноразовая система шифрует исходный открытый текст

$$\overline{X} = (x_0, x_1, ..., x_{n-1})$$

в шифртекст

$$\overline{Y} = (y_0, y_1, ..., y_{n-1})$$

посредством подстановки Цезаря

$$Y_i = (X_i + K_i) \bmod m, \quad 0 \le i < n,$$

где K_i - i-й элемент случайной ключевой последовательности.

Ключевое пространство $\overline{K}$ одноразовой системы представляет собой набор дискретных случайных величин из $\overline{Z}_m$ и содержит m^n значений. Процедура расшифрования описывается соотношением

$$X_i = (Y_i - K_i) \bmod m,$$

где K_i - i-й элемент той же самой случайной ключевой последовательности.

Одноразовая система изобретена в 1917 г. американцами Дж. Моборном и Г. Вернамом. Для реализации этой системы подстановки иногда используют одноразовый блокнот. Этот блокнот составлен из отрывных страниц, на каждой из которых напечатана таблица со случайными числами (ключами) K_j. Блокнот выполняется в двух экземплярах: один используется отправителем, а другой - получателем. Для каждого символа X_j сообщения используется свой ключ K_j из таблицы только один раз. После того как таблица использована, она должна быть удалена из блокнота и уничтожена. Шифрование нового сообщения начинается с новой страницы.

Этот шифр абсолютно надежен, если набор ключей K_i действительно случаен и непредсказуем. Если криптоаналитик попытается использовать для заданного шифртекста все возможные наборы ключей и восстановить все возможные варианты исходного текста, то они все окажутся равновероятными. Не существует способа выбрать исходный текст, который был действительно послан. Теоретически доказано, что одноразовые системы являются не-раскрываемыми системами, поскольку их шифртекст не содержит достаточной информации для восстановления открытого текста.

Казалось бы, что благодаря данному достоинству одноразовые системы следует применять во всех случаях, требующих абсолютной информационной безопасности. Однако возможности Применения одноразовой системы ограничены чисто практическими аспектами. Существенным моментом является требование одноразового использования случайной ключевой последовательности. Ключевая последовательность с длиной, не меньшей длины сообщения, должна передаваться получателю сообщения заранее или отдельно по некоторому секретному каналу. Это требование не будет слишком обременительным для передачи действительно важных одноразовых сообщений, например, по горячей линии Вашингтон-Москва. Однако такое требование практически неосуществимо для

современных систем обработки информации, где требуется шифровать многие миллионы символов.

12.5 Шифрование методом гаммирования

Под ***гаммированием*** понимают процесс наложения по определенному, закону гаммы шифра на открытые данные.

Гамма шифр - *это псевдослучайная последовательность, выработанная по заданному алгоритму для зашифрования открытых данных и расшифрования зашифрованных данных.*

Процесс зашифрования заключается в генерации гаммы шифра и наложении полученной гаммы на исходный открытый текст обратимым образом, например с использованием операции сложения по модулю 2.

Следует отметить, что перед зашифрованием открытые данные разбивают на блоки $T_0^{(i)}$ одинаковой длины, обычно по 64 бита. Гамма шифр вырабатывается в виде последовательности блоков $Г_ш^{(i)}$ аналогичной длины.

Уравнение зашифрования можно записать в виде

$$T_ш^{(i)} = Г_ш^{(i)} \oplus T_0^{(i)}, i = 1...M,$$

где $T_ш^{(i)}$ - i-й блок шифртекста;
$Г_ш^{(i)}$ - i-й блок гаммы шифра;
$T_0^{(i)}$ - i-й блок открытого текста;
M - количество блоков открытого текста.

Процесс расшифрования сводится к повторной генерации Гаммы шифра и наложению этой гаммы на зашифрованные данные. Уравнение расшифрования имеет вид

$$T_0^{(i)} = Г_ш^{(i)} \oplus T_ш^{(i)}.$$

Получаемый этим методом шифр-текст достаточно труден для раскрытия, поскольку теперь ключ является переменным. По сути дела гамма шифра должна изменяться случайным образом для каждого шифруемого блока. Если период гаммы превышает длину всего шифруемого текста и злоумышленнику неизвестна никакая часть исходного текста, то такой шифр можно раскрыть только прямым перебором всех вариантов ключа. В этом случае крипто-стойкость шифра определяется длиной ключа.

12.6 Стандарт шифрования данных DES

Стандарт шифрования данных DES (Data Encryption Standard) опубликован в 1977 г. Национальным бюро стандартов США. Стандарт DES предназначен для защиты от несанкционированного доступа к важной, но несекретной информации в государственных и коммерческих организациях США. Алгоритм, положенный в основу стандарта, распространялся

достаточно быстро, и уже в 1980 г. был одобрен Национальным институтом стандартов и технологий США (НИСТ). С этого момента DES превращается в стандарт не только по названию (Data Encryption Standard), но и фактически. Появляются программное обеспечение и специализированные микро-ЭВМ, предназначенные для шифрования и расшифрования информации в сетях передачи данных.

К настоящему времени DES является наиболее распространенным алгоритмом, используемым в системах защиты коммерческой информации. Более того, реализация алгоритма DES в таких системах становится признаком хорошего тона.

Основные достоинства алгоритма DES:

- *используется только один ключ длиной 56 бит;*
- *зашифровав сообщение с помощью одного пакета программ, для расшифровки можно использовать любой другой пакет программ. соответствующий стандарту DES;*
- *относительная простота алгоритма обеспечивает высокую скорость обработки;*
- *достаточно высокая стойкость алгоритма.*

Алгоритм DES использует комбинацию подстановок и перестановок. DES осуществляет шифрование 64-битовых блоков данных с помощью 64-битового ключа, в котором значащими являются 56 бит (остальные 8 бит - проверочные биты для контроля на четность). Дешифрование в DES является операцией, обратной Шифрованию, и выполняется путем повторения операций шифрования в обратной последовательности. Обобщенная схема процесса шифрования в алгоритме DES показана на рисунке 12.6 в.

Процесс шифрования заключается в начальной перестановке битов 64-битового блока, шестнадцати циклах шифрования и, наконец, в конечной перестановке битов (рисунок 15.6, в). Следует сразу отметить, что все приводимые таблицы (на рисунке 15.6, а,б) являются стандартными и должны включаться в реализацию алгоритма DES в неизменном виде. Все перестановки и коды в таблицах подобраны разработчиками таким образом, чтобы максимально затруднить процесс расшифровки путем подбора ключа.

При описании алгоритма DES (рисунок 15.6, в) применены следующие обозначения:

1. L и R - последовательности битов (левая (left) и правая (right));
2. LR - конкатенация последовательностей L и R, т.е. такая последовательность битов, длина которой равна сумме длин L и R; в последовательности LR биты последовательности R следуют за битами последовательности L;
3. $\oplus$ -операция побитового сложения по модулю 2.

Матрица начальной перестановки IP

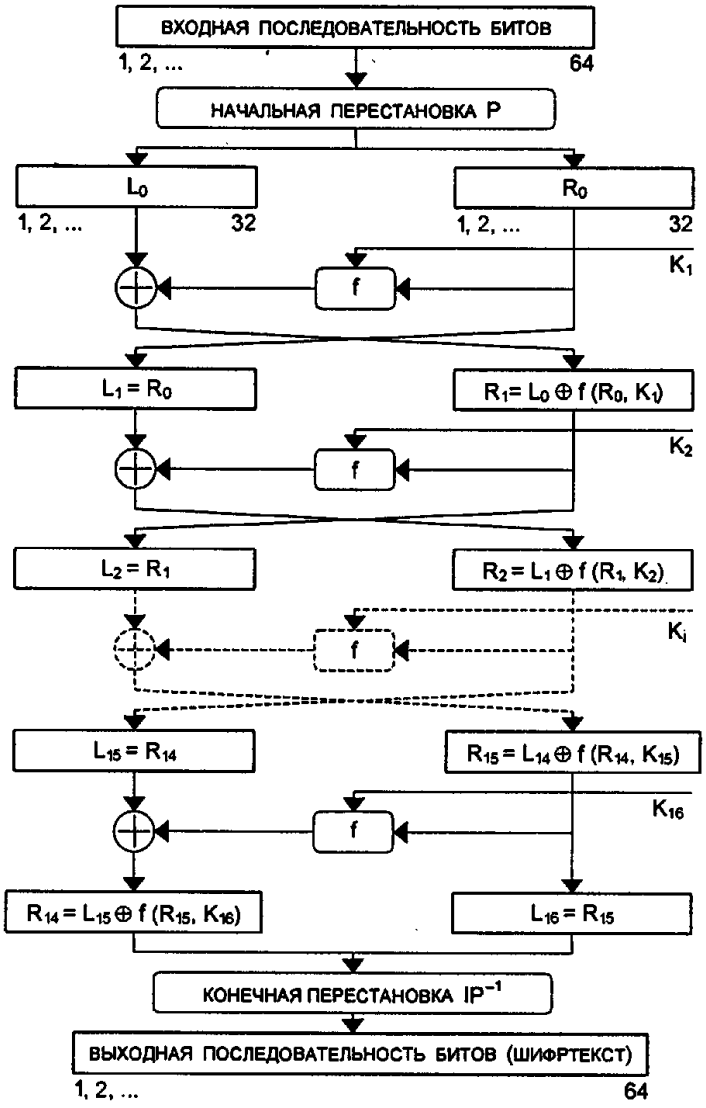
58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

а.

Матрицв обратной перестановки IP^{-1}

40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25

б.



в

Рисунок 12.6 – Таблицы перестановки и алгоритм DES

Пусть из файла исходного текста считан очередной 64-битовый (8-байтовый) блок ***T***. Этот блок ***T*** преобразуется с помощью *матрицы начальной перестановки* ***IP*** (рисунок 12.6, а).

Биты входного блока ***T*** (64 бита) переставляются в соответствии с матрицей ***IP***: бит 58 входного блока ***T*** становится битом 1, бит 50-битом 2 и т.д. Эту перестановку можно описать выражением $\boldsymbol{T_0 = IP(T)}$. Полученная последовательность битов $\boldsymbol{T_0}$ разделяется на две последовательности: $\boldsymbol{L_0}$ - левые или старшие биты, $\boldsymbol{R_0}$ - правые или младшие биты, каждая из которых содержит 32 бита.

Затем выполняется итеративный процесс шифрования, состоящий из 16 шагов (циклов). Пусть $\boldsymbol{T_i}$ - результат *i*-й итерации:

$$T_i = L_i R_i$$

где $L_i = t_1 t_2 ... t_{32}$ (первые 32 бита); $R_i = t_{33} t_{34} ... t_{64}$ (последние 32 бита).

Тогда результат *i*-й итерации описывается следующими формулами:

$$L_i = R_{i-1},\ i = 1,2,...,16;\quad R_i = L_{i-1} \oplus f(R_{i-1}, K_i),\ i = 1,2,...,16.$$

Функция ***f*** называется функцией шифрования. Ее аргументами являются последовательность $\boldsymbol{R_{i-1}}$, получаемая на предыдущем шаге итерации, и 48-битовый ключ $\boldsymbol{K_i}$, который является результатом преобразования 64-битового ключа шифра ***K***. (Подробнее функция шифрования f и алгоритм получения ключа $\boldsymbol{K_i}$ описаны ниже.) На последнем шаге итерации получают последовательности $\boldsymbol{R_{16}}$ и $\boldsymbol{L_{16}}$ (без перестановки местами), которые конкатенируются в 64-битовую последовательность $\boldsymbol{R_{16}L_{16}}$

По окончании шифрования осуществляется восстановление позиций битов с помощью матрицы обратной перестановки $\boldsymbol{IP^{-1}}$ (рисунок 12.6, б).

Процесс расшифрования данных является инверсным по отношению к процессу шифрования. Все действия должны быть выполнены в обратном порядке. Это означает, что расшифровываемые данные сначала переставляются в соответствии с матрицей $\boldsymbol{IP^{-1}}$, а затем над последовательностью битов $\boldsymbol{R_{16}L_{16}}$ выполняются те же действия, что и в процессе шифрования, но в обратном порядке:

$$R_{i-1} = L_i,\ i = 1,2,...,16;\quad L_{i-1} = R_i \oplus f(L_i, K_i),\ i = 1,2,...,16.$$

В настоящее время блочный алгоритм DES считается относительно безопасным алгоритмом шифрования. Он подвергался тщательному криптоанализу в течение 20 лет, и самым практичным способом его взламывания является метод перебора всех возможных вариантов ключа. Ключ DES имеет длину 56 бит, поэтому существует 2^{55} возможных вариантов такого ключа. Если предположить, что суперкомпьютер может испытать миллион вариантов ключа за секунду, то потребуется 2285 лет для нахождения правильного ключа. Если бы ключ имел длину 128 бит, то потребовалось бы 10^{25} лет (для сравнения: возраст Вселенной около 10^{10}лет).

ЧАСТЬ 4. ВРЕДОНОСТНЫЕ ПРОГРАММЫ

13. ВРЕДОНОСТНЫЕ ПРОГРАММЫ И КОМПЬЮТЕРНЫЕ ВИРУСЫ

13.1 Основные понятия

Для хранения информации на любом компьютере используются два вида памяти - постоянные запоминающие устройства (ПЗУ) и постоянные запоминающие устройства (ПЗУ). К первому типу относятся жесткие диски (винчестеры), дискеты, CD-ROM и другие мобильные носители информации, ко второму - оперативная память, то есть микросхема в системном блоке. Поэтому ПЗУ также называют внешней долговременной памятью, а ОЗУ - внутренней. Главное отличие оперативной памяти от внешней состоит в том, что информация, записанная на ОЗУ, может храниться только во время работы компьютера, при выключении или перезагрузке она теряется. Как следствие, большинство устройств ПЗУ предназначены для хранения значительно большего объема информации, чем оперативная память.

Для организации хранилища информации на ПЗУ используются файлы.

Файл *- это логический блок информации, хранимой на носителях информации.* Файл обязательно имеет имя и может содержать произвольный объем информации. Максимальная длина имени и максимальный объем файла определяются файловой системой.

Файловая система *- это совокупность правил, определяющих систему хранения информации*: различные атрибуты файлов, такие как максимальная длина имени, максимальный допустимый размер файла. Примеры файловых систем - FAT, FAT32, NTFS, EXT2, ISO9660.

Компьютерная программа*- это последовательность инструкций (команд) для выполнения компьютером определенных действий. Программы записываются при помощи специальных языков программирования или машинного кода.* Примеры компьютерных программ - программа чтения и записи данных на дискету, программа воспроизведения музыки с диска, записная книжка в мобильном телефоне, Microsoft Word.

Передача программе пользовательских данных может осуществляться с помощью графического интерфейса, командной строки, конфигурационного файла или косвенно через другие программы.

Конфигурационный файл *представляет собой текстовый файл с последовательным перечнем данных и команд, которые необходимо передать программе.* При взаимодействии же двух программ между собой пользователь как правило явного участия не принимает.

Вызов компьютерной программы*, то есть запуск программы на выполнение, производится путем последовательной загрузки содержимого соответствующего ей файла в оперативную память, после чего компьютер начинает выполнять последовательность заложенных в эту программу действий.*

Запустить программу можно также непрямым методом. Например, при доступе к любому файлу, содержащему текстовую информацию, должна запускаться программа, позволяющая его прочесть, то есть преобразовывающая машинный код, содержащийся в текстовом файле, в буквы, которые пользователь прочитает на экране.

Таким образом, практически все программы помимо основных функций, выполняют ряд дополнительных, служебных действий, не видимых обычному пользователю.

Вредоносная программа *- это программа, наносящая какой-либо вред компьютеру, на котором она запускается, или другим подключенным к нему компьютерам.*

Одним из способов для вредоносной программы оставаться незамеченной на компьютере является дописывание своего кода к файлу другой известной программы. При этом возможно как полное перезаписывание файлов (но в этом случае вредоносная программа обнаруживает себя при первом же запуске, поскольку ожидаемые действия полностью заменены), так и внедрение в начало, середину или конец файла.

Пример. CIH - вирус, который в ходе заражения записывает свои копии во все запускаемые пользователем программные файлы (PE EXE). Внедрение может происходить как одним куском, так и путем деления вредоносного кода на блоки и записи их в разных частях заражаемого файла. При этом инфицированная программа может дальше выполнять свои основные функции и вирус в ней никак себя не обнаруживает. Однако в определенный момент времени происходит уничтожение всей информации на жестком диске. Поскольку самая известная версия CIH срабатывала 26 апреля, то он получил второе имя - «Чернобыль».

13.2 Способы распространения вредоносных программ

В настоящее время имеется четыре основных способа передачи вредоносного ПО.

1. Мобильные носители. *К мобильным носителям можно отнести все виды энергонезависимых ПЗУ. То есть таких устройств, которые позволяют достаточно долго хранить информацию и при этом не требуют дополнительного питания от компьютера.* Это дискеты, компакт диски, flash-накопители, перфокарты и перфоленты.

Мобильные носители - достаточно распространенный способ для размножения компьютерных вирусов. Однако по скорости распространения этот путь существенно уступает компьютерным сетям.

2. Локальная вычислительная сеть *(ЛВС)*[2)] *- это компьютерная сеть, покрывающая относительно небольшую территорию* (дом, школу, институт, микрорайон).

Вредоносные программы в полной мере используют преимущества ЛВС - фактически, почти все современные вирусы имеют встроенные процедуры инфицирования по локальным сетям и как следствие высокие темпы распространения. Инфицирование обычно происходит в такой последовательности. Зараженный компьютер с заданным интервалом инициирует соединение поочередно со всеми другими компьютерами сети и проверяет наличие на них открытых для общего доступа файлов. Если такие есть, происходит инфицирование.

3. Глобальная вычислительная сеть *(ГВС) - это компьютерная сеть, покрывающая большие территории* - города, страны, континенты. Самая большая и самая известная на сегодняшний день глобальная вычислительная сеть - это всемирная сеть Интернет. *Наличие сети такого масштаба делает возможным всемирные эпидемии компьютерных вирусов.*

Пример. 30 апреля 2004 года были обнаружены первые экземпляры вируса Sasser - в течение дня им было атаковано около 4 тысяч компьютеров, что вызвало серьезные сбои в работе таких компаний как Postbank, Delta Air Lines, Goldman Sachs. Впоследствии было поражено более 8 млн. компьютеров, а убытки от Sasser были оценены в 979 млн. долларов США.

4. Электронная почта *- это способ передачи информации в компьютерных сетях, основанный на пересылке пакетов данных, называемых электронными письмами.*

На сегодняшний день электронная почта выступает основным путем распространения вирусов. Это происходит потому, что время доставки письма очень мало (обычно исчисляется минутами) и практически все пользователи Интернет имеют как минимум один почтовый ящик. При этом для того, чтобы доставить пользователю на компьютер зараженный файл, не нужно его принуждать куда-либо обратиться и скопировать к себе вирус. Достаточно лишь прислать на его электронный адрес инфицированное письмо и заставить адресата его открыть. Часто для инфицирования даже не требуется запускать вложение - существуют методы, позволяющие заражать даже при обычном прочтении письма.

Пример 1. Horillka распространяется через Интернет в виде файлов, прикрепленных к зараженным письмам с такими параметрами: заголовок - «Внимание!», текст: «Выпущено новое vbs обновление для поиска вирусов в памяти ОС Windows! Оно помогает бороться с вирусами, рассылающимися по почте. Антивирусный модуль написан на скрипт-языке, что помогает перехватывать vb и js вирусы, прежде чем они начнут деструктивную деятельность. Достаточно открыть файл и программа по устранению вирусов проведет поиск вредоносных программ в памяти компьютера». Во вложении находится файл с именем «WinSys32dll.vbs», после его запуска происходит заражение компьютера. Как результат, 11 декабря каждого года на экран

выдается сообщение «COOOOOOOOL» и после следующей перезагрузки уничтожаются все данные на жестком диске C.

Пример 2. LoveLetter в мае 2000 года в течение всего нескольких часов заразил миллионы компьютеров по всему миру. Такому успеху способствовала удачно выбранная тема, интригующий текст и имя вложенного файла - «ILOVEYOU», «kindly check the attached LOVELETTER coming from me» и «LOVE-LETTER-FOR-YOU.TXT.vbs. После заражения происходила кража конфиденциальной информации и искажение содержимого некоторых файлов на жестком диске.

13.3 Операционная система. Уязвимости и заплаты

Все программы можно разделить на два типа - прикладные и системные.

Прикладное программное обеспечение *(прикладные программы) - это программы, предназначенные для выполнения определенных пользовательских задач и рассчитанные на непосредственное взаимодействие с пользователем.* Прикладные программы часто называют приложениями.

Системное программное обеспечение *используется для обеспечения работы компьютера самого по себе и выполнения прикладных программ.*

В персональном компьютере под прикладными программами понимаются различные текстовые редакторы, игры, почтовые программы, электронные словари. Роль базового системного программного обеспечения играет операционная система.

Операционная система *(ОС) - это комплекс программ, который обеспечивает управление физическими устройствами компьютера, доступ к файлам, ввод и вывод данных, выполнение и взаимодействие пользовательских программ.* Наличие автозагрузки дает возможность вредоносным вирусам практически незаметно выполнять свои функции. Для этого во время заражения в список автозагрузки добавляется ссылка на программу, которая загружает вирус в оперативную память при каждой загрузке операционной системы. То есть фактически активация вируса происходит без участия пользователя при каждом включении компьютера.

Уязвимость *(или брешь в системе безопасности) - это место в программном коде, которое теоретически или реально может быть использовано для несанкционированного доступа к управлению программой. Уязвимости могут появляться как в системном, так и в прикладном программном обеспечении.*

После обнаружения уязвимости, производители программ обычно стараются как можно скорее выпустить дополнения, которые бы исправляли исходный код и закрывали брешь.

Заплата *или* ***патч*** *(от англ. patch - латать, ставить заплаты) - это программный код, используемый для модификации используемой программы.*

Другими словами заплата - это дополнительная программа, которую следует запустить на выполнение, если в уже используемой программе обнаружилась ошибка или уязвимость. При этом часто можно устанавливать патч без удаления основной программы и даже без завершения ее работы - в первую очередь это касается операционных систем.

Пример. В январе 2003 года началась эпидемия Slammer, заражающего сервера под управлением операционной системы Microsoft SQL Server 2000. Вирус использовал брешь в системе безопасности SQL Server, заплата к которой вышла еще в июле 2002. После проникновения Slammer начинал в бесконечном цикле посылать свой код на случайно выбранные адреса в сети - только за первые 10 минут было поражено около 90% (120 000 единиц) всех уязвимых серверов, при этом пять из тринадцати главных серверов Интернет вышли из строя.

13.4 Последствия заражений вредоносной программой

Последствия инфицирования компьютера вредоносной программой могут быть как явными, так и неявными.

***К неявным последствиям обычно относят** заражения программами, которые по своей сути являются вирусами, однако из-за ошибок в своем коде или нестандартному программному обеспечению целевого компьютера, вредоносную нагрузку выполнить не могут. При этом свое присутствие в системе они никак не выражают.*

Класс явных последствий постоянно увеличивается. К ним можно отнести:

***1. Несанкционированная рассылка электронных писем.** Ряд вирусов после заражения компьютера ищут на жестком диске файлы, содержащие электронные адреса и без ведома пользователя начинают рассылку по ним инфицированных писем.*

Пример. Sircam рассылал себя с зараженных компьютеров в виде файлов, вложенных в письма электронной почты. Для этого случайным образом на жестком диске выбирался файл, к которому прикреплялся вирусный код (дописывался в конец файла). Таким образом отсылаемые письма содержали вложение, состоящее из двух частей: вирус и файл-приманку. Имя вложения формировалось на основе выбранного файла - например, если исходный файл назывался photos.zip, то имя вложения было - photos.zip.pif, photos.zip.lnk, photos.zip.bat или photos.zip.com. Адреса получателей выбирались из найденных на зараженном компьютере, а текст писем составлялся так, чтобы внушить как можно меньше подозрений и заставить адресата запустить полученный файл. Побочным эффектом такого способа распространения является утечка с зараженного компьютера конфиденциальных документов.

2. Кража конфиденциальной информации. *После инфицирования вирус ищет файлы, содержащие конфиденциальную информацию (номера кредитных карт, различные пароли, секретные документы), для кражи которой он предназначен, и передает ее хозяину.* Это может происходить путем отправки выбранных данных в электронном сообщении на определенный адрес или прямой пересылки их на удаленный сервер.

3. Несанкционированное использование сетевых ресурсов. *Существуют вирусы, которые после заражения без ведома пользователя подключаются к различным платным службам с использованием личных данных, найденных на компьютере.* Впоследствии жертве приходится оплачивать не заказанные ею услуги, а злоумышленник обычно получает процент от этого счета.

Пример. Dialer - после попадания на компьютер, этот вирус начинал дозвон на международные телефонные номера для подключения к платным сервисам. Через некоторое время пользователю приходил огромный телефонный счет и доказать в подавляющем большинстве случаев что он никуда не звонил не представлялось возможным.

4. Удаленное управление компьютером. *После того, как произошло заражение, некоторые вирусы передают своему хозяину инструменты для удаленного управления инфицированным компьютером - открывают бекдоры (от англ. backdoor - черный ход). Обычно это выражается в возможности удаленно запускать размещенные на нем программы, а также загружать из Интернет по желанию злоумышленника любые файлы.* Свое присутствие такие программы обычно выражают только в использовании части ресурсов зараженного компьютера для своих нужд - в основном процессора и оперативной памяти. Такие компьютеры часто называют машинами-зомби.

5. Ботнеты. *Группа компьютеров, которыми централизованно управляет один злоумышленник, называется ботнетом.* Число таких компьютеров в Интернет на сегодняшний день достигает нескольких миллионов и продолжает увеличиваться каждый день.

Пример. Bagle - вирус, распространяющийся в виде вложения в электронные письма. Адрес отправителя и имя вложения - произвольные, тема - «Hi», текст - «Test =)». После заражения он копирует себя на жесткий диск под именем bbeagle.exe и регистрирует этот файл в автозапуске операционной системы. Далее происходят попытки соединиться с несколькими удаленными серверами. При этом злоумышленнику предоставляется возможность загружать на зараженный компьютер любые файлы и запускать их на выполнение. Первый вирус из этой серии, Bagle.a, был обнаружен 18 января 2004, однако по замыслу автора уже через 10 дней он перестал размножаться и вскоре появились новые, более совершенные

модификации Bagle. В результате автор получил огромную сеть подконтрольных ему компьютеров. Bagle-ботнет - одна из самых масштабных и известных сетей машин-зомби.

6. Несанкционированная атака на чужой сервер. Последнее время вирусописатели используют ботнеты для организации так называемых DoS-атак. DoS (от англ. Denial of Service) - это построенное на принципе отказа в обслуживании нападение на удаленный сайт. Это означает, что каждый инфицированный компьютер периодически (с интервалом обычно порядка 1 секунды) посылает произвольный запрос на получение информации с заданного злоумышленником сайта. Все веб-сайты рассчитаны на определенное число запросов в единицу времени, поэтому резкое увеличение нагрузки практически всегда выводит сервер из строя. Атака, которая производится одновременно с большого количества компьютеров, называется распределенной DoS-атакой или DDoS (от англ. Distributed Denial of Service).

Пример.Одна из самых известных DDoS-атак была предпринята в июле 2001 года. Объектом нападения стал веб-сайт Белого дома в США (www.whitehouse.gov). В атаке участвовало около 12000 (по другим данным - до 200000) компьютеров, зараженных во время прошедшей незадолго до этого эпидемии вируса CodeRed.

7. Рассылка спама*. Под этим термином обычно понимается ненужная, нежелательная, не запрошенная получателем корреспонденция.* Спам может приходить как по электронной почте, так и в виде других сообщений, например на мобильный телефон в виде SMS. Поскольку электронных адресов в Интернет очень много, рассылка спама занимает много ресурсов. Поэтому злоумышленники часто используют для этих целей ботнеты.

8. Фишинг. *Фактически фишинг - это метод кражи чужой информации, суть которого заключается в подделке известного сайта и рассылке электронных писем-приглашений зайти на него и ввести свою конфиденциальную информацию.*

Например, создается точная копия сайта какого-либо банка и с помощью спам-технологий рассылается письмо, максимально похожее на настоящее, с уведомлением о сбое в программном обеспечении и просьбой зайти на сайт и заново ввести свои данные. Тут же, в письме приводится адрес сайта - естественно, поддельный, но также максимально похожий на правду. Существует международная организация, ведущая учет фишинговым инцидентам - Anti-Phishing Working Group (www.antiphishing.org).

9. Уничтожение информации. *Большинство современных вредоносных программ если и несут в себе процедуры уничтожения*

информации на компьютере-жертве, то только в качестве дополнительной, не основной функции. Однако для многих пользователей это наиболее явное и болезненное последствие - удаленным и не подлежащим восстановлению может оказаться любой файл на жестком диске, как детские фотографии, так и только что законченная курсовая работа или книга.

10. Мистификации. Иногда на электронную почту или по другим каналам приходят так называемые предупреждения о новых вирусах. Обычно они содержат призывы не ходить по приведенным ссылкам, проверить свой компьютер на наличие на нем вируса указанным в сообщении методом или предостережение не принимать почту с определенными параметрами. Чаще всего это просто мистификация. Вреда, если не предпринимать указанные действия и не пересылать всем друзьям и знакомым, нет.

Пример. В апреле 2004 года произошла массовая рассылка предупреждения о якобы опасном вирусе, основным признаком присутствия которого на компьютерах под управлением операционной системы Microsoft Windows заявлялось наличие файла jdbgmgr.exe, который и содержит саму вредоносную программу. В действительности же этот файл является стандартной программой, входящей в большинство версий Microsoft Windows. Удаление или изменение содержимого jdbgmgr.exe влечет непредсказуемые последствия в работоспособности операционной системы.

13.5 Классификация вредоностных программ

Все вредоносные программы в соответствии со способами распространения и вредоносной нагрузкой можно разделить на четыре основные типа:

- *компьютерные вирусы,*
- *черви,*
- *трояны*
- *другие программы.*

Рассмотрим основные особенности указанных типов подробнее.

13.5.1 Вирусы

Основная черта компьютерного вируса - это способность к саморазмножению.

Компьютерный вирус- *это программа, способная создавать свои дубликаты (не обязательно совпадающие с оригиналом) и внедрять их в вычислительные сети и/или файлы, системные области компьютера и прочие выполняемые объекты. При этом дубликаты сохраняют способность к дальнейшему распространению.*

Условно жизненный цикл любого компьютерного вируса можно разделить на пять стадий:

- *Проникновение на чужой компьютер.*

- *Активация.*
- *Поиск объектов для заражения.*
- *Подготовка копий.*
- *Внедрение копий.*

Путями проникновения вируса могут служить как мобильные носители, так и сетевые соединения - фактически, все каналы, по которым можно скопировать файл. Однако в отличие от червей, вирусы не используют сетевые ресурсы - заражение вирусом возможно, только если пользователь сам каким-либо образом его активировал. Например, скопировал или получил по почте зараженный файл и сам его запустил или просто открыл.

После проникновения следует активация вируса. Это может происходить несколькими путями.

В соответствии с выбранным методом активации вирусы делятся на следующие виды:

1. Загрузочные вирусы *заражают загрузочные сектора жестких дисков и мобильных носителей.*

Примеры. Вредоносная программа Virus.Boot.Snow.a записывает свой код в MBR жесткого диска или в загрузочные сектора дискет. При этом оригинальные загрузочные сектора шифруются вирусом. После получения управления вирус остается в памяти компьютера (резидентность) и перехватывает прерывания INT 10h, 1Ch и 13h. Иногда вирус проявляет себя визуальным эффектом - на экране компьютера начинает падать снег.

Другой загрузочный вирус Virus.Boot.DiskFiller также заражает MBR винчестера или загрузочные сектора дискет, остается в памяти и перехватывает прерывания - INT 13h, 1Ch и 21h. При этом, заражая дискеты, вирус форматирует дополнительную дорожку с номером 40 или 80 (в зависимости от объема дискеты он может иметь 40 либо 80 дорожек с номерами 0-39 или 0-79 соответственно). Именно на эту нестандартную дорожку вне поля обычной видимости вирус записывает свой код, добавляя в загрузочный сектор лишь небольшой фрагмент - головную часть вируса.

2. Файловые вирусы *- заражают файлы. Отдельно* ***по типу среды обитания в этой группе также выделяют следующие типы****.*

2.1 Классические файловые вирусы *- они различными способами внедряются в исполняемые файлы (внедряют свой вредоносный код или полностью их перезаписывают), создают файлы-двойники, свои копии в различных каталогах жесткого диска или используют особенности организации файловой системы.*

Пример. Самый известный файловый вирус всех времен и народов — Virus.Win9x.CIH, известный также как «Чернобыль». Имея небольшой размер - около 1 кб - вирус заражает PE-файлы (Portable Executable) на

компьютерах под управлением операционных систем Windows 95/98 таким образом, что размер зараженных файлов не меняется. Для достижения этого эффекта вирус ищет в файлах «пустые» участки, возникающие из-за выравнивания начала каждой секции файла под кратные значения байт. После получения управления вирус перехватывает IFS API, отслеживая вызовы функции обращения к файлам и заражая исполняемые файлы. 26 апреля срабатывает деструктивная функция вируса, которая заключается в стирании Flash BIOS и начальных секторов жестких дисков. Результатом является неспособность компьютера загружаться вообще (в случае успешной попытки стереть Flash BIOS) либо потеря данных на всех жестких дисках компьютера.

***2.2 Макровирусы**, которые написаны на внутреннем языке, так называемых макросах какого-либо приложения.* Подавляющее большинство макровирусов используют макросы текстового редактора Microsoft Word.

Пример. Одними из наиболее разрушительных макровирусов являются представители семейства Macro.Word97.Thus. Эти вирусы содержат три процедуры Document_Open, Document_Close и Document_New, которыми подменяет стандартные макросы, выполняющиеся при открытии, закрытии и создании документа, тем самым обеспечивая заражение других документов. 13 декабря срабатывает деструктивная функция вируса - он удаляет все файлы на диске C:, включая каталоги и подкаталоги.

***2.3 Скрипт-вирусы**, написанные в виде скриптов для определенной командной оболочки* - например, bat-файлы для DOS или VBS и JS - скрипты для Windows Scripting Host (WSH).

Пример. Virus.VBS.Sling написан на языке VBScript (Visual Basic Script). При запуске он ищет файлы с расширениями .VBS или .VBE и заражает их. При наступлении 16-го июня или июля вирус при запуске удаляет все файлы с расширениями .VBS и .VBE, включая самого себя.

Дополнительным отличием вирусов от других вредоносных программ служит их жесткая привязанность к операционной системе или программной оболочке, для которой каждый конкретный вирус был написан. Это означает, что вирус для Microsoft Windows не будет работать и заражать файлы на компьютере с другой установленной операционной системой, например Unix. Точно также макровирус для Microsoft Word 2003 скорее всего не будет работать в приложении Microsoft Excel 97.

При подготовке своих вирусных копий для маскировки от антивирусов могут применяться такие технологии как:

- ***Шифрование** — вирус состоит из двух функциональных кусков: собственно вирус и шифратор.* Каждая копия вируса состоит из шифратора, случайного ключа и собственно вируса, зашифрованного этим ключом.

- ***Метаморфизм*** *— создание различных копий вируса путем замены блоков команд на эквивалентные, перестановки местами кусков кода,* вставки между значащими кусками кода «мусорных» команд, которые практически ничего не делают.

Сочетание этих двух технологий приводит к появлению следующих типов вирусов классифицируемых по технологии защиты от обнаружения:

1. ***Шифрованный вирус*** *— вирус, использующий простое шифрование со случайным ключом и неизменный шифратор.* Такие вирусы легко обнаруживаются по сигнатуре шифратора.
2. ***Метаморфный вирус*** *— вирус, применяющий метаморфизм ко всему своему телу для создания новых копий.*
3. ***Полиморфный вирус*** *— вирус, использующий метаморфный шифратор для шифрования основного тела вируса со случайным ключом.* При этом часть информации, используемой для получения новых копий шифратора также может быть зашифрована. Например, вирус может реализовывать несколько алгоритмов шифрования и при создании новой копии менять не только команды шифратора, но и сам алгоритм.

Пример. Одним из наиболее сложных и относительно поздних полиморфных вирусов является Virus.Win32.Etap. При заражении файла вирус перестраивает и шифрует собственный код, записывает его в одну из секций заражаемого файла, после чего ищет в коде файла вызов функции ExitProcess и заменяет его на вызов вирусного кода. Таким образом, вирус получает управление не перед выполнением исходного кода зараженного файла, а после него.

Основные цели любого компьютерного вируса - это распространение на другие ресурсы компьютера и выполнение специальных действий при определенных событиях или действиях пользователя (например, 26 числа каждого четного месяца или при перезагрузке компьютера). Специальные действия нередко оказываются вредоносными.

13.5.2 Черви

В отличие от вирусов черви - это вполне самостоятельные программы. Главной их особенностью также является способность к саморазмножению, однако при этом они способны к самостоятельному распространению с использованием сетевых каналов. Для подчеркивания этого свойства иногда используют термин «сетевой червь».

Червь (сетевой червь) *- это вредоносная программа, распространяющаяся по сетевым каналам и способная к самостоятельному преодолению систем защиты компьютерных сетей, а также к созданию и*

дальнейшему распространению своих копий, не обязательно совпадающих с оригиналом.

Пример. Классическими сетевыми червями являются представители семейства Net-Worm.Win32.Sasser. Эти черви используют уязвимость в службе LSASS Microsoft Windows. При размножении, червь запускает FTP-службу на TCP-порту 5554, после чего выбирает IP-адрес для атаки и отсылает запрос на порт 445 по этому адресу, проверяя, запущена ли служба LSASS. Если атакуемый компьютер отвечает на запрос, червь посылает на этот же порт эксплойт уязвимости в службе LSASS, в результате успешного выполнения которого на удаленном компьютере запускается командная оболочка на TCP-порту 9996. Через эту оболочку червь удаленно выполняет загрузку копии червя по протоколу FTP с запущенного ранее сервера и удаленно же запускает себя, завершая процесс проникновения и активации.

Жизненный цикл червей состоит из таких стадий:

- *Проникновение в систему.*
- *Активация.*
- *Поиск объектов для заражения.*
- *Подготовка копий.*
- *Распространение копий.*

В зависимости от способа проникновения в систему черви делятся на типы:

- ***сетевые черви*** *используют для распространения локальные сети и Интернет;*
- ***почтовые черви*** *- распространяются с помощью почтовых программ;*
- ***IM-черви*** *используют системы мгновенного обмена сообщениями;*
- ***IRC-черви*** *распространяются по каналам IRC;*
- ***P2P-черви*** *- при помощи пиринговых файлообменных сетей.*

После проникновения на компьютер, червь должен активироваться - иными словами запуститься.

По методу активации все черви можно разделить на две большие группы.

1. ***Требующие активного участия пользователя.*** *Отличительная особенность таких является использование обманных методов. Это проявляется, например, когда получатель инфицированного файла вводится в заблуждение текстом письма и добровольно открывает вложение с почтовым червем, тем самым его активируя.*
2. ***Не требующие активного участия пользователя.*** *Активация сетевого червя без участия пользователя всегда означает, что червь использует бреши в безопасности программного обеспечении*

компьютера. Это приводит к очень быстрому распространению червя внутри корпоративной сети с большим числом станций, существенно увеличивает загрузку каналов связи и может полностью парализовать сеть. Именно этот метод активации использовали черви Lovesan и Sasser.

В последнее время наметилась тенденция к совмещению этих двух технологий - такие черви наиболее опасны и часто вызывают глобальные эпидемии.

Сетевые черви могут кооперироваться с вирусами - такая пара способна самостоятельно распространяться по сети (благодаря червю) и в то же время заражать ресурсы компьютера (функции вируса).

13.5.3 Троянские программы

Трояны или программы класса троянский конь, в отличие от вирусов и червей, не обязаны уметь размножаться.

Троян (троянский конь) *- программа, основной целью которой является вредоносное воздействие по отношению к компьютерной системе путем выполнения несанкционированных пользователем действий: кражи, порчи или удаления конфиденциальных данных, нарушения работоспособности компьютера или использования его ресурсов в неблаговидных целях.*

Некоторые трояны способны к самостоятельному преодолению систем защиты компьютерной системы, с целью проникновения в нее. Однако в большинстве случаев они проникают на компьютеры вместе с вирусом либо червем - то есть такие трояны можно рассматривать как дополнительную вредоносную нагрузку, но не как самостоятельную программу. Нередко пользователи сами загружают троянские программы из Интернет.

Жизненный цикл троянов состоит всего из трех стадий:

- *Проникновение в систему.*
- *Активация.*
- *Выполнение вредоносных действий.*

Как уже говорилось выше, проникать в систему трояны могут двумя путями - самостоятельно и в кооперации с вирусом или сетевым червем. В первом случае обычно используется маскировка, когда троян выдает себя за полезное приложение, которое пользователь самостоятельно копирует себе на диск (например, загружает из Интернет) и запускает. При этом программа действительно может быть полезна, однако наряду с основными функциями она может выполнять действия, свойственные трояну.

Пример. Trojan.SymbOS.Hobble.a является архивом для операционной системы Symbian (SIS-архивом). При этом он маскируется под антивирус Symantec и носит имя symantec.sis. После запуска на смартфоне троян подменяет оригинальный файл оболочки FExplorer.app на поврежденный

файл. В результате при следующей загрузке операционной системы большинство функций смартфона оказываются недоступными.

Для проникновения на компьютер, трояну необходима активация и здесь он похож на червя - либо требует активных действий от пользователя или же через уязвимости в программном обеспечении самостоятельно заражает систему.

Поскольку главная цель написания троянов - это производство несанкционированных действий, они классифицируются по типу вредоносной нагрузки.

1. Клавиатурные шпионы, постоянно находясь в оперативной памяти, записывают все данные, поступающие от клавиатуры с целью последующей их передачи своему автору.

2. Похитители паролей предназначены для кражи паролей путем поиска на зараженном компьютере специальных файлов, которые их содержат.

Пример. Trojan-PSW.Win32.LdPinch.kw собирает сведения о системе, а также логины и пароли для различных сервисов и прикладных программ - мессенджеров, почтовых клиентов, программ дозвона. Часто эти данные оказываются слабо защищены, что позволяет трояну их получить и отправить злоумышленнику по электронной почте.

3. Утилиты скрытого удаленного управления *- это трояны, которые обеспечивают несанкционированный удаленный контроль над инфицированным компьютером.* Перечень действий, которые позволяет выполнять тот или иной троян, определяется его функциональностью, заложенной автором. Обычно это возможность скрыто загружать, отсылать, запускать или уничтожать файлы. Такие трояны могут быть использованы как для получения конфиденциальной информации, так и для запуска вирусов, уничтожения данных.

Пример. Backdoor.Win32.Netbus.170 предоставляет полный контроль над компьютером пользователя, включая выполнение любых файловых операций, загрузку и запуск других программ, получение снимков экрана и т. д.

4. Люки (backdoor) *— трояны предоставляющие злоумышленнику ограниченный контроль над компьютером пользователя. От утилит удаленного управления отличаются более простым устройством и, как следствие, небольшим количеством доступных действий.* Тем не менее, обычно одними из действий являются возможность загрузки и запуска любых файлов по команде злоумышленника, что позволяет при необходимости превратить ограниченный контроль в полный.

Пример. Троян Backdoor.win32.Wootbot.gen использует IRC-канал для получения команд от «хозяина». По команде троян может загружать и запускать на выполнение другие программы, сканировать другие компьютеры на наличие уязвимостей и устанавливать себя на компьютеры через обнаруженные уязвимости.

5. Анонимные SMTP-сервера и прокси-сервера *- разновидность троянов, которые на зараженном компьютере организовывают несанкционированную отправку электронной почты, что часто используется для рассылки спама.*

Пример. Трояны из семейства Trojan-Proxy.Win32.Mitglieder распространяются с различными версиями червей Bagle. Троян запускается червем, открывает на компьютере порт и отправляет автору вируса информацию об IP-адресе зараженного компьютера. После этого компьютер может использоваться для рассылки спама.

6. Утилиты дозвона *- в скрытом от пользователя режиме инициируют подключение к платным сервисам Интернет.*

7. Модификаторы настроек браузера меняют стартовую страницу в браузере, страницу поиска или еще какие-либо настройки, открывают дополнительные окна, имитируют нажатия на рекламные баннеры и т. п.

8. Логические бомбы *характеризуются способностью при срабатывании заложенных в них условий (в конкретный день, время суток, определенное действие пользователя или команды извне) выполнять какое-либо действие, например, удаление файлов.*

Пример. Virus.Win9x.CIH, Macro.Word97.Thus

Отдельно отметим, что существуют программы из класса троянов, которые наносят вред другим, удаленным компьютерам и сетям, при этом не нарушая работоспособности инфицированного компьютера. Яркие представители этой группы - организаторы DDoS-атак.

13.5.4 Другие вредоносные программы

Среди множества других вредоносных программ, для которых нельзя привести общий критерий, можно выделить следующие небольшие группы.

1. Условно опасные программы*, то есть такие, о которых нельзя однозначно сказать, что они вредоносны.* Такие программы обычно становятся опасными только при определенных условиях или действиях пользователя.

К условно опасным программам относятся:

- ***Riskware*** *- вполне легальные программы, которые сами по себе не опасны, но обладают функционалом, позволяющим злоумышленнику использовать их с вредоносными целями.* К riskware относятся обычные утилиты удаленного управления, которыми часто пользуются администраторы больших сетей, клиенты IRC, программы для загрузки файлов из Интернет, утилиты восстановления забытых паролей и другие.
- ***Рекламные утилиты*** *(adware) - условно-бесплатные программы, которые в качестве платы за свое использование демонстрируют пользователю рекламу, чаще всего в виде графических баннеров.* После официальной оплаты и регистрации обычно показ рекламы заканчивается и программы начинают работать в обычном режиме. Проблема adware кроется в механизмах, которые используются для загрузки рекламы на компьютер. Кроме того, что для этих целей часто используются программы сторонних и не всегда проверенных производителей, даже после регистрации такие модули могут автоматически не удаляться и продолжать свою работу в скрытом режиме.
- ***Pornware*** *- к этому классу относятся утилиты, так или иначе связанные с показом пользователям информации порнографического характера.* На сегодняшний день это программы, которые самостоятельно дозваниваются до порнографических телефонных служб, загружают из Интернет порнографические материалы или утилиты, предлагающие услуги по поиску и показу такой информации. Отметим, что к вредоносным программам относятся только те утилиты класса pornware, которые устанавливаются на компьютер пользователя несанкционированно - через уязвимость в операционной системы или браузера или при помощи троянов. Обычно это делается с целью насильственного показа рекламы платных порнографических сайтов или служб.

2. Хакерские утилиты *- К этому виду программ относятся программы скрытия кода зараженных файлов от антивирусной проверки (шифровальщики файлов), автоматизации создания сетевых червей, компьютерных вирусов и троянских программ (конструкторы вирусов), наборы программ, которые использую*т хакеры для скрытного взятия под контроль взломанной системы (RootKit) и другие подобные утилиты. То есть такие специфические программы, которые обычно используют только хакеры.

3. Злые шутки *- программы, которые намеренно вводят пользователя в заблуждение путем показа уведомлений о, например, форматировании диска или обнаружении вирусов, хотя на самом деле ничего не происходит.* Текст таких сообщений целиком и полностью отражает фантазию автора.

13.6 Примеры угроз безопасности информации реализуемых вредоностными программами

Рассмотрим угрозы безопасности информации с точки зрения вирусов. Учитывая тот факт, что общее число вирусов по состоянию на сегодня превосходит 100000, проанализировать угрозы со стороны каждого из них является слишком трудоемкой и бесполезной задачей, поскольку ежедневно возрастает количество вирусов, а значит, необходимо ежедневно модифицировать полученный список. В этой работе мы будем считать, что вирус способен реализовать любую из угроз безопасности информации.

Существует множество способов классификации угроз безопасности информации, которая обрабатывается в автоматизированной системе. Наиболее часто используется классификация угроз по результату их влияния на информацию, а именно - нарушение конфиденциальности, целостности и доступности.

Для каждой угрозы существует несколько способов ее реализации со стороны вирусов.

Угроза нарушения конфиденциальности.

- Кража информации и ее распространение с помощью штатных средств связи либо скрытых каналов передачи: Email-Worm.Win32.Sircam - рассылал вместе с вирусными копиями произвольные документы, найденные на зараженном компьютере.
- Кража паролей доступа, ключей шифрования и пр.: любые трояны, крадущие пароли, Trojan-PSW.Win32.LdPinch.gen.
- Удаленное управление: Backdoor.Win32.NetBus, Email-Worm.Win32.Bagle (backdoor-функциональность).

Угроза нарушения целостности.

- Модификация без уничтожения (изменение информации): любой паразитирующий вирус.
- Модификация посредством уничтожения либо шифрации (удаление некоторых типов документов): Virus.DOS.OneHalf - шифрование содержимого диска, Virus.Win32.Gpcode.f - шифрует файлы с определенными расширениями, после чего самоуничтожается, оставляя рядом с зашифрованными файлами координаты для связи по вопросам расшифровки файлов.
- Модификация путем низкоуровневого уничтожения носителя (форматирование носителя, уничтожение таблиц распределения файлов): Virus.MSWord.Melissa.w - 25 декабря форматирует диск C:

Угроза нарушения доступности.

- Загрузка каналов передачи данных большим числом пакетов: Net-Worm.Win32.Slammer - непрерывная рассылка инфицированных пакетов в бесконечном цикле.
- Любая деятельность, результатом которой является невозможность доступа к информации; различные звуковые и визуальные эффекты: Email-Worm.Win32.Bagle.p - блокирование доступа к сайтам антивирусных компаний.
- Вывод компьютера из строя путем уничтожения либо порчи критических составляющих (уничтожение Flash BIOS): Virus.Win9x.CIH - порча Flash BIOS.

Как несложно было убедиться, для каждого из приведенных выше способов реализации угроз можно привести конкретный пример вируса, реализующего один или одновременно несколько способов.

13.7 История компьютерных вирусов

Теоретические основы создания компьютерных вирусов были заложены в 40-х годах XX столетия американским ученым Джоном фон Нейманом (John von Neumann), который также известен как автор базовых принципов работы современного компьютера. Впервые же термин вирус в отношении компьютерных программ применил Фред Коэн (Fred Cohen). Это случилось 3 ноября 1983 года на еженедельном семинаре по компьютерной безопасности в Университете Южной Калифорнии (США), где был предложен проект по созданию самораспространяющейся программы, которую тут же окрестили вирусом. Для ее отладки потребовалось 8 часов компьютерного времени на машине VAX 11/750 под управлением операционной системы Unix и ровно через неделю, 10 ноября состоялась первая демонстрация. Фредом Коэном по результатам этих исследований была опубликована работа «Computer Viruses: theory and experiments» с подробным описанием проблемы.

Поскольку рассматриваемые вирусы - это по сути компьютерные программы, то об их истории можно говорить только начиная с появления компьютеров, то есть с 1946 года, когда в США была выпущена первая электронно-вычислительная машина (ЭВМ) - ENIAC (Electronic Numerical Integrator And Computer). Однако до появления в 1960 году коммерческих компьютеров, доступ к ЭВМ был сильно ограничен и вирусных инцидентов зафиксировано не было.

Первый известный вирус был написан для компьютера Univac 1108 (конец 1960-х - начало 1970-х годов). Он назывался Pervading Animal и фактически представлял собой игру, написанную с ошибкой - с помощью наводящих вопросов программа пыталась определить имя животного, задуманного играющим. Ошибка заключалась в том, что при добавлении новых вопросов модифицированная игра записывалась поверх старой версии

плюс копировалась в другие директории. Следовательно через некоторое время диск становился переполненным. Поскольку Pervading Animal не был настоящим вирусом, он не содержал процедуры самораспространения и передавался исключительно через пользователей, желающих по собственной воле переписать программу.

В 1969 году в США была создана первая глобальная компьютерная сеть, прародитель современной Интернет, ARPANET (Advanced Research Projects Agency Network). Она объединяла четыре ведущие научные центра США и служила для быстрого обмена научной информацией. Не удивительно, что уже в начале 1970-х в ARPANET появился первый вирус, умеющий распространяться по сети. Он назывался Creeper и был способен самостоятельно выйти в сеть через модем и сохранить свою копию на удаленной машине. На зараженных компьютерах вирус обнаруживал себя сообщением «I'M THE CREEPER: CATCH ME IF YOU CAN». Для удаления назойливого, но в целом безобидного вируса неизвестным была создана программа Reaper. По сути это был вирус, выполнявший некоторые функции, свойственные антивирусу: он распространялся по компьютерной сети и в случае обнаружения на машине вируса Creeper, уничтожал его.

В это время компьютеры использовались исключительно в промышленных целях - они были очень дороги и сложны в эксплуатации, время работы на них было расписано по минутам. Выпуск персональных компьютеров, то есть таких, которые могли быть приобретены отдельными людьми и использованы в личных целях, был налажен в конце 70-х - начале 80-х годов прошлого века. Это были персональные компьютеры Apple и IBM Personal Computer. Однако с развитием компьютерной техники прогрессировали и компьютерные вирусы. В 1981 году были зафиксированы случаи заражения Elk Cloner, который распространялся через пиратские копии компьютерных игр. Поскольку жестких дисков тогда еще не было, он записывался в загрузочные сектора дискет и проявлял себя переворачиванием изображения на экране.

В 1984 году вышли в свет первые антивирусные программы - CHK4BOMB и BOMBSQAD. Их автором был Энди Хопкинс (Andy Hopkins). Программы анализировали загрузочные модули и позволяли перехватывать запись и форматирование, выполняемые через BIOS. На то время они были очень эффективны и быстро завоевали популярность.

Первую настоящую глобальную эпидемию вызвал в 1986 году вирус Brain. Он был написан двумя братьями-программистами Баситом Фарук и Амжадом Алви (Basit Farooq Alvi и Amjad Alvi) из Пакистана с целью определения уровня компьютерного пиратства у себя в стране: вирус заражал загрузочные сектора, менял метку диска на «(c) Brain» и оставлял сообщение с именами, адресом и телефоном авторов. Отличительная черта Brain - умение подставлять незараженный оригинал вместо реальных данных при попытке просмотра пользователем инфицированного загрузочного сектора (так называемая стелс-технология). В течение нескольких месяцев программа

вышла за пределы Пакистана и к лету 1987 года эпидемия достигла глобальных масштабов. Ничего деструктивного вирус не делал.

В этом же году произошло еще одно знаменательное событие. Немецкий программист Ральф Бюргер (Ralf Burger) открыл возможность создания программой своих копий путем добавления своего кода к выполняемым DOS-файлам формата COM. Опытный образец программы, получившей название Virdem, был продемонстрирован на форуме компьютерного андеграунда - Chaos Computer Club (декабрь 1986 года, Гамбург, ФРГ). По результатами исследований Бюргер выпустил книгу «Computer Viruses. The Disease of High Technologies», послужившую толчком к написанию тысяч компьютерных вирусов, частично или полностью использовавших описанные автором идеи.

В следующем 1987 году был написан первый по-настоящему вредоносный вирус - Lehigh. Он вызвал эпидемию в Лехайском университете (США). Lehigh заражал только системные файлы COMMAND.COM и был запрограммирован на удаление всей информации на инфицированном диске. В течение нескольких дней было уничтожено содержимое сотен дискет из библиотеки университета и личных дискет студентов. Всего за время эпидемии было заражено около четырех тысяч компьютеров. Однако за пределы университета Lehigh не вышел.

Mike RoChenle - псевдоним автора первой известной вирусной мистификации. В октябре 1988 года он разослал на станции BBS большое количество сообщений о вирусе, который передается от модема к модему со скоростью 2400 бит/с. В качестве панацеи предлагалось перейти на использование модемов со скоростью 1200 бит/с. Как это ни смешно, многие пользователи действительно последовали этому совету.

В ноябре 1988 года случилась глобальная эпидемия червя Морриса. Небольшая программа, написанная 23-летним студентом Корнельского университета (США) Робертом Моррисом, использовала ошибки в системе безопасности операционной системы Unix для платформ VAX и Sun Microsystems. С целью незаметного проникновения в вычислительные системы, связанные с сетью ARPANET, использовался подбор паролей (из списка, содержащего 481 вариант). Это позволяло маскироваться под задачу легальных пользователей системы. Однако из-за ошибок в коде безвредная по замыслу программа неограниченно рассылала свои копии по другим компьютерам сети, запускала их на выполнение и таким образом забирала под себя все сетевые ресурсы. Червь Морриса заразил по разным оценкам от 6000 до 9000 компьютеров в США (включая Исследовательский центр NASA) и практически парализовал их работу на срок до пяти суток. Общие убытки были оценены в минимум 8 миллионов часов потери доступа и свыше миллиона часов прямых потерь на восстановление работоспособности систем. Общая стоимость этих затрат оценивается в 96 миллионов долларов. Ущерб был бы гораздо больше, если бы червь изначально создавался с разрушительными целями. Роберт Моррис также стал первым человеком,

осужденным за написание и распространение компьютерных вирусов - 4 мая 1990 года состоялся суд, который приговорил его к 3 годам условно, 400 часам общественных работ и штрафу в 10 тысяч долларов США.

Примечательно, что в том же году, когда случилась эпидемия червя Морриса, известный программист Питер Нортон (Peter Norton) высказался резко против существования вирусов. Он официально объявил их несуществующим мифом и сравнил со сказками о крокодилах, живущих в канализации Нью-Йорка. Это показывает сколь низка была культура антивирусной безопасности в то время.

Тогда же, в 1988, вышла первая широко известная антивирусная программа английским программистом Аланом Соломоном (Alan Solomon) и называлась Dr. Solomon's Anti-Virus Toolkit. Она завоевала огромную популярность и просуществовала вплоть до 1998 года, когда компания Dr. Solomon была поглощена другим производителем антивирусов - американской Network Associates (NAI).

В декабре 1989 года разразилась первая эпидемия троянской программы - Aids Information Diskette. Ее автор разослал около 20000 дискет с вирусом по почтовым адресам в Европе, Африке и Австралии, похищенным из баз данных Организации всемирного здравоохранения и журнала PC Business World. После запуска вредоносная программа автоматически внедрялась в систему, создавала свои собственные скрытые файлы и директории и модифицировала системные файлы. Через 90 загрузок операционной системы все файлы на диске становились недоступными, кроме одного - с сообщением, предлагавшим прислать $189 на указанный адрес. Автор троянца, Джозеф Попп (Joseph Popp), признанный позднее невменяемым, был задержан в момент обналичивания чека и осужден за вымогательство. Фактически, Aids Information Diskette - это первый и единственный вирус, для массовой рассылки, использовавший настоящую почту.

В том же году был обнаружен вирус Cascade, вызывающий характерный видеоэффект - осыпание букв на экране. Примечателен тем, что послужил толчком для профессиональной переориентации Евгения Касперского на создание программ-антивирусов, будучи обнаруженным на его рабочем компьютере. Уже через месяц второй инцидент (вирус Vacsina) был закрыт при помощи первой версии антивируса -V, который несколькими годами позже был переименован в AVP - AntiViral Toolkit Pro.

Вскоре после этого, в конце 1990, несмотря на громкое заявление Питера Нортона, прозвучавшее двумя годами ранее и где он авторитетно заявлял о надуманности проблемы вирусов, вышла первая версия антивирусной программы Norton AntiVirus.

Первый общедоступный конструктор вирусов VCL (Virus Creation Laboratory), представляющий собой графическую среду для разработки вирусов для операционной системы MS DOS, появился в июле 1992 года. Начиная с этого момента, любой человек мог легко сформировать и написать

вирус. Этот год также положил начало эпохи вирусов для Windows - был создан первый вирус, поражающий исполняемые файлы Microsoft Windows 3.1. Однако поскольку Win.Vir эпидемии не вызвал, его появление осталось практически незаметным.

OneHalf, очень сложный вирус, обнаруженный в июне 1994 года, вызвал глобальную эпидемию во всем мире, в том числе в России. Он заражал загрузочные сектора дисков и COM/EXE-файлы, увеличивая их размер на 3544, 3577 или 3518 байта, в зависимости от модификации. При каждой перезагрузке зараженного компьютера зашифровывались два последних незашифрованных ранее цилиндра жесткого диска. Это продолжалось до тех пор, пока весь винчестер не оказывался зашифрованным. Встроенная стелс-процедура позволяла вирусу при запросе зашифрованной информации производить расшифровку на лету - следовательно, пользователь долгое время пребывал в неведении. Единственным визуальным проявлением вируса было сообщение «Dis is one half. Press any key to continue...», выводившееся в момент достижения количеством зашифрованных цилиндров диска половины от их общего числа. Однако при первой же попытке лечения, после вылечивания загрузочных секторов диска, вся информация на винчестере становилась недоступной, без возможности восстановления. Популярности этого вируса в России поспособствовала компания Доктор Веб, которая выпустила новую версию своего антивируса, анонсировав его как средство от OneHalf. Однако на практике после лечения загрузочных секторов от этого вируса, Dr.Web забывал расшифровать информацию на диске и восстановить ее было уже невозможно.

Следующий год запомнился инцидентом в корпорации Microsoft. В феврале 1995 года, в преддверии выпуска новой операционной системы Windows 95, была разослана демонстрационная дискета, зараженная загрузочным вирусом Form. Копии этого диска получили 160 бета-тестеров, один из которых не поленился провести антивирусную проверку. Вслед за Microsoft отличились журналы PC Magazine (английская редакция) и Computer Life, которые разослали своим подписчикам дискеты, зараженные загрузочными вирусами Sampo и Parity_Boot соответственно.

В августе 1995 появился Concept - первый вирус, поражавший документы Microsoft Word.

В том же 1995 году, в бесплатном приложении полуподпольного издания известного специалиста в области компьютерных вирусов Марка Людвига (Mark Ludwig) «Underground Technology Review», был приведен исходный код вируса Green Stripe, который заражал документы AmiPro, популярного в то время текстового редактора.

До февраля 1997 года считалось, что операционная систем Linux неуязвима перед вирусами, пока не появился Linux.Bliss. В марте того же года были зафиксированы первые случаи использования возможностей

электронной почты - ShareFun, по совместительству первый макро-вирус для MS Word 6/7, распространялся с помощью почтовой программы MS Mail.

Первая утилита удаленного администрирования - BackOrifice, Backdoor.BO - была обнаружена в августе 1998 года. Единственное ее отличие от обычных программ для удаленного управления - это несанкционированная установка и запуск. Действие утилиты сводилось к скрытому слежению за системой: ссылка на BackOrifice отсутствовала в списке активных приложений, но при этом зараженный компьютер был открыт для удаленного доступа. Фактически, на зараженные компьютеры предоставлялся свободный вход для других вредоносных программ. Впоследствии возник целый класс вирусов - червей, размножение которых базировалось на оставленных BackOrifice дырах.

26 марта 1999 года началась глобальная эпидемия Melissa - первого вируса для MS Word, сочетавшего в себе также и функциональность интернет-червя. Сразу же после заражения системы он считывал адресную книгу почтовой программы MS Outlook и рассылал по первым 50 найденным адресам свои копии. Причем это делалось абсолютно незаметно для пользователя и, что самое страшное, от его имени. Такие компании как Microsoft, Intel, Lockheed Martin были вынуждены временно отключить свои корпоративные службы электронной почты. По разным оценкам совокупный ущерб от вируса варьировался от нескольких миллионов до десятков миллионов долларов США.

Через некоторое время был обнаружен и арестован автор вируса Melissa, Дэвид Л. Смит (David L. Smith). 9 декабря он был признан виновным и осужден на 10 лет тюремного заключения и к штрафу в размере 400 000 долларов США.

В декабре 1999 года был впервые обнаружен вирус-червь с заложенными в нем функциями удаленного самообновления - Babylonia. Он ежеминутно пытался соединиться с сервером, находящемся в Японии и загрузить оттуда список вирусных модулей.

LoveLetter - это скрипт-вирус, 5 мая 2000 года побивший рекорд вируса Melissa по скорости распространения. Всего в течение нескольких часов были поражены миллионы компьютеров - LoveLetter попал в Книгу Рекордов Гиннеса. Успех гарантировали методы социальной инженерии: электронное сообщение имело тему «I love you» и интригующий текст, призывающий открыть вложенный файл с вирусом.

Август 2000 года ознаменовался завоеванием вирусами мобильных устройств - вирус Liberty заражал карманные компьютеры Palm Pilot с операционной системой PalmOS.

На тот момент все известные вирусы для хранения собственных копий использовали файлы, то есть ПЗУ компьютера. Обнаруженный 12 июля 2001 года CodeRed стал первым представителем нового типа вредоносных программ, способных активно распространяться и работать на зараженных компьютерах без использования файлов. В процессе работы такие вирусы

существуют исключительно в системной памяти, а при передаче на другие компьютеры - в виде специальных пакетов данных. Для проникновения на удаленные компьютеры CodeRed использовал брешь в системе безопасности IIS (Internet Information Services), которая позволяет злоумышленникам запускать на удаленных серверах посторонний программный код. 18 июня 2001 года Microsoft выпустила соответствующую заплатку, однако подавляющее большинство пользователей не успело вовремя обновить свое программное обеспечение. CodeRed вызвал эпидемию, заразив около 12000 (по другим данным - до 200000) серверов по всему миру и провел крупномасштабную DDoS атаку на веб-сервер Белого дома, вызвав нарушение его нормальной работы. Через неделю, 19 июля появилась новая модификация CodeRed, показавшая чудеса распространения - более 350000 машин за 14 часов (до 2000 компьютеров в минуту).

В это же время был обнаружен почтовый червь Sircam (12 июля 2001 года). Этот вирус отличала необычная процедура выбора имени зараженного вложения. Для этого случайным образом на диске инфицированного компьютера выбирался документ, к имени которого добавлялось расширение .pif, .lnk, .bat или .com. Полученная конструкция вида mydiary.doc.com служила темой рассылаемых писем и именем новой копии программы: к отобранному файлу дописывался код червя - таким образом Sircam мог привести к утечке конфиденциальной информации. При рассылке в поле от указывался один из адресов, найденных на зараженном компьютере, а сообщение содержало текст вида «Hi! How are you? I send you this file in order to have your advice. See you later. Thanks». Кроме этого, в определенный момент времени (в зависимости от системного времени и модификации вируса) на зараженном компьютере удалялись все файлы на системном жестком диске.

18 сентября 2001 года началась эпидемия Nimda - этот вирус-червь в течение всего 12 часов поразил до 450000 компьютеров. Для распространения были задействованы пять методов: электронная почта (брешь в системе безопасности Internet Explorer, позволяющая автоматически выполнять вложенный исполняемый файл), по локальной сети, внедрение на IIS-сервера, заражение браузеров, а также с помощью бекдор-процедур, оставленных предыдущими вирусами. После заражения Nimda открывал локальные диски на полный доступ для всех желающих.

Вскоре после Nimda появился Klez - почтовый червь, различные модификации которого на протяжении следующих нескольких лет занимали первые строки в рейтингах популярности. Программа проникала на компьютер по сети или через электронную почту, используя брешь в защите IFrame браузера Internet Explorer, которая допускала автоматический запуск вложенного файла. Также вирус имел встроенную функцию поиска и подавления антивирусного программного обеспечения. Klez дописывал свой код к одному из документов на зараженной машине и начинал массовую рассылку. В поле «От» подставлялся любой адрес, найденный на компьютере

или же случайно сгенерированный. При этом список всех обнаруженных на зараженном компьютере адресов электронной почты также присоединялся к вложению. Кроме рассылки своих копий, червь обнаруживал себя по 13-м числам четных месяцев или шестым нечетных, в зависимости от модификации: в такой день все файлы на зараженных компьютерах заполнялись случайным содержимым.

Стоит также отметить Tanatos/Bugbear (впервые обнаружен в октябре 2001 года) - почтовый червь, устанавливающий бекдор-процедуру (Backdoor) и троян - клавиатурный шпион. Процедура распространения практически полностью была списана с Klez - копирование по сети, массовая рассылка с зараженным документом во вложении, использование уязвимости IFrame в Internet Explorer, подавление антивирусных программ. Кроме увеличения трафика, вирус проявлял себя спонтанной печатью разнообразного мусора на сетевых принтерах.

В январе 2003 года грянула эпидемия интернет-червя Slammer, заражающего сервера под управлением Microsoft SQL Server 2000. Вирус использовал брешь в системе безопасности SQL Server, заплата к которой вышла шестью месяцами ранее. После проникновения на компьютер Slammer начинал в бесконечном цикле посылать свой код на случайно выбранные адреса в сети - только за первые 10 минут было поражено около 90% (120 000 единиц) всех уязвимых серверов, при этом пять из тринадцати главных DNS-серверов сети Интернет вышли из строя. Slammer имел крайне небольшой размер - всего 376 байт (CodeRed - 4 КБ, Nimda - 60 КБ) и присутствовал только в памяти зараженных компьютеров. Более того, при работе червя никакие файлы не создавались, и червь никак не проявлял себя (помимо сетевой активности зараженного компьютера). Это означает, что лечение заключалось только в перезагрузке сервера, а антивирусы в данной ситуации бессильны.

В августе 2003 года около 8 миллионов компьютеров во всем мире оказались заражены интернет-червем Lovesan/Blaster. Для размножения использовалась очередная брешь - на этот раз в службе DCOM RPC Microsoft Windows. Кроме того, Lovesan/Blaster включал в себя функцию DDoS-атаки на сервер с обновлениями для Windows.

Неделей позже новый вирус, Sobig.f, установил новый рекорд по скорости - доля зараженных им писем доходила до 10 % от всей корреспонденции. Это достигалось использованием спамерских технологий. Sobig.f также инициировал цепную реакцию: каждый новый вариант червя создавал сеть инфицированных компьютеров, которая позднее использовалась в качестве платформы для новой эпидемии. Однако конец эпидемии запрограммировал сам автор - 10 сентября 2003 года Sobig.f прекратил размножение.

В феврале 2004 года появился Bizex (также известный как Exploit) - первый ICQ-червь. Для распространения использовалась массовая несанкционированная рассылка по ICQ сообщения

«http://www.jokeworld.biz/index.html :)) LOL». Получив от знакомого человека такую ссылку, ничего не подозревающая жертва открывала указанную страницу и в случае, если использовался браузер Internet Explorer с незакрытой уязвимостью, на компьютер загружались файлы вируса. После установки в систему, Bizex закрывал запущенный ICQ-клиент и подключившись к серверу ICQ с данными зараженного пользователя начинал рассылку по найденным на компьютере спискам контактов. Одновременно происходила кража конфиденциальной информации - банковские данные, различные логины и пароли.

В этом же 2004 году разразилась так называемая война вирусописателей. Несколько преступных группировок, известных по вирусам Bagle, Mydoom и Netsky выпускали новые модификации своих программ буквально каждый час. Каждая новая программа несла в себе очередное послание к противостоящей группировке, изобилующее нецензурными выражениями, а Netsky даже удалял любые обнаруженные экземпляры вирусов Mydoom и Bagle.

Mydoom известен также массированной 12-дневной DDoS-атакой на веб-сайт компании SCO, начавшейся 1 февраля 2004 года. За пару часов работа сервера была полностью парализована и вернуться в нормальный режим www.sco.com смог только 5 марта. В ответ руководители SCO объявили награду в размере 250 тысяч долларов США за информацию об авторе червя.

Нельзя не упомянуть червя Sasser, который в мае 2004 года поразил более 8 млн. компьютеров, а убытки от него оцениваются в 979 млн. долларов США. Для проникновения Sasser использовал уязвимость в службе LSASS Microsoft Windows.

Вскоре после того, как начали активно использоваться смартфоны (от англ. smartphone) - устройства, сочетающие в себе функции карманного компьютера и телефона, появился и первый вирус для них - Cabir (июнь 2004 года). Он распространялся через протокол Bluetooth и заражал мобильные телефоны, работающие под управлением OS Symbian. При каждом включении инфицированного телефона вирус получал управление и начинал сканировать список активных Bluetooth-соединений, выбирал первое доступное и пытался передать туда свой основной файл caribe.sis. Ничего деструктивного Cabir не делал - только снижал стабильность работы телефона за счет постоянных попыток сканирования активных Bluetooth-устройств.

Вредоносные программы - это не только вирусы, черви и трояны. К этому классу в полной мере можно отнести и adware - программы, которые отображают на экране рекламу без ведома и согласия пользователя, и pornware - программы, самостоятельно инициирующие соединения с платными порнографическими сайтами. Начиная с 2004 отмечается широкое распространение использования вирусных технологий для установки adware/pornware на целевые компьютеры. Этот год также запомнился

масштабными арестами вирусописателей - было осуждено около 100 хакеров, причем трое из них находились в двадцатке самых разыскиваемых ФБР преступников.

В следующем, 2005 году глобальных эпидемий зафиксировано не было. Это не означает уменьшение числа вирусов - наоборот, с каждым днем их появляется все больше. Но при этом можно отметить увеличение избирательности вредоносных программ - становятся популярны черви, главной целью которых является похищение определенной информации. Кроме уже ставших привычными краж номеров кредитных карт, участились случаи воровства персональных данных игроков различных онлайновых. Развитие получили и вирусные технологии для мобильных устройств. В качестве пути проникновения используются не только Bluetooth-устройства, но и обычные MMS-сообщения (червь ComWar).

В современном Интернет в среднем каждое тридцатое письмо заражено почтовым червем, около 70% всей корреспонденции - нежелательна. С ростом сети Интернет увеличивается количество потенциальных жертв вирусописателей, выход новых операционных систем влечет за собой расширение спектра возможных путей проникновения в систему и вариантов возможной вредоносной нагрузки для вирусов.

13.8 Ответственность за написание и распространение вредоносных программ

Написание и распространение вирусов - уголовно наказуемые действия. *Как и для других преступлений, меры их пресечения регулирует Уголовный Кодекс Российской Федерации. В нем* ***к вирусописателям и распространителям вирусов можно применить ряд статей из главы 28 «Преступления в сфере компьютерной информации»:***

Статья 146. Нарушение авторских и смежных прав. Незаконное использование объектов авторского права или смежных прав, а равно присвоение авторства, если эти деяния причинили крупный ущерб, - наказываются штрафом в размере от двухсот до четырехсот минимальных размеров оплаты труда или в размере заработной платы или иного дохода осужденного за период от двух до четырех месяцев, либо обязательными работами на срок от ста восьмидесяти до двухсот сорока часов, либо лишением свободы на срок до двух лет. Те же деяния, совершенные неоднократно либо группой лиц по предварительному сговору или организованной группой, - наказываются штрафом в размере от четырехсот до восьмисот минимальных размеров оплаты труда или в размере заработной платы или иного дохода осужденного за период от четырех до восьми месяцев, либо арестом на срок от четырех до шести месяцев, либо *лишением свободы на срок до пяти лет*.

Статья 138. Нарушение тайны переписки, телефонных переговоров, почтовых, телеграфных или иных сообщений. Нарушение

тайны переписки, телефонных переговоров, почтовых, телеграфных или иных сообщений граждан - наказывается штрафом в размере от пятидесяти до ста минимальных размеров оплаты труда или в размере заработной платы или иного дохода осужденного за период до одного месяца, либо обязательными работами на срок от ста двадцати до ста восьмидесяти часов, либо исправительными работами на срок до одного года. То же деяние, совершенное лицом с использованием своего служебного положения или специальных технических средств, предназначенных для негласного получения информации, - наказывается штрафом в размере от ста до трехсот минимальных размеров оплаты труда или в размере заработной платы или иного дохода осужденного за период от одного до трех месяцев, либо лишением права занимать определенные должности или заниматься определенной деятельностью на срок от двух до пяти лет, либо арестом на срок от двух до четырех месяцев. Незаконные производство, сбыт или приобретение в целях сбыта специальных технических средств, предназначенных для негласного получения информации, - наказываются штрафом в размере от двухсот до пятисот минимальных размеров оплаты труда или в размере заработной платы или иного дохода осужденного за период от двух до пяти месяцев, либо ограничением свободы на срок до трех лет, либо *лишением свободы на срок до трех лет с лишением права занимать определенные должности* или заниматься определенной деятельностью на срок до трех лет.

Статья 272. Неправомерный доступ к компьютерной информации. Неправомерный доступ к охраняемой законом компьютерной информации, то есть информации на машинном носителе, в электронно-вычислительной машине (ЭВМ), системе ЭВМ или их сети, если это деяние повлекло уничтожение, блокирование, модификацию либо копирование информации, нарушение работы ЭВМ, системы ЭВМ или их сети, - наказывается штрафом в размере от двухсот до пятисот минимальных размеров оплаты труда или в размере заработной платы или иного дохода осужденного за период от двух до пяти месяцев, либо исправительными работами на срок от шести месяцев до одного года, либо лишением свободы на срок до двух лет. То же деяние, совершенное, группой лиц по предварительному сговору или организованной группой либо лицом с использованием своего служебного положения, а равно имеющим доступ к ЭВМ, системе ЭВМ или их сети, - наказывается штрафом в размере от пятисот до восьмисот минимальных размеров оплаты труда или в размере заработной платы или иного дохода осужденного за период от пяти до восьми месяцев, либо исправительными работами на срок от одного года до двух лет, либо арестом на срок от трех до шести месяцев, либо *лишением свободы на срок до пяти лет*

Статья 273. Создание, использование и распространение вредоносных программ для ЭВМ. Создание программ для ЭВМ или внесение изменений в существующие программы, заведомо приводящих к несанкционированному уничтожению, блокированию, модификации либо

копированию информации, нарушению работы ЭВМ, системы ЭВМ или их сети, а равно использование либо распространение таких программ или машинных носителей с такими программами - наказываются лишением свободы на срок до трех лет со штрафом в размере от двухсот до пятисот минимальных размеров оплаты труда или в размере заработной платы или иного дохода осужденного за период от двух до пяти месяцев. Те же деяния, повлекшие по неосторожности тяжкие последствия, - наказываются *лишением свободы на срок от трех до семи лет*.

Статья 274. Нарушение правил эксплуатации ЭВМ, системы ЭВМ или их сети. Нарушение правил эксплуатации ЭВМ, системы ЭВМ или их сети лицом, имеющим доступ к ЭВМ, системе ЭВМ или их сети, повлекшее уничтожение, блокирование или модификацию охраняемой законом информации ЭВМ, если это деяние причинило существенный вред, - наказывается лишением права занимать определенные должности или заниматься определенной деятельностью на срок до пяти лет, либо свободы на срок до двух лет. То же деяние, повлекшее по неосторожности тяжкие последствия, - наказывается *лишением свободы на срок до четырех лет*.

14. ОСНОВЫ БОРЬБЫ С ВРЕДОНОСТНЫМИ ПРОГРАММАМИ

14.1 Самостоятельная диагностика заражения вредоносными программами

14.1.1 Признаки и диагностика заражений через браузер

Явные проявления обычно заражений через Браузер выражаются в неожиданно появляющихся рекламных сообщениях и баннерах - обычно это следствие проникновения на компьютер рекламной утилиты. Поскольку их главная цель - это привлечь внимание пользователя к рекламируемой услуге или товару, то им сложно оставаться незаметными. Также явные проявления могут вызывать ряд троянских программ, например утилиты несанкционированного дозвона к платным сервисам. Они вынуждены быть явными, поскольку используемые ими приложения сложно использовать незаметно от пользователя.

14.1.2 Подозрительные процессы

Одним из основных проявлений вредоносных программ является наличие в списке запущенных процессов (в ОС семейства Windows вызывается через CTRL+ALT+DEL, рис 14.1) подозрительных программ. Исследуя этот список и особенно сравнивая его с перечнем процессов, которые были запущены на компьютере сразу после установки системы, то есть до начала работы, можно сделать достаточно достоверные выводы об инфицировании. Это часто помогает при обнаружении вредоносных программ, имеющих лишь только скрытые или косвенные проявления.

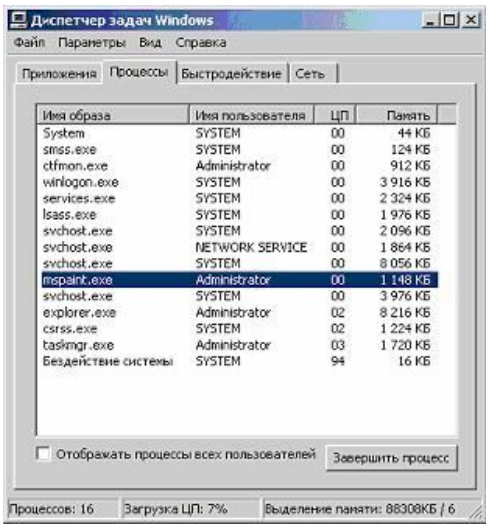


Рис. 14.1 – Просмотр запущенных в системе Windows процессов

14.1.3 Сетевая активность

Неожиданно возросшая сетевая активность может служить ярким свидетельством работы на компьютере подозрительный программы. Но при этом нужно не забывать, что ряд вполне легальных приложений также имеют свойство иногда связываться с сайтом фирмы-производителя, например для проверки наличия обновлений или более новых версий. Поэтому, прежде чем отключать сеть необходимо уметь определять какие программы и приложения вызвали эту подозрительную активность.

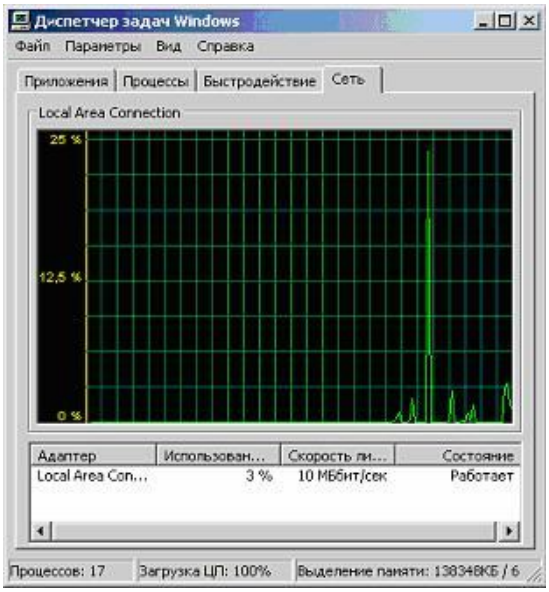


Рис. 14.2

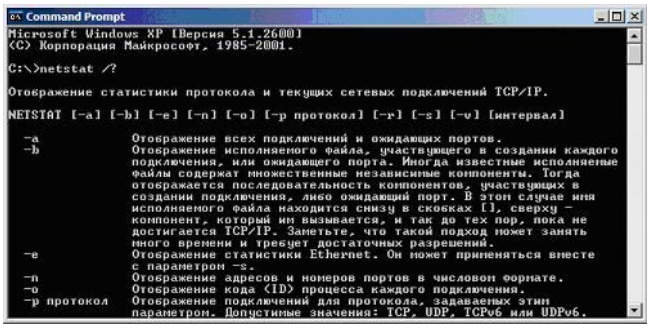


Рис. 14.3

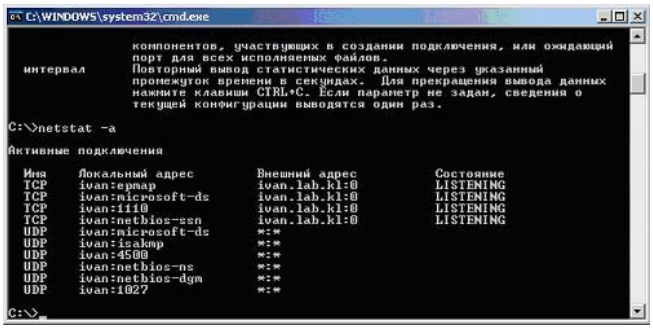


Рис. 14.4

Изучить и проанализировать сетевую активность можно с помощью встроенных в операционную систему инструментов или же воспользовавшись специальными отдельно устанавливаемыми приложениями. В этом задании это предлагается сделать с помощью **Диспетчера задач Windows** (рис. 14.2) и встроенной утилиты netstat (рис. 14.3, 14.4), которая выводит на экран мгновенную статистику сетевых соединений.

14.1.4 Элементы автозапуска

Для того, чтобы прикладная программа начала выполняться, ее нужно запустить. Следовательно, и вирус нуждается в том, чтобы его запустили.

Оптимальным с точки зрения вируса вариантом служит запуск одновременно с операционной системой - в этом случае запуск практически гарантирован.

Вредоносная программа может вносить изменения в системные файлы win.ini и system.ini.

Следует также отметить, что в файле system.ini кроме секции [boot] вредоносные программы могут использовать секцию [Drivers].

Вредоносные программы могут вносить изменения в следующие ветки реестра:

- HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion в ключи Run, RunOnce, RunOnceEx, RunServices, RunServicesOnce - для того чтобы система запускала созданные червем файлы
- HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion в ключ Run.

Кроме выше перечисленных ветвей и ключей реестра вредоносные программы могут вносить изменения и в другие ветки и ключи реестра, например:

- HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\ CurrentVersion\WOW\boot
- HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\ CurrentVersion\Drivers32
- HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\ CurrentVersion\WinLogon
- HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services
- HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Active Setup\ Installed Components
- HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\ CurrentVersion\AeDebug

Диагностика элементов автозапуска возможно путем изучения секций: SYSTEM.INI (рис. 14.5); WIN.INI (рис. 14.6); «Автозапуск» (рис. 14.7) и «Службы» (рис. 14.8), в утилите конфигурирования msconfig. Удаление запускаемого вируса из автозагрузке возможно путем использования утилиты regedit.

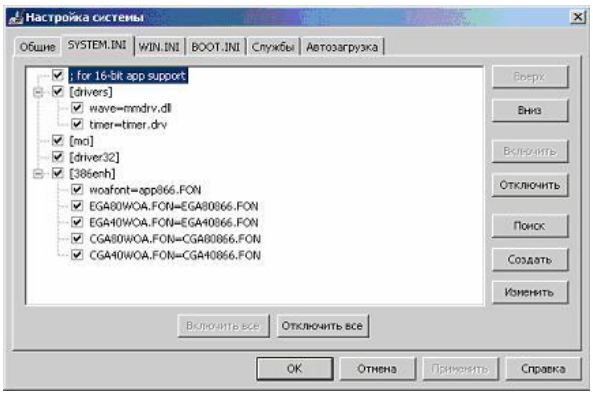


Рис. 14.5

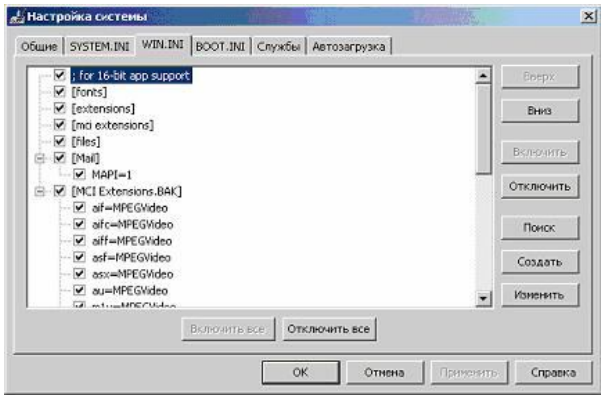


Рис. 14.6

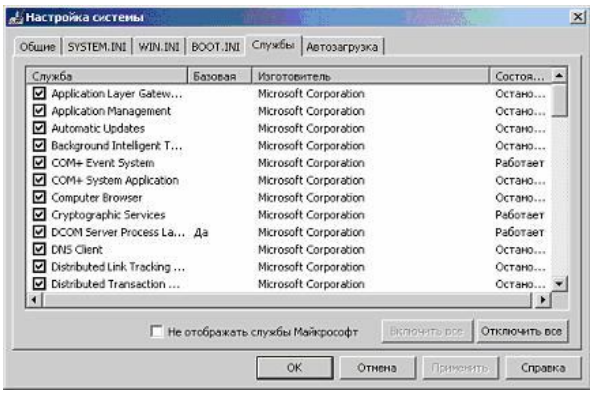


Рис. 14.7

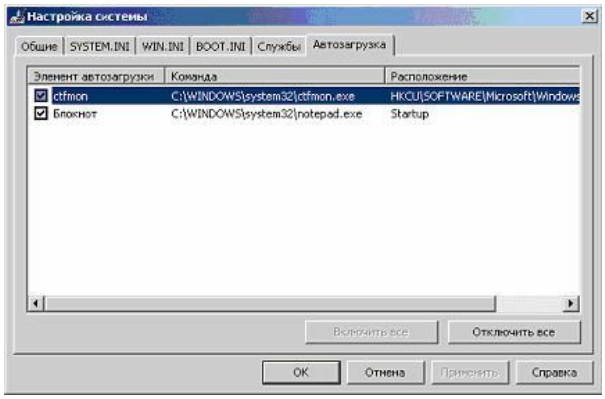


Рис. 14.8

14.2 Основы функционирования антивирусного программного обеспечения

Антивирус *- программное средство, предназначенное для борьбы с вирусами.*

Основными задачами антивируса является:

- *Препятствование проникновению вирусов в компьютерную систему.*
- *Обнаружение наличия вирусов в компьютерной системе.*
- *Устранение вирусов из компьютерной системы без нанесения повреждений другим объектам системы.*
- *Минимизация ущерба от действий вирусов.*

14.2.1 Технологии обнаружения вирусов

Технологии, применяемые в антивирусах, можно разбить на две группы

1. *Технологии сигнатурного анализа.*
2. *Технологии вероятностного анализа:*
   - *2.1. Эвристический анализ.*
   - *2.2. Поведенческий анализ.*
   - *2.3. Анализ контрольных сумм.*

Сигнатурный анализ *- метод обнаружения вирусов, заключающийся в проверке наличия в файлах сигнатур вирусов.* Сигнатурный анализ является наиболее известным методом обнаружения вирусов и используется практически во всех современных антивирусах. Для проведения проверки антивирусу необходим набор вирусных сигнатур, который хранится в антивирусной базе. Ввиду того, что сигнатурный анализ предполагает проверку файлов на наличие сигнатур вирусов, антивирусная база нуждается в периодическом обновлении для поддержания актуальности антивируса.

Недостатки сигнатурного анализа *определяют границы его функциональности - возможность обнаруживать лишь уже известные вирусы - против новых вирусов сигнатурный сканер бессилен.*

Достоинством сигнатурного анализа *является то, что наличие сигнатур вирусов предполагает возможность лечения инфицированных файлов, обнаруженных при помощи сигнатурного анализа.* Однако, лечение допустимо не для всех вирусов - трояны и большинство червей не поддаются лечению по своим конструктивным особенностям, поскольку являются цельными модулями, созданными для нанесения ущерба. Грамотная реализация вирусной сигнатуры позволяет обнаруживать известные вирусы со стопроцентной вероятностью.

Эвристический анализ *- технология, основанная на вероятностных алгоритмах, результатом работы которых является выявление подозрительных объектов. В процессе эвристического анализа проверяется структура файла, его соответствие вирусным шаблонам.* Наиболее популярной эвристической технологией является проверка содержимого файла на предмет наличия модификаций уже известных сигнатур вирусов и их комбинаций. ***Достоинством эвристического анализа*** *является то, что он может определять гибриды и новые версии ранее известных вирусов без дополнительного обновления антивирусной базы.* ***Недостатком*** *– то, что эвристический анализ не предполагает лечения.* Данная технология не способна на 100% определить вирус перед ней или нет, и как любой вероятностный алгоритм грешит ложными срабатываниями.

Поведенческий анализ *- технология, в которой решение о характере проверяемого объекта принимается на основе анализа выполняемых им операций. Поведенческий анализ весьма узко применим на практике, так как большинство действий, характерных для вирусов, могут выполняться и обычными приложениями.* Наибольшую известность получили поведенческие анализаторы скриптов и макросов, поскольку соответствующие вирусы практически всегда выполняют ряд однотипных действий. Помимо этого поведенческие анализаторы могут отслеживать попытки прямого доступа к файлам, внесение изменений в загрузочную запись дискет, форматирование жестких дисков и т. д. Поведенческие анализаторы не используют для работы дополнительных объектов, подобных вирусным базам и, как следствие, неспособны различать известные и неизвестные вирусы - все подозрительные программы априори считаются неизвестными вирусами. Аналогично, особенности работы средств, реализующих технологии поведенческого анализа, не предполагают лечения.

Например, средства защиты, вшиваемые в BIOS, также можно отнести к поведенческим анализаторам. При попытке внести изменения в MBR компьютера, анализатор блокирует действие и выводит соответствующее уведомление пользователю.

Анализ контрольных сумм *- это способ отслеживания изменений в объектах компьютерной системы. На основании анализа характера изменений - одновременность, массовость, идентичные изменения длин файлов - можно делать вывод о заражении системы.* Анализаторы контрольных сумм (также используется название «ревизоры изменений») как и поведенческие анализаторы не используют в работе дополнительные объекты и выдают вердикт о наличии вируса в системе исключительно методом экспертной оценки. Чаще подобные технологии применяются в сканерах при доступе - при первой проверке с файла снимается контрольная сумма и помещается в кэше, перед следующей проверкой того же файла сумма снимается еще раз, сравнивается, и в случае отсутствия изменений файл считается незараженным.

14.2.2 Классификация антивирусного программного обеспечения

Помимо используемых технологий, антивирусы отличаются друг от друга условиями эксплуатации. Уже из анализа задач можно сделать вывод о том, что препятствование проникновению вредоносного кода должно осуществляться непрерывно, тогда как обнаружение вредоносного кода в существующей системе - скорее разовое мероприятие. Следовательно, средства, решающие эти две задачи должны функционировать по-разному.

Таким образом, антивирусы можно разделить на две большие категории:

- ***Предназначенные для непрерывной работы*** - к этой категории относятся средства проверки при доступе, почтовые фильтры, системы сканирования проходящего трафика Интернет, другие средства, сканирующие потоки данных.
- ***Предназначенные для периодического запуска*** - различного рода средства проверки по запросу, предназначенные для однократного сканирования определенных объектов. К таким средствам можно отнести сканер по требованию файловой системы в антивирусном комплексе для рабочей станции, сканер по требованию почтовых ящиков и общих папок в антивирусном комплексе для почтовой системы (в частности, для Microsoft Exchange).

Антивирусный комплекс *- набор антивирусов, использующих одинаковое антивирусное ядро или ядра, предназначенный для решения практических проблем по обеспечению антивирусной безопасности компьютерных систем.* В антивирусный комплекс также в обязательном порядке входят средства обновления антивирусных баз.

Антивирусное ядро *- реализация механизма сигнатурного сканирования и эвристического анализа на основе имеющихся сигнатур вирусов.*

Исходя из текущей необходимости в средствах защиты выделяют следующие типы антивирусных комплексов:

1. ***Антивирусный комплекс для защиты рабочих станций*** - *предназначен для обеспечения антивирусной защиты рабочей станции*, на которой он установлен. Состоит, как и указывалось ранее из средств непрерывной работы и предназначенных для периодического запуска, а также средств обновления антивирусных баз.
2. ***Антивирусный комплекс для защиты файловых серверов*** - *предназначен для обеспечения антивирусной защиты сервера, на котором установлен.* Указание на файловый сервер в названии является скорее данью истории, корректней будет звучать термин «сетевой». Определение того, насколько нуждается в антивирусной защите сервер, осуществляется не только исходя из его назначения (является сервер файловым, почтовым, либо выполняет другую функцию), а и из используемой на нем платформы.
3. ***Антивирусный комплекс для защиты почтовых систем****, назаначение комплекса назначение - препятствовать доставке зараженных сообщений пользователям сети, но он не предназначен для защиты почтовой системы от поражения вирусами*. Как уже указывалось ранее, сегодня одним из главных средств доставки вирусов в локальную сеть является именно электронная почта. Поэтому, при наличии в локальной сети специализированного узла, обрабатывающего входящую и исходящую из сети почтовую корреспонденцию (почтового сервера), логично будет использовать средство централизованной проверки всего почтового потока на наличие вирусов)
4. ***Антивирусный комплекс для защиты шлюзов*** - *предназначен для проверки на наличие вирусов данных, через этот шлюз передаваемых. Как правило в его состав входят:*
   - *4.1.* ***Сканер HTTP-потока*** *— предназначен для проверки данных, передаваемых через шлюз по протоколу HTTP.*
   - *4.2.* ***Сканер FTP-потока*** *— предназначен для проверки данных, передаваемых через шлюз по протоколу FTP. В случае использования FTP over HTTP FTP-запросы будут проверяться сканером HTTP-потока.*
   - *4.3.* ***Сканер SMTP-потока*** *— предназначен для проверки данных, передаваемых через шлюз по SMTP.*

14.3 Комплексые средства антивирусной зашиты

14.3.1 Комплексы антивирусной зашиты для сетевых шлюзов

Задача антивируса установленного на шлюзе *— не допустить проникновения вирусов вовнутрь сети через поток данных Интернет.*

Существующие реализации универсальных антивирусов для шлюзов позволяют проверять данные, поступающие по следующим протоколам:

- *HTTP;*
- *FTP;*
- *SMTP.*

Требования к антивирусам для шлюзов:

1. ***Основные требования*** - являющиеся по сути требованием, выполнения антивирусом свою основной задачи:
   1.1. *Проверка Интернет-потоков данных (HTTP, FTP и, возможно, SMTP)* на наличие вирусов и предотвращать проникновение вирусов в сеть.
   1.2. *Проверка составных объектов* - архивов, самораспаковывающиеся архивов, упакованных исполняемых файлоа, почтовых баз, файлов почтовых форматов.
   1.3. *Возможность настраивать действия, которые будут выполняться при обнаружении вредоносных программ в потоках данных*. Стандартными действиями при этом являются - пропустить, удалить, поместить на карантин.
   1.4. *Возможность лечениять зараженных объектов*.
2. ***Требования к управлению****:*
   2.1. *Масштабируемость* — настройки одного сервера должны легко распространяться на другие сервера или группу серверов, особенно если речь идет о массивах серверов Microsoft ISA Server или подобных решениях.
   2.2. *Удаленное управление* — администратор антивирусной безопасности должен иметь возможность управлять всеми антивирусными средствами непосредственно со своего рабочего места.

При проверке протоколов Интернет, сервер антивирусной проверки наиболее оптимально устанавливать перед прокси-сервером, но за брандмауэром, если смотреть со стороны защищаемой сети (рис. 14.9). Впрочем, брандмауэр может и отсутствовать. В этом случае антивирус получает на вход тот же поток, который до этого получал прокси-сервер, выполняет проверку поступающих данных на наличие вредоносного кода и передает уже проверенные данные на прокси-сервер.

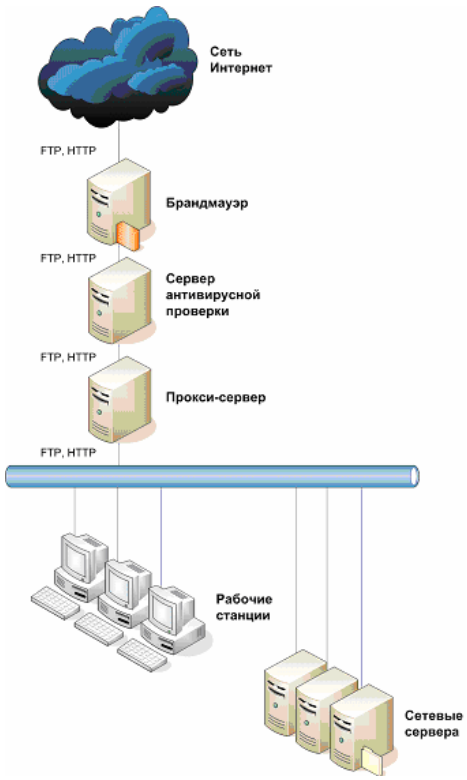


Рис. 14.9 - Установка сервера антивирусной проверки перед прокси-сервером

14.3.2 Комплексы антивирусной защиты почтовых систем

В силу того, что по разным оценкам от 80 до 95 процентов вирусов проникает в корпоративные сети через электронную почту, проверка почтового трафика является одной из важнейших задач обеспечения антивирусной безопасности организации. При этом под почтовым трафиком понимается SMTP-поток. Антивирусные комплексы для защиты почтовых систем не проверяют данные, передаваемые по протоколам IMAP и POP при обращении пользователей к своим персональным внешним по отношению к организации ящикам, поскольку это - задача антивирусного комплекса для защиты рабочих станций. Задачей антивирусного комплекса для почтовой системы является проверка и всего потока писем, поступающих и исходящих из почтовой системы организации.

К почтовому антивирусному комплексу, предьявляются требования, удовлетворение которых существенно упрощает задачу защиты организации от проникновения вирусов через почтовый поток:

1. **Карантин** — кроме удаления и доставки пользователю сообщения, возможна реализация карантинного хранилища - в этом случае пользователю доставляется уведомление со ссылкой на место в карантине, где хранится вложение к его письму.
2. **Добавление информации о проверке в письмо** — в конец проверенного письма, либо в служебный заголовок добавляется

информация о том, что письмо было проверено, а также статус проверки. Указание в таком сообщении версии использованных антивирусных баз, а также точного времени проверки позволит существенно упростить служебные расследования при поражении вирусами узлов сети.

3. **Генерация списка обнаруживаемых вирусов** — может пригодиться для точного определения состояния антивирусного комплекса во время проведения служебного расследования, при условии реализации предыдущего пункта
4. **Возможность выделения различных групп пользователей и задания различных настроек проверки для этих групп** — логичное продолжение требования к возможности исключения пользователей из проверки. Некоторые пользователи, наоборот, могут входить в группу риска, поскольку обрабатывают информацию, составляющую коммерческую либо государственную тайну. Требования к проверке корреспонденции таких пользователей должны быть более жесткими чем обычные.
5. **Возможность модификации уведомлений, в том числе и для различных групп пользователей** — может пригодиться для указания адресов и телефонов, по которым нужно обращаться с вопросами касательно антивирусной защиты.
6. **Возможность блокировки объектов, не прошедших проверку** — в некоторых случаях проверить вложение на наличие вирусов не представляется возможным — к примеру, если это часть многотомного архива либо архив, защищенный паролем. В этом случае антивирусный комплекс должен обладать возможностью блокировать сообщения, содержащие подобные вложения.
7. **Вирусная атака** — при обнаружении *N* вирусов в *M* минут иногда полезно уведомить администратора об этом факте. Подобное поведение продукта скорее всего будет свидетельствовать о вирусной эпидемии либо атаке на сервер. В обоих случаях администратор может попытаться внести изменения в настройки самого комплекса с тем, чтобы отсеивать зараженные письма на более раннем этапе, снижая, тем самым, нагрузку на сервер

Примером антивирусной защиты для почтовых систем может являться VS API 2.0 для Microsoft Exchange Server 2000. Общая схема работы VS API 2.0 приведена на рис. 14.10.

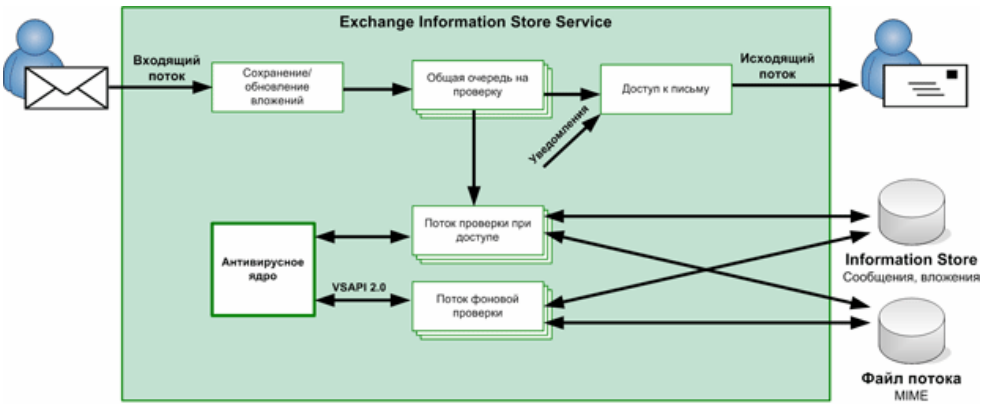


Рис. 14.10 - Схема работы VS API 2.0

14.3.4 Системы централизованного управления антивирусной защитой

Для локальной сети, больше количкство компьютеров, использование системы удаленного централизованного управления антивирусной защитой оказывается максимльно эффективным. Она позволяет администратору на своем рабочем месте обслуживать все рабочие станции и сервера сети:

1. возможность осуществлять полный контроль за вирусной активностью и состоянием антивирусной защиты в сети (основное преимущество),
2. быстро обнаруживать и оперативно устраненять все вирусные инцидентов,
3. удаленно настраивать политики антивирусной безопасности,
4. запускать проверку объектов на наличие в них вирусов,
5. включать или выключать постоянную защиту,
6. централизованно обновлять антивирусные базы,
7. разрешать или запрещать пользователям самим менять какие-либо настройки, в том числе позволять или не позволять им видеть, что на компьютере вообще установлен и работает антивирус.

Система удаленного централизованного управления обычно состоит из таких отдельных программных компонентов:

- **Клиентской антивирусной программы**, то есть антивирусного комплекса для рабочих станций или сетевых серверов.
- **Сервера администрирования** - так называется программа, которая собирает, обрабатывает и хранит все настройки, информацию обо всех событиях и инцидентах, имевших место в сети, рассылает уведомления и отчеты. Для полноценного функционирования ведется база данных для хранения всей собранной информации. Сервер администрирования и база данных могут устанавливаться как на отдельном выделенном для этого компьютере, так и на рабочем месте администратора, на одной машине или на разных.
- **Агента администрирования**, который устанавливается на все компьютеры, входящие в логическую сеть системы антивирусной

защиты. Его задача - обеспечить связь клиентской программы с сервером администрирования и оперативно передать ему информацию о состоянии антивирусной защиты на этой машине, получить новые антивирусные базы или другие указания и команды.

- **Консоли администрирования**, устанавливаемой на рабочем месте администратора. Это небольшая программа, которая позволяет в вывести данные с сервера администрирования, на их основе создать отчеты, произвести настройку клиентских компьютеров, удаленно запустить проверку или обновить антивирусные базы одновременно на нескольких машинах. Возможности той или иной консоли полностью зависят от заложенных в нее фирмой-производителем функций.

На рис. 14.11 представлена схема взаимодействия перечисленных компонентов, а на рис. 14.12 - схема сбора и передачи на хранение серверу администрирования данных о состоянии антивирусной защиты.

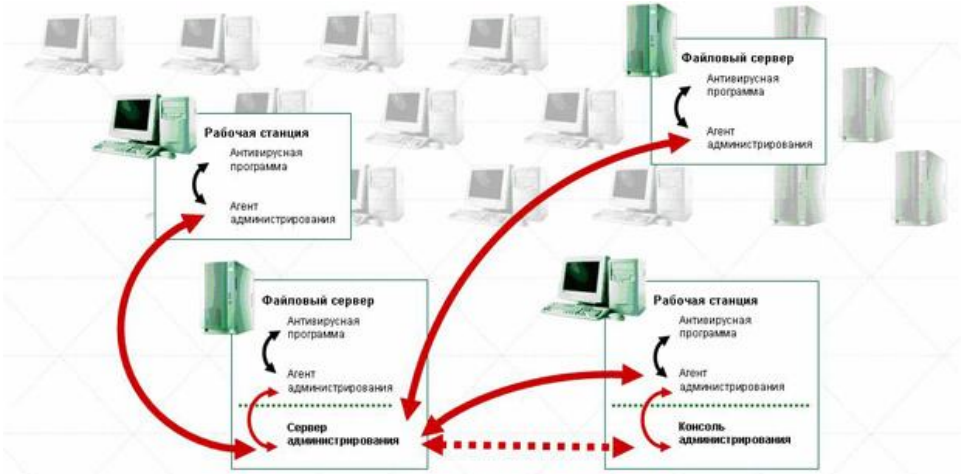


Рис. 14.11 - Схема взаимодействия компонентов централизованно управляемого комплекса антивирусной защиты в сети

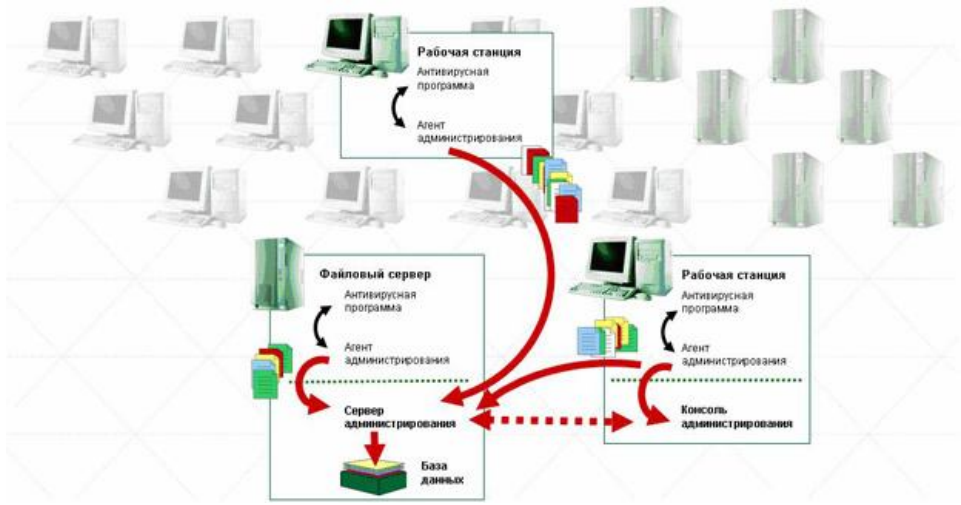


Рис. 14.12 - Схема сбора статистики в системе антивирусной защиты

ЧАСТЬ 5. ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ В ГЛОБАЛЬНЫХ СЕТЯХ

15. ТИПОВЫЕ УДАЛЕННЫЕ АТАКИ В ГЛОБАЛЬНЫХ КОМПЬЮТЕРНЫХ СЕТЯХ

15.1 Понятие типовой удаленной атаки

Рассмотрим ряд определений.

Распределенная вычислительная система (РВС) - *совокупность структурно и пространственно распределенных информационных систем, рабочих станций и вычислительных узлов обьединенных каналами связи в единую сеть для решения информационно-вычислительных задач.*

Типовая удаленная атака (УА) *- это удаленное информационное разрушающее воздействие, осуществляемое по каналам связи и характерное для любой распределенной вычислительной системы.*

Субъект атаки (или источник атаки) *- это атакующая программа или оператор, непосредственно осуществляющие воздействие.*

Хост (host) *- сетевой компьютер (рабочая станция).*

Маршрутизатор (router) *- устройство, обеспечивающее маршрутизацию пакетов обмена в глобальной сети.*

Подсеть (subnetwork) *- совокупность хостов, являющихся частью глобальной сети, для которых маршрутизатором выделен одинаковый номер подсети.* Подсеть - логическое объединение хостов маршрутизатором. Хосты внутри одной подсети могут взаимодействовать между собой непосредственно, минуя маршрутизатор.

Сегмент сети *- физическое объединение хостов.* Например, сегмент сети образуют совокупность хостов, подключенных к серверу по схеме «общая шина». При такой схеме подключения каждый хост имеет возможность подвергать анализу любой пакет в своем сегменте.

Удаленные атаки становятся возможными благодаря уязвимостям в существующих протоколах обмена данными и системах зашиты сетей, а также в ОС и ПО хостов.

Основные причины уязвимости хостов сети:

1. *открытость системы, свободный доступ к информации по организации сетевого взаимодействия, протоколам и механизмам защиты;*
2. *наличие ошибок в программном обеспечении, операционных системах и утилитах, которые открыто публикуются в сети;*
3. *разнородность используемых версий программного обеспечения и операционных систем;*
4. *сложность организации защиты межсетевого взаимодействия;*
5. *ошибки конфигурирования систем и средств защиты;*
6. *неправильное или ошибочное администрирование систем;*

7. *несвоевременное отслеживание и выполнение рекомендаций специалистов по защите и анализу случаев вторжения для ликвидации лазеек и ошибок в программном обеспечении;*
8. *«экономия» на средствах и системах обеспечения безопасности или игнорирование их;*
9. *умолчание о случаях нарушения безопасности своего хоста или сети.*

Примеры уязвимостей некоторых распространенных служб Internet:

- ***Простой протокол передачи электронной почты (Simple Mail Transfer Protocol - SMTP)*** *позволяет осуществлять почтовую транспортную службу Internet.* Одна из проблем безопасности, связанная с этим протоколом, заключается в том, что пользователь не может проверить адрес отправителя в заголовке сообщения электронной почты. В результате хакер может послать во внутреннюю сеть большое количество почтовых сообщений, что приведет к перегрузке и блокированию работы почтового сервера. Программы электронной почты использует для работы IP-адрес отправителя. Перехватывая сообщения email, хакер может употребить эту информацию для нападений, например для спуфинга (подмены адресов).
- ***Служба сетевых имен (Domain Name System - DNS)*** *представляет собой распределенную базу данных, которая преобразует имена пользователей и хост-компьютеров в IP-адреса, указываемые в заголовках пакетов, и наоборот. DNS также хранит информацию о структуре сети компании, например количестве компьютеров с IP-адресами в каждом домене.* Одной из проблем DNS является то, что эту базу данных очень трудно «скрыть» от неавторизированных пользователей. В результате DNS часто используется хакерами как источник информации об именах доверенных хост-компьютеров.
- ***Служба эмуляции удаленного терминала (TELNET)*** *употребляется для подключения к удаленным системам, присоединенным к сети; применяет базовые возможности по эмуляции терминала. При использовании этого сервиса Internet пользователи должны регистрироваться на сервере TELNET, вводя свои имя и пароль. После аутентификации пользователя его рабочая станция функционирует в режиме терминала, подключенного к внешнему хост-компьютеру.* Подключившись к серверу TELNET, хакер может сконфигурировать его программу таким образом, чтобы она записывала имена и пароли пользователей.
- ***Всемирная паутина (World Wide Web - WWW)*** *- это система, основанная на сетевых приложениях, которые позволяют пользователям просматривать содержимое различных серверов в*

Internet или интрасетях. Полезным свойством WWW является использование гипертекстовых документов, что дает пользователям возможность легко переходить от одного узла к другому. Это же свойство является и наиболее слабым местом системы WWW, поскольку ссылки на Web-узлы, хранящиеся в гипертекстовых документах, содержат информацию о том, как осуществляется доступ к соответствующим узлам. Используя эту информацию, хакеры могут разрушить Web-узел или получить доступ к хранящейся в нем конфиденциальной информации.

- *К уязвимым службам и протоколам Internet относятся также протокол копирования **UUCP**, протокол маршрутизации **RIP**, графическая оконная система **X Windows** и др.*

15.2 Классификация удаленных атак

1. ***По характеру воздействия:***
   1.1. ***Пассивное воздействие.*** Пассивным воздействие не оказывает непосредственного влияния на работу системы, но может нарушать ее политику безопасности. Именно отсутствие *непосредственного* влияния на работу распределенной ВС приводит к тому, что пассивное удаленное воздействие практически невозможно обнаружить. Примером пассивного типового удаленного воздействия в РВС служит прослушивание канала связи в сети. В отличие от активного, при пассивном воздействии не остается никаких следов (от того, что атакующий просмотрит чужое сообщение в системе, в тот же момент ничего не изменится).
   1.2. ***активное воздействие.*** Активное воздействие, оказывает непосредственное влияние на работу системы (изменение конфигурации РВС, нарушение работоспособности и т. д.) и нарушает принятую в ней политику безопасности. Практически все типы удаленных атак являются активными воздействиями. Это связано с тем, что в самой природе разрушающего воздействия содержится активное начало. Очевидной особенностью активного воздействия по сравнению с пассивным является принципиальная возможность его обнаружения (естественно, с большей или меньшей степенью сложности), так как в результате его осуществления в системе происходят определенные изменения.

2. ***По цели воздействия:***
   2.1. ***нарушение конфиденциальности информации либо ресурсов системы – перехват информации.*** Возможность перехвата информации означает получение к ней доступа, но невозможность ее модификации. Следовательно, перехват информации ведет к нарушению ее конфиденциальности. Примером перехвата

информации может служить прослушивание канала в сети. В этом случае имеется несанкционированный доступ к информации без возможности ее искажения. Очевидно также, что нарушение конфиденциальности информации является пассивным воздействием.

2.2. ***нарушение целостности информации – искажение информации.*** Возможность искажения информации означает либо полный контроль над информационным потоком между объектами системы, либо возможность передачи сообщений от имени другого объекта. Таким образом, очевидно, что искажение информации ведет к нарушению ее целостности. Данное информационное разрушающее воздействие представляет собой яркий пример активного воздействия. Примером удаленной атаки, цель которой нарушение целостности информации, может служить типовая удаленная атака (УА) «Ложный объект РВС».

2.3. ***нарушение работоспособности (доступности) системы.*** В этом случае не предполагается получение атакующим несанкционированного доступа к информации. Его основная цель - добиться, чтобы операционная система на атакуемом объекте вышла из строя и для всех остальных объектов системы доступ к ресурсам атакованного объекта был бы невозможен. Примером удаленной атаки, целью которой является нарушение работоспособности системы, может служить типовая УА «Отказ в обслуживании».

3. ***По условию начала осуществления воздействия:***

3.1. ***Атака по запросу от атакуемого объекта.*** В этом случае атакующий ожидает передачи от потенциальной цели атаки запроса определенного типа, который и будет условием начала осуществления воздействия. Примером подобных запросов в сети Internet могут служить - DNS- и ARP-запросы. Важно отметить, что данный тип удаленных атак наиболее характерен для распределенных ВС.

3.2. ***Атака по наступлению ожидаемого события на атакуемом объекте.*** В этом случае атакующий осуществляет постоянное наблюдение за состоянием операционной системы удаленной цели атаки и при возникновении определенного события в этой системе начинает воздействие. Как и в предыдущем случае, инициатором осуществления начала атаки выступает сам атакуемый объект. Примером такого события может быть прерывание сеанса работы пользователя с сервером в ОС Novell NetWare без выдачи команды LOGOUT.

3.3. ***Безусловная атака.*** В этом случае начало осуществления атаки безусловно по отношению к цели атаки, то есть атака осуществляется немедленно и безотносительно к состоянию системы и атакуемого объекта. Следовательно, в этом случае атакующий является инициатором начала осуществления атаки.

4. ***По наличию обратной связи с атакуемым объектом:***
   - ***4.1. С обратной связью.*** Удаленная атака, осуществляемая при наличии обратной связи с атакуемым объектом, характеризуется тем, что на некоторые запросы, переданные на атакуемый объект, атакующему требуется получить ответ, а, следовательно, между атакующим и целью атаки существует обратная связь, которая позволяет атакующему адекватно реагировать на все изменения, происходящие на атакуемом объекте. Подобные удаленные атаки наиболее характерны для распределенных ВС.
   - ***4.2. Без обратной связи (однонаправленная атака).*** В отличие от атак с обратной связью удаленным атакам без обратной связи не требуется реагировать на какие-либо изменения, происходящие на атакуемом объекте. Атаки данного вида обычно осуществляются передачей на атакуемый объект одиночных запросов, ответы на которые атакующему не нужны. Подобную УА можно называть однонаправленной удаленной атакой. Примером однонаправленных атак является типовая УА «Отказ в обслуживании».

5. ***По расположению субъекта атаки относительно атакуемого объекта:***
   - ***5.1. Внутрисегментное.*** В случае внутрисегментной атаки, как следует из названия, субъект и объект атаки находятся в одном сегменте. В дальнейшем будет показано, что на практике межсегментную атаку осуществить значительно труднее, чем внутрисегментную. Важно отметить, что межсегментная удаленная атака представляет гораздо большую опасность, чем внутрисегментная. Это связано с тем, что в случае межсегментной атаки объект её и непосредственно атакующий могут находиться на расстоянии многих тысяч километров друг от друга, что может существенно воспрепятствовать мерам по отражению атаки.
   - ***5.2. Межсегментное.*** При межсегментной атаке субъект и объект атаки находятся в разных сегментах.

6. ***По уровню эталонной модели ISO/OSI, на котором осуществляется воздействие:***
   - ***6.1.*** *физический,*
   - ***6.2.*** *канальный,*
   - ***6.3.*** *сетевой,*
   - ***6.4.*** *транспортный,*
   - ***6.5.*** *сеансовый,*
   - ***6.6.*** *представительный,*
   - ***6.7.*** *прикладной.*

15.3 Типовые удаленные атаки и механизмы их реализации

15.3.1 Анализ сетевого трафика

Как уже отмечалось, основной особенностью распределенной ВС (РВС) является то, что ее объекты распределены в пространстве и связь между ними физически осуществляется по сетевым соединениям и программно – т.е. сообщения и данные, пересылаемые между объектами РВС, передаются по сетевым соединениям в виде пакетов. Эта особенность привела к появлению специфичного для РВС типового удаленного воздействия, заключающегося в прослушивании канала связи. Назовем данное типовое удаленное воздействие ***анализом сетевого трафика*** (или, сокращенно, сетевым анализом).

Анализ сетевого трафика позволяет:

1. ***изучить логику работы РВС****, то есть получить взаимно однозначное соответствие событий, происходящих в системе, и команд, пересылаемых друг другу ее объектами, в момент появления этих событий.* Это достигается путем перехвата и анализа пакетов обмена на канальном уровне. Знание логики работы РВС позволяет на практике моделировать и осуществлять типовые удаленные атаки, рассмотренные в следующих пунктах на примере конкретных РВС.
2. ***перехватить поток данных, которыми обмениваются объекты РВС.*** Таким образом, удаленная атака данного типа заключается в получении на удаленном объекте несанкционированного доступа к информации, которой обмениваются два сетевых абонента. Отметим, что при этом отсутствует возможность модификации трафика и сам анализ возможен только внутри одного сегмента сети. Примером перехваченной при помощи данной типовой удаленной атаки информации могут служить имя и пароль пользователя, пересылаемые в незашифрованном виде по сети.

По характеру воздействия анализ сетевого трафика является пассивным воздействием (класс 1.1). Осуществление данной атаки без обратной связи (класс 4.2) ведет к нарушению конфиденциальности информации (класс 2.1) внутри одного сегмента сети (класс 5.1) на канальном уровне OSI (класс 6.2). При этом начало осуществления атаки безусловно по отношению к цели атаки (класс 3.3).

15.3.2 Подмена доверенного объекта или субъекта системы

Одной из проблем безопасности РВС является недостаточная идентификация и аутентификация ее удаленных друг от друга объектов. Основная трудность заключается в осуществлении однозначной

идентификации сообщений, передаваемых между субъектами и объектами взаимодействия. Обычно в РВС эта проблема решается следующим образом: в процессе создания виртуального канала объекты РВС обмениваются определенной информацией, уникально идентифицирующей данный канал. Такой обмен обычно называется «рукопожатием» (handshake). Однако, необходимо отметить, что не всегда для связи двух удаленных объектов в РВС создается виртуальный канал. Практика показывает, что зачастую, особенно для служебных сообщений (например, от маршрутизаторов) используется передача одиночных сообщений, не требующих подтверждения.

Для адресации сообщений в РВС используется сетевой адрес, который уникален для каждого объекта системы (на канальном уровне модели OSI - это аппаратный адрес сетевого адаптера, на сетевом уровне - адрес определяется в зависимости от используемого протокола сетевого уровня (например, IP-адрес). Сетевой адрес также может использоваться для идентификации объектов РВС. Однако сетевой адрес достаточно просто подделывается и поэтому использовать его в качестве единственного средства идентификации объектов недопустимо.

В том случае, когда РВС использует нестойкие алгоритмы идентификации удаленных объектов, то оказывается возможной типовая удаленная атака, заключающаяся в передаче по каналам связи сообщений от имени произвольного объекта или субъекта РВС.

При этом существуют две разновидности типовой удаленной атаки путем подмены доверенного объекта или субъекта РВС:

- ***Атака при установленном виртуальном канале.*** В случае установленного виртуального соединения атака будет заключаться в присвоении прав доверенного субъекта взаимодействия, легально подключившегося к объекту системы, что позволит атакующему вести сеанс работы с объектом распределенной системы от имени доверенного субъекта. Реализация удаленных атак данного типа обычно состоит в передаче пакетов обмена с атакующего объекта на цель атаки от имени доверенного субъекта взаимодействия (при этом переданные сообщения будут восприняты системой как корректные). Для осуществления атаки данного типа необходимо преодолеть систему идентификации и аутентификации сообщений, которая, в принципе, может использовать контрольную сумму, вычисляемую с помощью открытого ключа, динамически выработанного при установлении канала, случайные многобитные счетчики пакетов и сетевые адреса станций. Однако на практике, например, в протоколе TCP для идентификации используются два 32-битных счетчика.
- ***атака без установленного виртуального канала.*** Для служебных сообщений в РВС часто используется передача одиночных сообщений, не требующих подтверждения, то есть не требуется

создание виртуального соединения. *Атака без установленного виртуального соединения заключается в передаче служебных сообщений от имени сетевых управляющих устройств, например, от имени маршрутизаторов.* Очевидно, что в этом случае для идентификации пакетов возможно лишь использование статических ключей, определенных заранее, что довольно неудобно и требует сложной системы управления ключами. Однако, при отказе от такой системы идентификация пакетов без установленного виртуального канала будет возможна лишь по сетевому адресу отправителя, который легко подделать. Например, типовая удаленная атака, использующая навязывание ложного маршрута, путем посылки ложных управляющих сообщений основана на описанной идее.

Подмена доверенного объекта РВС является активным воздействием (класс 1.2), совершаемым с целью нарушения конфиденциальности (класс 2.1) и целостности (класс 2.2) информации, по наступлению на атакуемом объекте определенного события (класс 3.2). Данная удаленная атака может являться как внутрисегментной (класс 5.1), так и межсегментной (класс 5.2), как с обратной связью (класс 4.1), так и без обратной связи (класс 4.2) с атакуемым объектом и осуществляется на сетевом (класс 6.3) и транспортном (класс 6.4) уровнях модели OSI.

15.3.3 Внедрение ложного объекта в систему

В том случае, если в РВС недостаточно надежно решены проблемы идентификации сетевых управляющих устройств (например, маршрутизаторов), возникающие при взаимодействии последних с объектами системы, то подобная распределенная система может подвергнуться типовой удаленной атаке, связанной с изменением маршрутизации и внедрением в систему ложного объекта. В том случае, если инфраструктура сети такова, что для взаимодействия объектов необходимо использование алгоритмов удаленного поиска, то это также позволяет внедрить в систему ложный объект. Таким образом, существуют две принципиально разные причины, обуславливающие появление типовой удаленной атаки «Ложный объект РВС».

15.3.3.1 Внедрение ложного объекта путем навязывания ложного маршрута

Современные глобальные сети представляют собой совокупность сегментов сети, связанных между собой через сетевые узлы. При этом маршрутом называется последовательность узлов сети, по которой данные передаются от источника к приемнику. Каждый маршрутизатор имеет специальную таблицу, называемую таблицей маршрутизации, в которой для каждого адресата указывается оптимальный маршрут. Отметим, что таблицы

маршрутизации существуют не только у маршрутизаторов, но и у любых хостов в глобальной сети. Для обеспечения эффективной и оптимальной маршрутизации в РВС применяются специальные управляющие протоколы, позволяющие маршрутизаторам:

- обмениваться информацией друг с другом: (RIP (Routing Internet Protocol), OSPF (Open Shortest Path First)),
- уведомлять хосты о новом маршруте - ICMP (Internet Control Message Protocol),
- удаленно управлять маршрутизаторами (SNMP (Simple Network Management Protocol)).

Важно отметить, что все описанные выше протоколы позволяют удаленно изменять маршрутизацию в сети Internet, то есть являются протоколами управления сетью.

Поэтому абсолютно очевидно, что маршрутизация в глобальных сетях играет важнейшую роль и, как следствие этого, может подвергаться атаке. *Основная цель атаки, связанной с навязыванием ложного маршрута, состоит в том, чтобы изменить исходную маршрутизацию на объекте РВС так, чтобы новый маршрут проходил через ложный объект - хост атакующего. Реализация данной типовой удаленной атаки состоит в несанкционированном использовании протоколов управления сетью для изменения исходных таблиц маршрутизации.*

Данная атака проходит в две стадии:

1. Для атакующему необходимо послать Рассылка по сети определенные данными протоколами управления сетью специальные служебные сообщения от имени сетевых управляющих устройств (например, маршрутизаторов), что приводит к изменению маршрутизации в сети. В результате успешного изменения маршрута атакующий получит полный контроль над потоком информации, которой обмениваются два объекта РВС.
2. прием, анализ и передача сообщений, получаемых от дезинформированных объектов РВС.

Навязывание объекту РВС ложного маршрута - активное воздействие (класс 1.2), совершаемое с любой из целей из класса 2, безусловно по отношению к цели атаки (класс 3.3). Данная типовая удаленная атака может осуществляться как внутри одного сегмента (класс 5.1), так и межсегментно (класс 5.2), как с обратной связью (класс 4.1), так и без обратной связи с атакуемым объектом (класс 4.2) на транспортном (класс 6.3) и прикладном (класс 6.7) уровне модели OSI.

15.3.3.2 Внедрение ложного объекта путем использования недостатков алгоритмов удаленного поиска

В РВС часто оказывается, что ее удаленные объекты изначально не имеют достаточно информации, необходимой для адресации сообщений. Обычно такой информацией являются аппаратные (адрес сетевого адаптера) и логические (IP-адрес, например) адреса объектов РВС. Для получения подобной информации в распределенных ВС используются различные *алгоритмы удаленного поиска*, заключающиеся в передаче по сети специального вида поисковых запросов, и в ожидании ответов на запрос с искомой информацией. После получения ответа на запрос, запросивший субъект РВС обладает всеми необходимыми данными для адресации. Руководствуясь полученными из ответа сведениями об искомом объекте, запросивший субъект РВС начинает адресоваться к нему. Примером подобных запросов, на которых базируются алгоритмы удаленного поиска, могут служить ARP- и DNS-запросы в сети Internet.

В случае использования распределенной ВС механизмов удаленного поиска существует возможность на атакующем объекте перехватить посланный запрос и послать на него ложный ответ, где указать данные, использование которых приведет к адресации на атакующий ложный объект. В дальнейшем весь поток информации между субъектом и объектом взаимодействия будет проходить через ложный объект РВС.

Другой вариант внедрения в РВС ложного объекта использует недостатки алгоритма удаленного поиска и состоит в периодической передаче на атакуемый объект заранее подготовленного ложного ответа без приема поискового запроса. В самом деле, атакующему для того, чтобы послать ложный ответ, не всегда обязательно дожидаться приема запроса (он может, в принципе, не иметь подобной возможности перехвата запроса). При этом атакующий может спровоцировать атакуемый объект на передачу поискового запроса, и тогда его ложный ответ будет немедленно иметь успех. Данная типовая удаленная атака чрезвычайно характерна для глобальных сетей, когда у атакующего из-за нахождения его в другом сегменте относительно цели атаки просто нет возможности перехватить поисковый запрос.

Ложный объект РВС - активное воздействие (класс 1.2), совершаемое с целью нарушения конфиденциальности (класс 2.1) и целостности информации (класс 2.2), которое может являться атакой по запросу от атакуемого объекта (класс 3.1), а также безусловной атакой (класс 3.3). Данная удаленная атака является как внутрисегментной (класс 5.1), так и межсегментной (класс 5.2), имеет обратную связь с атакуемым объектом (класс 4.1) и осуществляется на канальном (класс 6.2) и прикладном (класс 6.7) уровнях модели OSI.

15.3.4 Использование ложного объекта для организации удаленной атаки на систему

Получив контроль над проходящим потоком информации между объектами, ложный объект РВС может применять различные методы воздействия на перехваченную информацию. В связи с тем, что внедрение в распределенную ВС ложного объекта является целью многих удаленных атак и представляет серьезную угрозу безопасности РВС в целом, **выделяют нижерасмотренные методы воздействия на информацию, перехваченную ложным объектом.**

15.3.4.1 Селекция потока информации и сохранение его на ложном объекте системы

Одной из атак, которую может осуществлять ложный объект РВС, является перехват передаваемой между субъектом и объектом взаимодействия информации. Важно отметить, что факт перехвата информации (файлов, например) возможен из-за того, что при выполнении некоторых операций над файлами (чтение, копирование и т. д.) содержимое этих файлов передается по сети, а, значит, поступает на ложный объект. Простейший способ реализации перехвата - это сохранение в файле всех получаемых ложным объектом пакетов обмена.

Тем не менее, данный способ перехвата информации оказывается недостаточно информативным. Это происходит вследствие того, что в пакетах обмена кроме полей данных существуют служебные поля, не представляющие в данном случае для атакующего непосредственного интереса. Следовательно, для того, чтобы получить непосредственно передаваемый файл, необходимо проводить на ложном объекте динамический семантический анализ потока информации для его селекции.

15.3.4.2 Модификация информации

Одной из особенностей любой системы воздействия, построенной по принципу ложного объекта, является то, что она способна модифицировать перехваченную информацию. Следует особо отметить, что *это один из способов, позволяющих программно модифицировать поток информации между объектами РВС с другого объекта.* Ведь для реализации перехвата информации в сети необязательно атаковать распределенную ВС по схеме «ложный объект». Эффективней будет атака, осуществляющая анализ сетевого трафика, позволяющая получать все пакеты, проходящие по каналу связи, но, в отличие от удаленной атаки по схеме «ложный объект», она не способна к модификации информации.

Рассматривают два вида модификации информации.

1. Модификация передаваемых данных. В результате селекции потока перехваченной информации и его анализа система может распознавать тип

передаваемых файлов (исполняемый или текстовый). Соответственно, в случае обнаружения текстового файла или файла данных появляется возможность модифицировать проходящие через ложный объект данные. Особую угрозу эта функция представляет для сетей обработки конфиденциальной информации.

2. Модификация передаваемого кода. Ложный объект РВС, проводя семантический анализ проходящей через него информации, может выделять из потока данных исполняемый код. Известный принцип неймановской архитектуры гласит, что не существует различий между данными и командами. Следовательно, для того, чтобы определить, что передается по сети - код или данные, необходимо использовать определенные особенности, свойственные реализации сетевого обмена в конкретной распределенной ВС или некоторые особенности, присущие конкретным типам исполняемых файлов в данной локальной ОС.

Представляется возможным выделить два различных по цели вида модификации кода:

2.1 Внедрение в РПС разрушающих программных средств - при передачи в РПС исполняемый файл модифицируется по вирусной технологии: к исполняемому файлу одним из известных способов дописывается тело РПС, а также одним из известных способов изменяется точка входа так, чтобы она указывала на начало внедренного кода РПС. Описанный способ, в принципе, ничем не отличается от стандартного заражения исполняемого файла вирусом, за исключением того, что ***файл оказался поражен вирусом или РПС в момент передачи его по сети!*** Такое возможно лишь при использовании системы воздействия, построенной по принципу «ложный объект».

2.2 Изменение логики работы исполняемого файла - происходит модификация исполняемого кода с целью изменения логики его работы. Данное воздействие требует предварительного исследования работы исполняемого файла.

15.3.4.3 Подмена информации

Ложный объект позволяет не только модифицировать, но и подменять перехваченную им информацию. *При возникновении в сети определенного контролируемого ложным объектом события одному из участников обмена посылается заранее подготовленная дезинформация.* При этом такая дезинформация в зависимости от контролируемого события может быть воспринята либо как исполняемый код, либо как данные.

Рассмотрим пример подобного рода дезинформации. Предположим, что ложный объект контролирует событие, которое состоит в подключении пользователя к серверу. В этом случае он ожидает, например, запуска соответствующей программы входа в систему. В случае, если эта программа находится на сервере, то при ее запуске исполняемый файл передается на

рабочую станцию. Вместо того, чтобы выполнить данное действие, ложный объект передает на рабочую станцию код заранее написанной специальной программы - захватчика паролей. Эта программа выполняет визуально те же действия, что и настоящая программа входа в систему, например, запрашивая имя и пароль пользователя, после чего полученные сведения посылаются на ложный объект, а пользователю выводится сообщение об ошибке. При этом пользователь, посчитав, что он неправильно ввел пароль (пароль обычно не отображается на экране) снова запустит программу подключения к системе (на этот раз настоящую) и со второго раза получит доступ. Результат такой атаки - имя и пароль пользователя, сохраненные на ложном объекте.

15.3.5 Отказ в обслуживании

В общем случае в РВС каждый субъект системы должен иметь возможность подключиться к любому объекту РВС и получить в соответствии со своими правами удаленный доступ к его ресурсам.

Обычно в вычислительных сетях возможность предоставления удаленного доступа реализуется следующим образом: на объекте РВС в сетевой ОС запускаются на выполнение ряд программ-серверов (например, FTP-сервер, WWW-сервер и т.п.), предоставляющих удаленный доступ к ресурсам данного объекта. В случае получения запроса на соединение сервер должен по возможности передать на запросивший объект ответ, в котором либо разрешить подключение, либо нет. Очевидно, что сетевая ОС способна отвечать лишь на ограниченное число запросов. Эти ограничения зависят от параметров ОС и ЭВМ, основными из которых являются быстродействие ЭВМ, объем оперативной памяти и пропускная способность канала связи.

Различают три типа удаленных атак «отказа в обслуживании»:

- Если инфраструктура РВС позволяет с одного объекта системы передавать на другой атакуемый объект бесконечное число анонимных запросов на подключение от имени других объектов, то в этом случае будет иметь успех типовая удаленная атака **«Отказ в обслуживании»**. Результат применения этой удаленной атаки - нарушение на атакованном объекте работоспособности соответствующей службы предоставления удаленного доступа, то есть невозможность получения удаленного доступа с других объектов РВС - отказ в обслуживании!
- Вторая разновидность этой типовой удаленной атаки состоит в передаче с одного адреса такого количества запросов на атакуемый объект, какое позволяет пропускная способность канала связи (**направленный «шторм» запросов**). В этом случае, если в системе не предусмотрены правила, ограничивающие число принимаемых запросов с одного объекта (адреса) в единицу времени, то результатом этой атаки может являться как переполнение очереди запросов и отказа одной из телекоммуникационных служб, так и

полная остановка компьютера из-за невозможности системы заниматься ничем другим, кроме обработки запросов.

- Третьей разновидностью атаки «Отказ в обслуживании» является передача на атакуемый объект некорректного, специально подобранного запроса. В этом случае при наличии ошибок в удаленной системе возможно **зацикливание процедуры обработки запроса**, переполнение буфера с последующим зависанием системы.

Типовая удаленная атака «Отказ в обслуживании» является активным воздействием (класс 1.2), осуществляемым с целью нарушения работоспособности системы (класс 2.3), безусловно относительно цели атаки (класс 3.3). Данная УА является однонаправленным воздействием (класс 4.2), как межсегментным (класс 5.1), так и внутрисегментным (класс 5.2), осуществляемым на транспортном (класс 6.4) и прикладном (класс 6.7) уровнях модели OSI.

Соответствие рассматренных типовых удаленных атак классификации приведена в таблице 15.1.

Таблица 15.1 - Классификация типовых удаленных атак на РВС

Типовая удаленная атака	Характер воздействия		Цель воздействия			Условие начала осуществления воздействия			Наличие обратной связи с атакуемым объектом		Расположение субъекта атаки относительно атакуемого объекта		Уровень модели OSI						
Класс воздействия	1.1	1.2	2.1	2.2	2.3	3.1	3.2	3.3	4.1	4.2	5.1	5.2	6.1	6.2	6.3	6.4	6.5	6.6	6.7
Анализ сетевого трафика	+	-	+	-	-	-	-	+	-	+	+	-	-	+	-	-	-	-	-
Подмена доверенного объекта РВС	-	+	+	+	-	-	+	-	+	+	+	+	-	-	+	+	-	-	-
Внедрение в РВС ложного объекта путем навязывания ложного маршрута	-	+	+	+	+	-	-	+	+	+	+	+	-	-	+	-	-	-	-
Внедрение в РВС ложного объекта за счет недостатков алгоритмов удаленного поиска	-	+	+	+	-	+	-	+	+	-	+	+	-	+	+	+	-	-	-
Отказ в обслуживании	-	+	-	-	+	-	-	+	-	+	+	+	-	+	+	+	+	+	+

15.4 Анализ типовых уязвимостей позволяющих реализовать успешные удаленные атаки

Анализ механизмов реализации типовых УА и их практическое осуществление на примере сети Internet позволили сформулировать набор типовых уязвимостей и причин, по которым данные удаленные атаки оказались возможными. Особо отметим, что рассматриваемые ниже уязвимости основываются на *базовых принципах* построения сетевого взаимодействия объектов РВС.

Для устранения причин атак зачастую необходимо либо отказаться от определенных служб (DNS, например), либо изменить конфигурацию системы (наличие широковещательной среды приводит к возможности прослушивания канала, осуществляемого программным образом), либо изменить систему в целом. Все дело в том, что причины успеха удаленных атак данного типа кроются в инфраструктуре РВС, поэтому создание таксономии причин их успеха представляется весьма важной задачей, решение которой позволит выработать принципы построения защищенного взаимодействия в РВС.

Итак, рассмотрим возможные причины успеха УА на инфраструктуру и базовые протоколы распределенных ВС.

15.4.1 Отсутствие выделенного канала связи между объектами системы

Атака «Анализ сетевого трафика» заключается в прослушивании канала передачи сообщений в сети. Результат этой атаки во-первых, выяснение логики работы распределенной ВС и, во-вторых, перехват потока информации, которой обмениваются объекты системы. Такая атака программно возможна только в случае, если атакующий находится в сети с физически широковещательной средой передачи данных как, например, всем известная и получившая широкое распространение среда Ethernet. Очевидно, что данная УА была бы программно невозможна, если бы у каждого объекта системы существовал для связи с любым другим объектом выделенный канал (вариант физического прослушивания выделенного канала не рассматривается, так как без специфических аппаратных средств подключение к выделенному каналу невозможно).

Следовательно, причина успеха данной типовой УА - наличие широковещательной среды передачи данных или отсутствие выделенного канала связи между объектами РВС.

15.4.2 Недостаточная идентификация и аутентификация объектов и субъектов системы

Как уже подчеркивалось, проблема идентификации и аутентификации субъектов и объектов РВС имеет чрезвычайно важное значение. От успеха ее решения зависит безопасность РВС в целом. Примеры успешно

осуществленных удаленных атак, доказывают, что отсутствие у разработчиков определенной заранее выработанной концепции и принципов идентификации объектов РВС в целом оставляют атакующему потенциальные возможности для компрометации объектов системы. Стандартными способами компрометации субъектов и объектов РВС являются:

- выдача себя за определенный объект или субъект с присвоением его прав и полномочий для доступа в систему (например, типовая УА «Подмена доверенного субъекта или объекта РВС»);
- внедрение в систему ложного объекта, выдающего себя за доверенный объект системы (например типовая УА «Ложный объект РВС»).

15.4.2.1 Взаимодействие объектов без установления виртуального канала

Одним из важнейших вопросов, на который необходимо ответить, говоря об идентификации/аутентификации объектов/субъектов РВС, является вопрос о видах взаимодействия между субъектами и объектами в распределенной ВС. Взаимодействие между субъектами и объектами РВС бывает двух видов:

- с использованием виртуального канала (ВК),
- без использования виртуального канала.

Практика показывает, что 99 % взаимодействия между объектами в сети Internet проходит с установлением ВК (при любом FTP-, TELNET-, HTTP- и т. п. подключении используется протокол TCP, а, следовательно, создается ВК). Это происходит из-за того, что взаимодействие по виртуальному каналу является единственным динамическим способом защиты сетевого соединения объектов РВС. Дело в том, что в процессе создания ВК объекты РВС обмениваются динамически вырабатываемой ключевой информацией, позволяющей уникально идентифицировать канал.

Таким образом, идентификация объектов РВС, при отсутствии статической ключевой информации, возможна только при взаимодействии объектов с использованием виртуального канала. Это, в свою очередь, означает, что взаимодействие объектов без установления ВК является одной из возможных причин успеха удаленных атак на РВС.

Но ошибочно считать распределенную вычислительную систему безопасной, даже если все взаимодействие объектов происходит с созданием ВК. Об этом речь пойдет в следующем пункте.

15.4.2.1 Использование нестойких алгоритмов идентификации объектов при создании виртуального канала

Ошибочно считать взаимодействие объектов по виртуальному каналу в РВС решением всех проблем, связанных с идентификацией объектов РВС.

ВК является необходимым, но не достаточным условием безопасного взаимодействия. Чрезвычайно важным в данном случае становится выбор алгоритма идентификации при создании ВК. Основное требование, которое следует предъявлять к данным алгоритмам, состоит в следующем: перехват ключевой информации, которой обмениваются объекты РВС при создании ВК не должен позволить атакующему получить итоговые идентификаторы канала и объектов. Это требование по сути очевидно. Оно должно предъявляться к алгоритмам идентификации исходя из принципиальной возможности прослушивания атакующим канала передачи. Однако в большинстве существующих сетевых ОС в базовых алгоритмах идентификации, используемых при создании ВК, этим требованием разработчики практически пренебрегают.

Так, например, в ОС Novell NetWare 3.12- 4.1 идентификатор канала - это число в диапазоне 0-FFh, идентификатор объекта (рабочей станции или файл-сервера) - также число от 0 до FFh; в протоколе TCP идентификаторами канала и объектов являются два 32-битных числа, формируемых в процессе создания TCP-соединения.

Из всего сказанного ясно, что создание виртуального канала с использованием нестойкого алгоритма идентификации не позволяет надежно обезопасить РВС от подмены объектов взаимодействия и выступает одной из причин успеха удаленных атак на распределенные вычислительные системы.

15.4.3 Отсутствие контроля за виртуальными каналами связи между объектами системы

Объекты РВС, взаимодействующие по виртуальным каналам, могут подвергаться типовой УА «Отказ в обслуживании». Особенность этой атаки состоит в том, что, действуя абсолютно легальными средствами системы, можно удаленно добиться нарушения ее работоспособности. Данная УА реализуется передачей множественных запросов на создание соединения (виртуального канала), в результате чего либо переполняется число возможных соединений, либо система, занятая обработкой ответов на запросы, вообще перестает функционировать.

Взаимодействие объектов РВС по виртуальным каналам позволяет единственным способом обеспечить защиту соединения в глобальной сети. Однако в использовании ВК есть как несомненные плюсы, так и очевидные минусы. К минусам относится необходимость контроля над соединением. При этом задача контроля распадается на две подзадачи:

- контроль за созданием соединения;
- контроль за использованием соединения.

Если задача контроля за использованием соединения решается довольно просто (обычно соединение разрывается по тайм-ауту, определенному системой - так сделано во всех известных сетевых ОС), то решение задачи контроля за созданием соединения представляется

нетривиальным. Именно отсутствие приемлемого решения этой задачи является основной причиной успеха типовой УА «Отказ в обслуживании». Сложность контроля над созданием ВК состоит в том, что в системе, в которой отсутствует статическая ключевая информация о всех ее объектах, невозможно отделить ложные запросы на создание соединения от настоящих.

Очевидно также, что если ***один*** субъект сетевого взаимодействия будет иметь возможность ***анонимно*** занимать ***неограниченное*** число каналов связи с удаленным объектом, то подобная система может быть полностью парализована данным субъектом (пример - существующая сеть Internet в стандарте IPv4)! Поэтому, если любой объект в распределенной системе может анонимно послать сообщение от имени любого другого объекта (например, в Internet маршрутизаторы не проверяют IP-адрес источника отправления), то в подобной РВС в принципе невозможен контроль за созданием виртуальных соединений. Поэтому основная причина, по которой возможна типовая УА «Отказ в обслуживании» и ей подобные - это отсутствие в РВС возможности контроля за маршрутом сообщений.

15.4.4 Отсутствие возможности контроля за маршрутом сообщений

В РВС в качестве начальной идентифицирующей объект информации обычно выступает его адрес. Под адресом в РВС понимается определенная системой уникальная информация, которой он наделяется при внесении в систему. Все сообщения от других объектов РВС, адресованные на этот адрес, поступят на данный объект. Путь, или, маршрут сообщения определяется топологией РВС и проходит через совокупность узлов-маршрутизаторов. Следовательно, в каждом приходящем на объект РВС пакете может быть полностью отмечен его маршрут - список адресов маршрутизаторов, пройденных на пути к адресату. Этот отмеченный в пакете ***маршрут станет информацией, аутентифицирующей (подтверждающей) с точностью до подсети, подлинность адреса субъекта, отославшего сообщение***. Другой вариант аутентификации адреса отправителя - фильтрация маршрутизатором пакетов с неверным адресом отправителя.

Если в РВС не предусмотреть подобных возможностей контроля за маршрутом сообщения, то адрес отправителя сообщения оказывается ничем не подтвержден. Таким образом, в системе будет существовать возможность отправки сообщения от имени любого объекта системы, а именно путем указания в заголовке сообщения чужого адреса отправителя. Также в подобной РВС будет невозможно определить, откуда на самом деле пришло сообщение, а, следовательно, вычислить координаты атакующего (в сети Internet невозможно доступным способом вычислить инициатора однонаправленной удаленной атаки).

Таким образом, мы убеждаемся, что отсутствие в распределенной ВС возможности контроля за маршрутом сообщений порождает, во-первых,

невозможность контроля за созданием соединений, и, во-вторых, возможность анонимной отправки сообщения, следовательно является причиной успеха удаленных атак на РВС.

15.4.5 Отсутствие в системе полной информации о ее объектах

В распределенной системе с разветвленной структурой, состоящей из большого числа объектов, может возникнуть ситуация, когда для доступа к определенному объекту системы у субъекта взаимодействия может не оказаться необходимой информации об интересующем объекте. Обычно такой недостающей информацией об объекте является его адрес. Такая ситуация характерна и вполне объяснима для сетей с разветвленной структурой.

Объясним это на простом примере. Предположим, что пользователь сети Internet решил подключиться, например, к WWW-серверу фирмы Novell. Он знает ее название, но не имеет информации об IP-адресе или имени ее сервера. В этом случае пользователь может послать широковещательный запрос всем хостам в сети с надеждой, что запрос дойдет до интересующего его сервера, и тот в ответ пришлет столь нужный для пользователя адрес. Очевидно, что в глобальной сети использование данной схемы по меньшей мере неразумно. Поэтому для подобных целей пользователь может подключиться к ближайшему известному ему поисковому серверу (***Altavista***, например) и послать запрос на поиск адреса интересующей его фирмы в базе данных информационного сервера.

Рассмотренный выше пример наглядно описывает возможные алгоритмы удаленного поиска, которые используют объекты РВС:

- когда поиск осуществляется внутри сегмента сети, субъект системы посылает широковещательный запрос, который получают все объекты РВС, и тот из них, для кого предназначался запрос, передает в ответ необходимую для адресации информацию.
- когда необходимо осуществить глобальный поиск, субъект распределенной системы посылает запрос на ближайший информационно-поисковый сервер, который, просканировав свою базу данных в поисках адреса запрашиваемого ресурса, либо отошлет в ответ на запрос найденный адрес, либо обратится к следующему в системе поисково-информационному серверу.

Таким образом, если в распределенной ВС существуют объекты, информация о которых не определена, то для обеспечения ее нормального функционирования необходимо использование описанных выше алгоритмов удаленного поиска.

Примером РВС с заложенной неопределенностью является сеть Internet, в которой, во-первых, у хостов, находящихся в одном сегменте, может не быть информации об аппаратных адресах друг друга, и, во-вторых, применяются непригодные для непосредственной адресации мнемонические

имена хостов, используемые для удобства пользователей при обращении к удаленным системам.

Очевиден тот факт, что в системе с заложенной в нее неопределенностью существуют потенциальные возможности внесения в систему ложного объекта и выдачи одного объекта системы за другой. Этот факт объясняется тем, что, являясь следствием неопределенности системы, алгоритмы удаленного поиска несут в себе потенциальную угрозу, состоящую в том, что на посланный запрос может прийти ложный ответ, в котором вместо информации о запрашиваемом объекте будет информация о ложном объекте. Вследствие этого распределенная ВС с заложенной неопределенностью является потенциально опасной системой и может подвергаться удаленным атакам.

15.4.6 Отсутствие криптозащиты сообщений

В РВС связь между объектами системы осуществляется по каналам связи. Поэтому всегда существует принципиальная возможность для атакующего прослушать канал и получить несанкционированный доступ к информации, которой обмениваются по сети ее абоненты. В том случае, если проходящая по каналу информация не зашифрована и атакующий каким-либо образом получает доступ к каналу, то УА «Анализ сетевого трафика» является наиболее эффективным способом получения информации. Очевидна и причина, делающая эту атаку столь эффективной. Эта причина - передача по сети незашифрованной информации.

Использование криптостойких алгоритмов шифрования пакетов обмена между объектами РВС на канальном, прикладном уровнях делает анализ сетевого трафика практически бессмысленным. В случае канального шифрования, которое обычно выполняется аппаратно, по сети передаются полностью зашифрованные пакеты. В том случае, если в сети используются алгоритмы шифрования пакетов на сетевом - прикладном уровнях, то шифрация применяется только к полям данных пакетов соответствующих уровней, то есть заголовки пакетов, содержащие служебную информацию, не являются зашифрованными, поэтому атакующий имеет возможность, перехватив пакет, подвергнуть анализу данную служебную информацию.

16. МЕХАНИЗМЫ РЕАЛИЗАЦИИ УДАЛЕННЫХ АТАК В ГЛОБАЛЬНОЙ СЕТИ INTERNET

В настоящее время возможности по реализации удаленных атак в сети Internet настолько многообразны, что рассмотреть их все непредставляется возможным. Исследованиям уязвимостей отдельных операционных систем и особенностям функционирования операционных систем посвящено большое количество соответсвующей литературы. Цель данного раздела, не производя исчерпующего анализа механизмов отдельных удаленных атак, на отдельных простых примерах продемонстрировать реализации типовых уязвимостей рабочих станций в сети Internet.

16.1 Анализ сетевого трафика

В сети Internet основными базовыми протоколами удаленного доступа являются TELNET и FTP (File Transfer Protocol). TELNET - это протокол виртуального терминала (ВТ), позволяющий с удаленных хостов подключаться к серверам Internet в режиме ВТ. FTP - протокол, предназначенный для передачи файлов между удаленными хостами. Для получения доступа к серверу по данным протоколам пользователю необходимо пройти на нем процедуру идентификации и аутентификации. В качестве информации, идентифицирующей пользователя, выступает его идентификатор (имя), а для аутентификации используется пароль. Особенностью протоколов FTP и TELNET является то, что пароли и идентификаторы пользователей передаются по сети в открытом, незашифрованном виде. Таким образом, необходимым и достаточным условием для получения удаленного доступа к хостам по протоколам FTP и TELNET являются имя и пароль пользователя.

Одним из способов получения паролей и идентификаторов пользователей в сети Internet является анализ сетевого трафика. Сетевой анализ осуществляется с помощью специальной программы-анализатора пакетов, перехватывающей все пакеты, передаваемые по сегменту сети, и выделяющей среди них те, в которых передаются идентификатор пользователя и его пароль (рис. 16.1). Сетевой анализ протоколов FTP и TELNET показывает, что TELNET разбивает пароль на символы и пересылает их по одному, помещая каждый символ из пароля в соответствующий пакет, а FTP, напротив, пересылает пароль целиком в одном пакете.

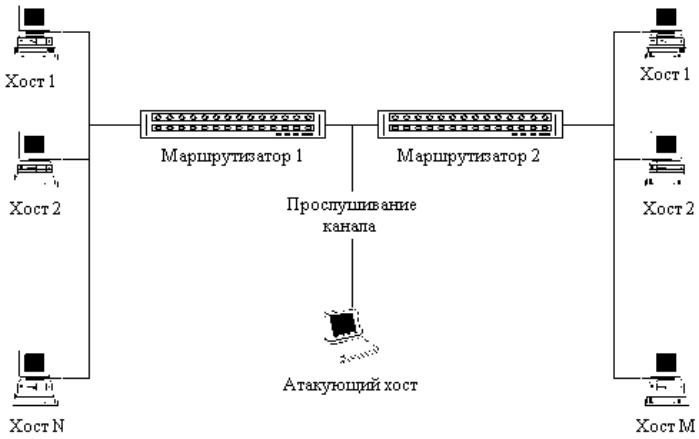


Рис. 16.1 - Анализ сетевого трафика

16.2 Ложный ARP-сервер

В вычислительных сетях связь между двумя удаленными хостами осуществляется путем передачи по сети сообщений, которые заключены в пакеты обмена. В общем случае передаваемый по сети пакет независимо от используемого протокола и типа сети (Token Ring, Ethernet, X.25 и др.) состоит из заголовка пакета и поля данных. В заголовок пакета обычно заносится служебная информация, определяемая используемым протоколом обмена и необходимая для адресации пакета, его идентификации, преобразования и т. д. В поле данных помещаются либо непосредственно данные, либо другой пакет более высокого уровня OSI.

Так, например, пакет транспортного уровня может быть вложен в пакет сетевого уровня, который, в свою очередь, вложен в пакет канального уровня. Таким образом, пакет TCP (транспортный уровень) вложен в пакет IP (сетевой уровень), который, в свою очередь, вложен в пакет Ethernet (канальный уровень). Схема на рис. 16.2 наглядно иллюстрирует как выглядит, например, TCP-пакет в сети Internet.

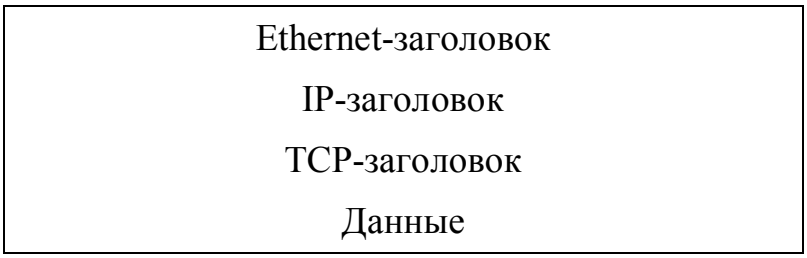


Рис. 16.2 - Структура TCP-пакета

Базовым сетевым протоколом обмена в сети Internet является протокол IP (Internet Protocol). Протокол IP - это межсетевой протокол, позволяющий передавать IP-пакеты в любую точку глобальной сети. Для адресации на сетевом уровне (IP-уровне) в сети Internet каждый хост имеет уникальный 32-разрядный IP-адрес. Для передачи IP-пакета на хост необходимо указать в IP-

заголовке пакета в поле Destination Address IP-адрес данного хоста. Однако, как видно из рис. 16.2, IP-пакет находится внутри Ethernet-пакета, поэтому каждый пакет в конечном счете адресуется на аппаратный адрес сетевого адаптера, непосредственно осуществляющего прием и передачу пакетов в сеть (при рассмотрении Ethernet-сети).

То есть, для адресации IP-пакетов в сети Internet кроме IP-адреса хоста необходим еще либо Ethernet-адрес его сетевого адаптера (в случае адресации внутри одной подсети), либо Ethernet-адрес маршрутизатора (в случае межсетевой адресации). Первоначально хост может не иметь информации о Ethernet-адресах других хостов, находящихся с ним в одном сегменте, в том числе и о Ethernet-адресе маршрутизатора. Следовательно, перед хостом встает стандартная проблема, решаемая с помощью алгоритма удаленного поиска.

В сети Internet для решения проблемы удаленного поиска Ethernet-адресов используется протокол ARP (Address Resolution Protocol). Протокол ARP позволяет получить взаимно однозначное соответствие IP- и Ethernet-адресов для хостов, находящихся внутри одного сегмента.

Это достигается следующим образом:

- При первом обращении к сетевым ресурсам хост отправляет широковещательный ARP-запрос на Ethernet-адрес FFFFFFFFFFFFh, в котором указывает IP-адрес маршрутизатора и просит сообщить его Ethernet-адрес. Этот широковещательный запрос получат все станции в данном сегменте сети, в том числе и маршрутизатор.
- Получив данный запрос, маршрутизатор внесет запись о запросившем хосте в свою ARP-таблицу, а затем отправит на запросивший хост ARP-ответ, в котором сообщит свой Ethernet-адрес.
- Полученный в ARP-ответе Ethernet-адрес будет занесен в ARP-таблицу, находящуюся в памяти операционной системы на запросившем хосте и содержащую записи соответствия IP- и Ethernet-адресов для хостов внутри одного сегмента.

В случае использования в РВС алгоритмов удаленного поиска существует возможность осуществления в такой сети типовой удаленной атаки «Ложный объект РВС». Из анализа безопасности протокола ARP становится ясно, что, перехватив на атакующем хосте внутри данного сегмента сети широковещательный ARP-запрос, можно послать ложный ARP-ответ, в котором объявить себя искомым хостом (например, маршрутизатором), и в дальнейшем активно контролировать и воздействовать на сетевой трафик «обманутого» хоста по схеме «Ложный объект РВС».

Рассмотрим обобщенную функциональную схему ложного ARP-сервера (рис. 16.3):

- ожидание ARP-запроса;

- при получении ARP-запроса передача по сети на запросивший хост ложного ARP-ответа, в котором указывается адрес сетевого адаптера атакующей станции (ложного ARP-сервера) или тот Ethernet-адрес, на котором будет принимать пакеты ложный ARP-сервер (совершенно необязательно указывать в ложном ARP-ответе свой настоящий Ethernet-адрес, так как при работе непосредственно с сетевым адаптером его можно запрограммировать на прием пакетов на любой Ethernet-адрес);
- прием, анализ, воздействие и передача пакетов обмена между взаимодействующими хостами, а также по возможности воздействие на перехваченную информацию.

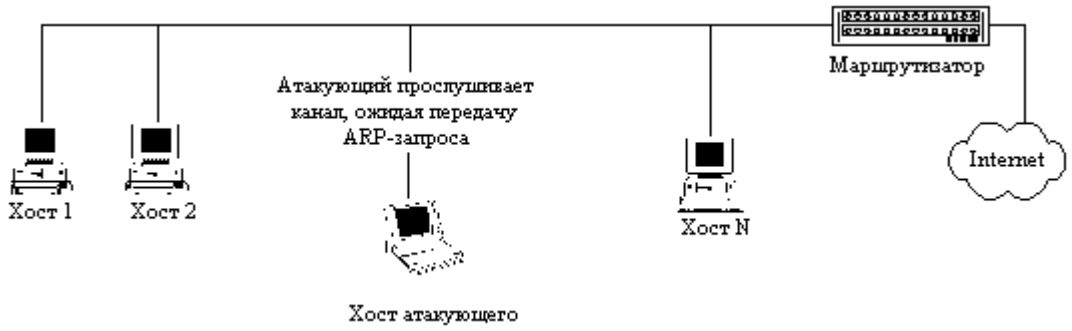


Рис. 16.3а - Фаза ожидания ARP-запроса

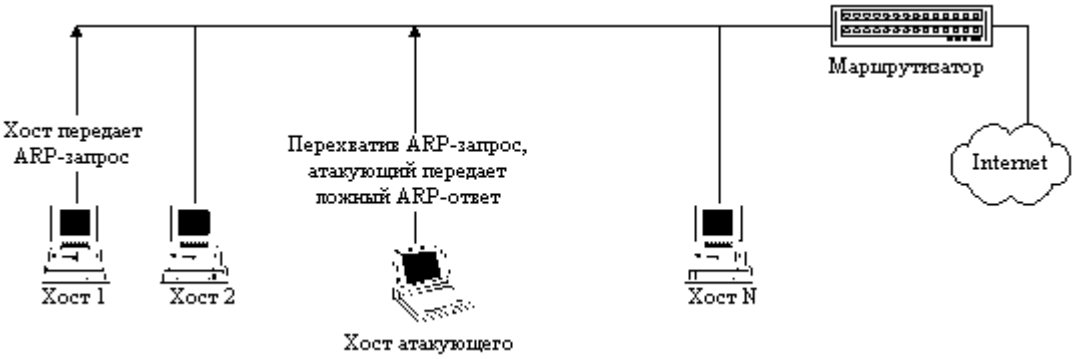


Рис. 16.3б - Фаза атаки

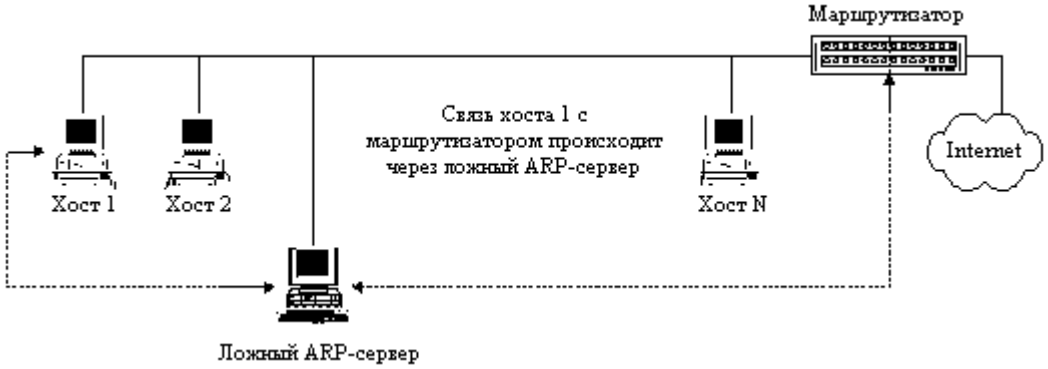


Рис. 16.3в - Фаза приема, анализа, воздействия и передачи перехваченной информации на ложном ARP-сервере

Рис. 16.3 - Ложный ARP-сервер

В заключение необходимо отметить, что, во-первых, причина успеха данной удаленной атаки кроется, не столько в Internet, сколько в широковещательной среде Ethernet и, во-вторых, очевидно, что эта удаленная атака является внутрисегментной и поэтому представляет угрозу только в случае нахождения атакующего внутри вашего сегмента сети.

16.3 Ложный DNS-сервер

Как известно, для обращения к хостам в сети Internet используются 32-разрядные IP-адреса, уникально идентифицирующие каждый сетевой компьютер в этой глобальной сети. Однако, для пользователей применение IP-адресов при обращении к хостам является не слишком удобным и далеко не самым наглядным. Использование в Internet породило проблему преобразования имен в IP-адреса. Такое преобразование необходимо, так как на сетевом уровне адресация пакетов идет не по именам, а по IP-адресам, следовательно, для непосредственной адресации сообщений в Internet имена не годятся. Для решения задачи преобразования мнемонически понятных для пользователей имен в IP-адреса была создана система преобразования имен, позволяющая хосту в случае отсутствия у него информации о соответствии имен и IP-адресов получить необходимые сведения от ближайшего информационно-поискового сервера – DNS (Domain Name System)-сервера.

Основной задачей, решаемой службой DNS является поиск по имени удаленного хоста его IP-адреса, который и необходим для непосредственной адресации.

Рассмотрим DNS-алгоритм удаленного поиска IP-адреса по имени в сети Internet.

- Хост посылает на IP-адрес ближайшего DNS-сервера (он устанавливается при настройке сетевой ОС) DNS-запрос, в котором указывает имя сервера, IP-адрес которого необходимо найти.
- DNS-сервер, получив запрос, просматривает свою базу имен на наличие в ней указанного в запросе имени. В случае, если имя найдено, а, следовательно, найден и соответствующий ему IP-адрес, то на запросивший хост DNS-сервер отправляет DNS-ответ, в котором указывает искомый IP-адрес.
- В случае, если указанное в запросе имя DNS-сервер не обнаружил в своей базе имен, то DNS-запрос отсылается DNS-сервером на один из корневых DNS-серверов и описанная в этом пункте процедура повторяется, пока имя не будет найдено (или не найдено).

Анализируя с точки зрения безопасности уязвимость этой схемы удаленного поиска с помощью протокола DNS, можно сделать вывод о возможности осуществления в сети, использующей протокол DNS, типовой удаленной атаки «Ложный объект РВС». Практические изыскания и

критический анализ безопасности службы DNS позволяют предложить три возможных варианта удаленной атаки на эту службу.

16.3.1 Внедрение в сеть Internet ложного DNS-сервера путем перехвата DNS-запроса

Для реализации атаки путем перехвата DNS-запроса атакующему необходимо перехватить DNS-запрос, извлечь из него номер UDP-порта отправителя запроса, двухбайтовое значение ID идентификатора DNS-запроса и искомое имя и затем послать ложный DNS-ответ на извлеченный из DNS-запроса UDP-порт, в котором указать в качестве искомого IP-адреса настоящий IP-адрес ложного DNS-сервера. Это позволит в дальнейшем полностью перехватить трафик между атакуемым хостом и сервером и активно воздействовать на него по схеме «Ложный объект РВС».

Рассмотрим обобщенную схему работы ложного DNS-сервера (рис. 16.4):

- ожидание DNS-запроса;
- извлечение из полученного запроса необходимых сведений и передача по сети на запросивший хост ложного DNS-ответа, от имени (с IP-адреса) настоящего DNS-сервера, в котором указывается IP-адрес ложного DNS-сервера;
- в случае получения пакета от хоста, изменение в IP-заголовке пакета его IP-адреса на IP-адрес ложного DNS-сервера и передача пакета на сервер (то есть ложный DNS-сервер ведет работу с сервером от своего имени);
- в случае получения пакета от сервера, изменение в IP-заголовке пакета его IP-адреса на IP-адрес ложного DNS-сервера и передача пакета на хост (для хоста ложный DNS-сервер и есть настоящий сервер).

Необходимым условием осуществления данного варианта атаки является перехват DNS-запроса. Это возможно только в том случае, если атакующий находится либо на пути основного трафика, либо в сегменте настоящего DNS-сервера. Выполнение одного из этих условий местонахождения атакующего в сети делает подобную удаленную атаку трудно осуществимой на практике. Однако в случае выполнения этих условий возможно осуществить ***межсегментную*** удаленную атаку на сеть Internet.

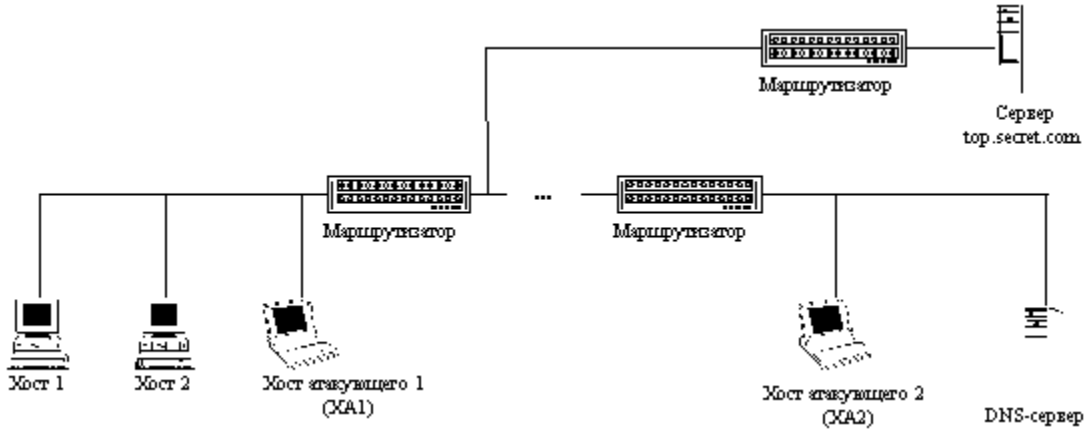


Рис. 16.4 а - Фаза ожидания атакующим DNS-запроса (он находится на ХА1, либо на ХА2)

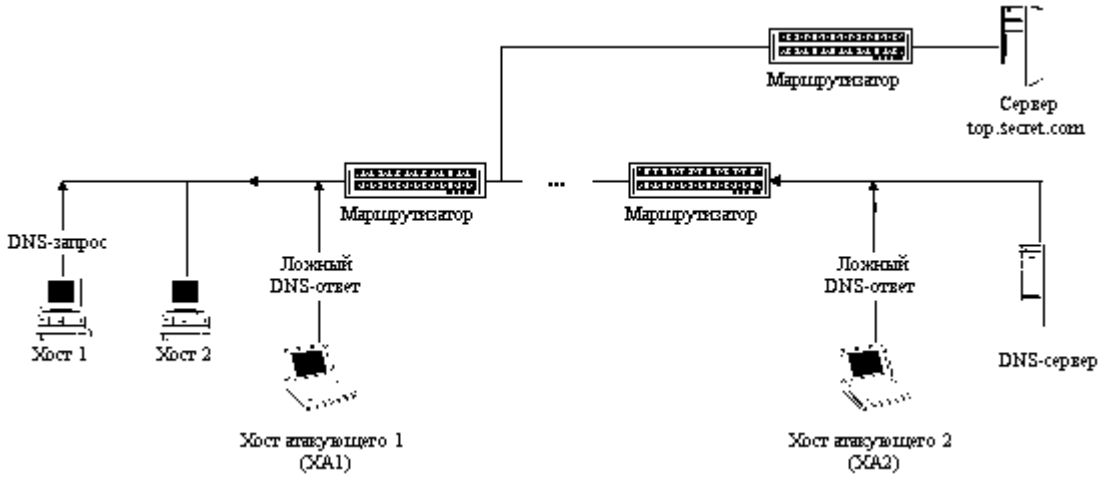


Рис. 16.4 б - Фаза передачи атакующим ложного DNS-ответа

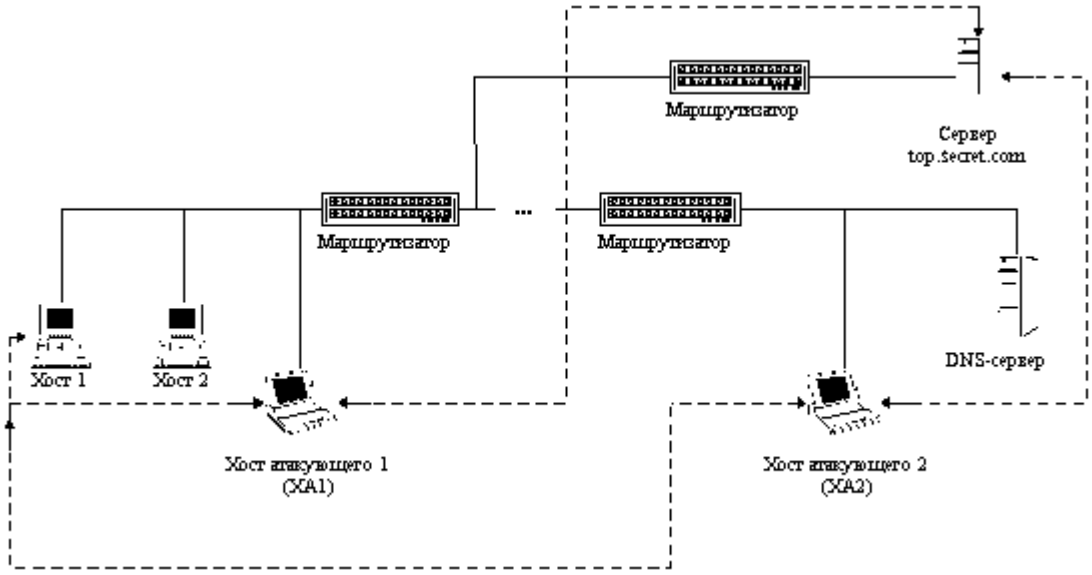


Рис. 16.4 в - Фаза приема, анализа, воздействия и передачи перехваченной информации на ложном сервере

Рис. 16.4 - Функциональная схема ложного DNS-сервера

16.3.2 Внедрение в сеть Internet ложного сервера путем создания направленного «шторма» ложных DNS-ответов на атакуемый хост

Другой вариант осуществления удаленной атаки, направленной на службу DNS, основан на второй разновидности типовой УА «Ложный объект РВС» (при использовании недостатков алгоритмов удаленного поиска). В этом случае атакующий осуществляет постоянную передачу на атакуемый хост заранее подготовленного ложного DNS-ответа от имени настоящего DNS-сервера *без приема DNS-запроса*! Другими словами, атакующий создает в сети Internet направленный «шторм» ложных DNS-ответов.

Это возможно, так как обычно для передачи DNS-запроса используется протокол UDP, в котором отсутствуют средства идентификации пакетов. Критериями, предъявляемыми сетевой ОС хоста к полученному от DNS-сервера ответу, является:

- совпадение IP-адреса отправителя ответа с IP-адресом DNS-сервера;
- DNS-ответ должен содержать то же имя, что и в DNS-запросе,
- DNS-ответ должен быть направлен на тот же UDP-порт, с которого был послан DNS-запрос (в данном случае это первая проблема для атакующего),
- в DNS-ответе поле идентификатора запроса в заголовке DNS (ID) должно содержать то же значение, что и в переданном DNS-запросе (а это вторая проблема).

В данном случае, так как атакующий не имеет возможности перехватить DNS-запрос, то основную проблему для него представляет номер UDP-порта, с которого был послан запрос. Однако, как было отмечено ранее, номер порта отправителя принимает ограниченный набор значений (≥ 1023), поэтому атакующему достаточно действовать простым перебором, направляя ложные ответы на соответствующий перечень портов. На первый взгляд, второй проблемой может быть двухбайтовый идентификатор DNS-запроса, но, в связи с особенностями функционироания протокола DNS он либо равен единице, либо в случае DNS-запроса от Netscape Navigator (например) имеет значение близкое к нулю (один запрос - ID увеличивается на 1).

Поэтому для осуществления данной удаленной атаки атакующему необходимо выбрать интересующий его хост (например, ***top.secret.com***), маршрут к которому требуется изменить так, чтобы он проходил через ложный сервер - хост атакующего. Это достигается постоянной передачей (направленным «штормом») атакующим ложных DNS-ответов на атакуемый хост от имени настоящего DNS-сервера на соответствующие UDP-порты. В этих ложных DNS-ответах указывается в качестве IP-адреса хоста ***top.secret.com*** IP-адрес атакующего.

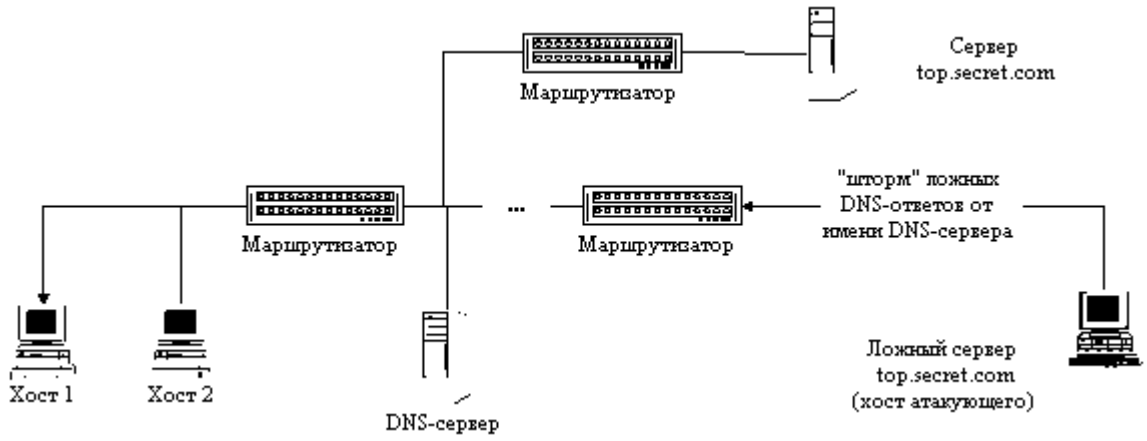


Рис. 16.5 а - Атакующий создает направленный «шторм» ложных DNS-ответов на Хост 1

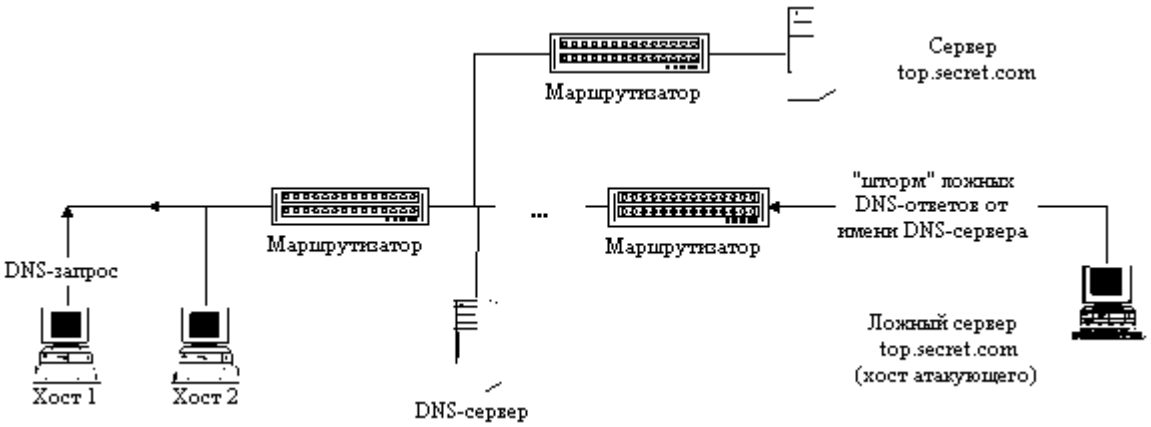


Рис. 16.5 б - Хост 1 посылает DNS-запрос и немедленно получает ложный DNS-ответ

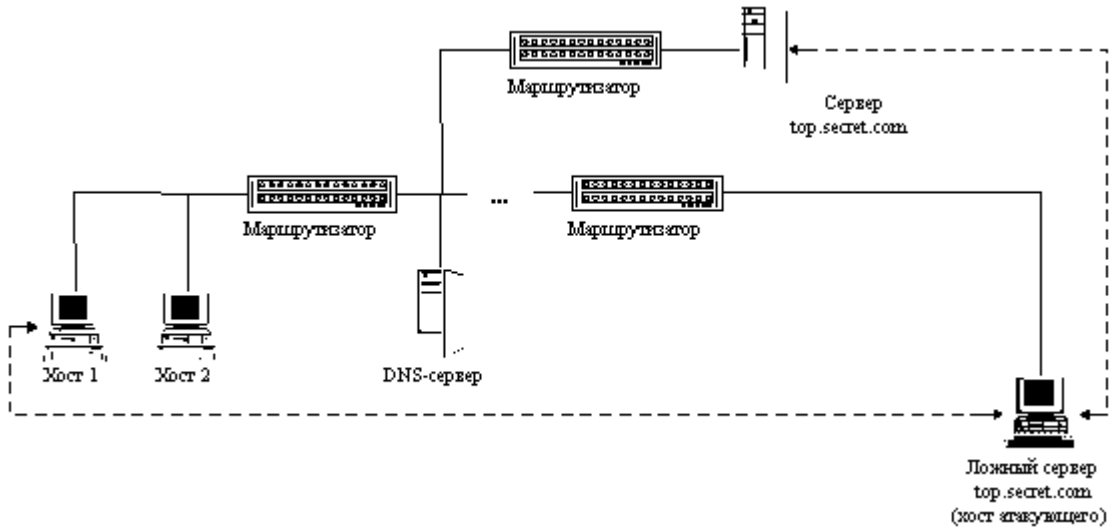


Рис. 16.5 в - Фаза приема, анализа, воздействия и передачи перехваченной информации на ложном сервере

Рис. 16.5 - Внедрение в Internet ложного сервера путем создания направленного «шторма» ложных DNS-ответов на атакуемый хост

Далее атака развивается по следующей схеме. Как только цель атаки (атакуемый хост) обратится по имени к хосту ***top.secret.com***, то от данного

хоста в сеть будет передан DNS-запрос, который атакующий никогда не получит, но этого ему и не требуется, так как на хост сразу же поступит постоянно передаваемый ложный DNS-ответ, что и будет воспринят ОС атакуемого хоста как настоящий ответ от DNS-сервера. Все! Атака состоялась, и теперь атакуемый хост будет передавать все пакеты, предназначенные для ***top.secret.com***, на IP-адрес хоста атакующего, который, в свою очередь, будет переправлять их на ***top.secret.com***, воздействуя на перехваченную информацию по схеме «Ложный объект РВС».

Рассмотрим функциональную схему предложенной удаленной атаки на службу DNS:

- постоянная передача атакующим ложных DNS-ответов на атакуемый хост на различные UDP-порты и, возможно, с различными ID, от имени (с IP-адреса) настоящего DNS-сервера с указанием имени интересующего хоста и его ложного IP-адреса, которым будет являться IP-адрес ложного сервера - хоста атакующего;
- в случае получения пакета от хоста, изменение в IP-заголовке пакета его IP-адреса на IP-адрес атакующего и передача пакета на сервер (то есть ложный сервер ведет работу с сервером от своего имени - со своего IP-адреса);
- в случае получения пакета от сервера, изменение в IP-заголовке пакета его IP-адреса на IP-адрес ложного сервера и передача пакета на хост (для хоста ложный сервер и есть настоящий сервер).

Таким образом, реализация данной удаленной атаки, использующей пробелы в безопасности службы DNS, позволяет из любой точки сети Internet нарушить маршрутизацию между двумя заданными объектами (хостами)! Данная удаленная атака осуществляется межсегментно по отношению к цели атаки и угрожает безопасности любого хоста Internet, использующего обычную службу DNS.

16.3.3 Внедрение в сеть Internet ложного сервера путем перехвата DNS-запроса или создания направленного «шторма» ложных DNS-ответов на атакуемый DNS-сервер

Из рассмотренной схемы удаленного DNS-поиска следует, что в том случае, если указанное в запросе имя DNS-сервер не обнаружил в своей базе имен, то запрос отсылается сервером на один из корневых DNS-серверов, адреса которых содержатся в файле настроек сервера ***root.cache***.

Итак, в случае, если DNS-сервер не имеет сведений о запрашиваемом хосте, то он сам, пересылая запрос далее, является инициатором удаленного DNS-поиска. Поэтому ничто не мешает атакующему, действуя описанными в предыдущих пунктах методами, *перенести свою атаку непосредственно на DNS-сервер*. В качестве цели атаки теперь будет выступать не хост, а DNS-

сервер и ложные DNS-ответы будут направляться атакующим от имени корневого DNS-сервера на атакуемый DNS-сервер.

При этом важно учитывать следующую особенность работы DNS-сервера. Для ускорения работы каждый DNS-сервер кэширует в области памяти свою таблицу соответствия имен и IP-адресов хостов. В том числе в кэш заносится динамически изменяемая информация об именах и IP-адресах хостов, найденных в процессе функционирования DNS-сервера, а именно, если DNS-сервер, получив запрос, не находит у себя в кэш-таблице соответствующей записи, он пересылает ответ на следующий сервер и, получив ответ, заносит найденные сведения в кэш-таблицу в память. Таким образом, при получении следующего запроса DNS-серверу уже не требуется вести удаленный поиск, так как необходимые сведения уже находятся у него в кэш-таблице.

Из анализа только что подробно описанной схемы удаленного DNS-поиска становится очевидно, что в том случае, если в ответ на запрос от DNS-сервера атакующий направит ложный DNS-ответ (или в случае «шторма» ложных ответов будет вести их постоянную передачу), то в кэш-таблице сервера появится соответствующая запись с ложными сведениями и в дальнейшем все хосты, обратившиеся к данному DNS-серверу, будут дезинформированы, и при обращении к хосту, маршрут к которому атакующий решил изменить, связь с ним будет осуществляться через хост атакующего по схеме «Ложный объект РВС». И, что хуже всего, с течением времени эта ложная информация, попавшая в кэш DNS-сервера, будет распространяться на соседние DNS-серверы высших уровней, а, следовательно, все больше хостов в Internet будут дезинформированы и атакованы!

В том случае, если атакующий не может перехватить DNS-запрос от DNS-сервера, то для реализации атаки ему необходим «шторм» ложных DNS-ответов, направленный на DNS-сервер. При этом возникает следующая проблема, отличная от проблемы подбора портов в случае атаки, направленной на хост. Как уже отмечалось ранее, DNS-сервер, посылая запрос на другой DNS-сервер, идентифицирует этот запрос двухбайтовым значением (ID). Это значение увеличивается на единицу с каждым передаваемым запросом. Узнать атакующему это текущее значение идентификатора DNS-запроса не представляется возможным. Поэтому предложить что-либо, кроме перебора 2^{16} возможных значений ID, достаточно сложно. Зато исчезает проблема перебора портов, так как все DNS-запросы передаются DNS-сервером на 53 порт.

Следующая проблема, являющаяся условием осуществления этой удаленной атаки на DNS-сервер при направленном «шторме» ложных DNS-ответов, состоит в том, что атака будет иметь успех только в случае, если DNS-сервер пошлет запрос на поиск имени, которое содержится в ложном DNS-ответе. DNS-сервер посылает этот столь необходимый и желанный для атакующего запрос в том и только том случае, когда на него приходит DNS-

запрос от какого-либо хоста на поиск данного имени и этого имени не оказывается в кэш-таблице DNS-сервера. В принципе, этот запрос может возникнуть когда угодно, и атакующему придется ждать результатов атаки неопределенное время. Однако, ничто не мешает атакующему, не дожидаясь никого, самому послать на атакуемый DNS-сервер подобный DNS-запрос и ***спровоцировать*** DNS-сервер на поиск указанного в запросе имени. Тогда эта атака с большой вероятностью будет иметь успех практически сразу же после начала ее осуществления.

Для примера вспомним скандал (28 октября 1996 года) с одним из московских провайдеров Internet - компанией РОСНЕТ, когда пользователи данного провайдера при обращении к обычному информационному WWW-серверу попадали, как было сказано в телевизионном репортаже, WWW-сервер «сомнительного» содержания. В связи с абсолютным непониманием случившегося как журналистами (их можно понять - они не специалисты в этом вопросе), так и теми, кто проводил пресс-конференцию (специалистов к общению с прессой, наверное, просто не допустили) информационные сообщения о данном событии были настолько убоги, что понять, что случилось, было толком невозможно. Тем не менее, этот инцидент вполне укладывается в только что описанную схему удаленной атаки на DNS-сервер. С одним исключением: вместо адреса хоста атакующего в кэш-таблицу DNS-сервера был занесен IP-адрес хоста ***www.playboy.com***.

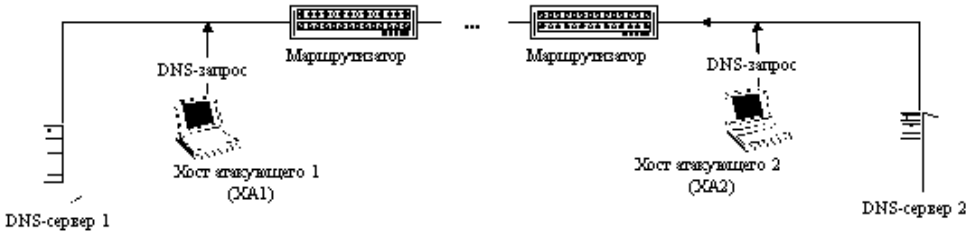


Рис. 16.6 а - Фаза ожидания атакующим DNS-запроса от DNS-сервера (для ускорения атакующий генерирует необходимый DNS-запрос)

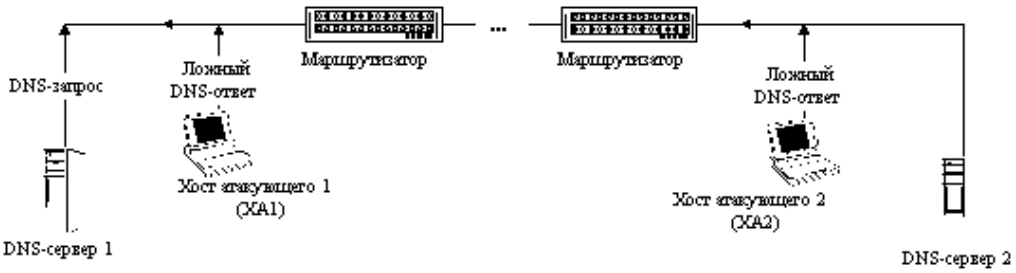


Рис. 16.6 б - Фаза передачи атакующим ложного DNS-ответа на DNS-сервер 1

Рис. 16.6 - Внедрение в Internet ложного сервера путем перехвата DNS-запроса от DNS-сервера

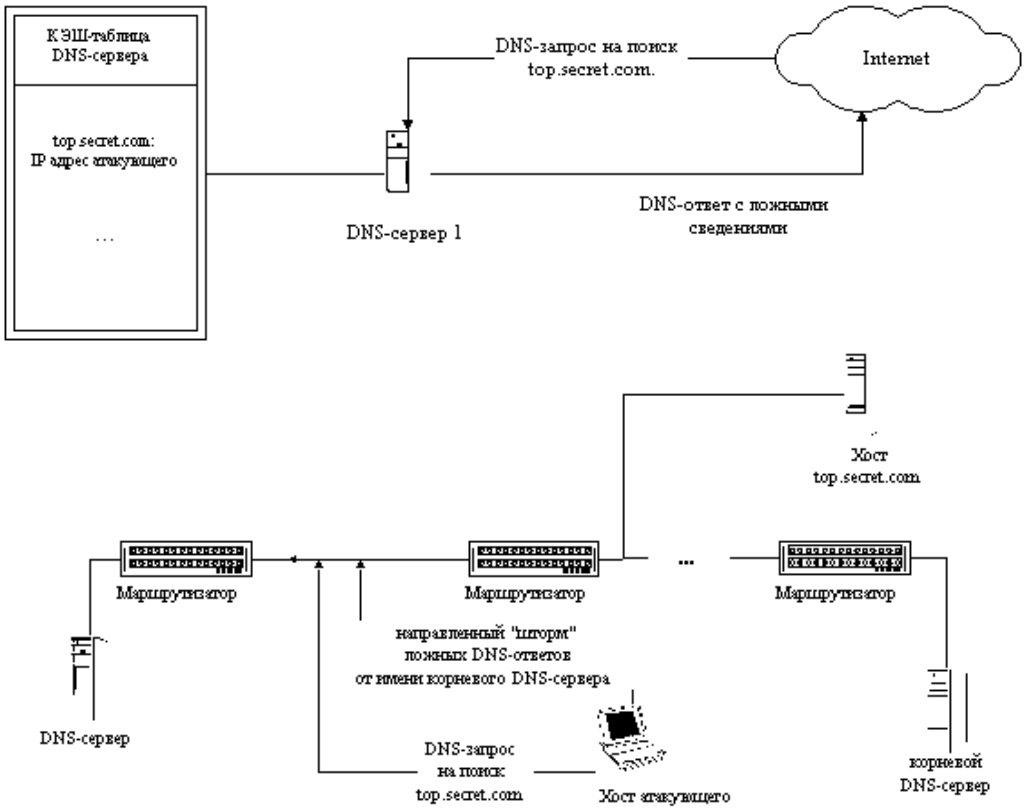


Рис16.7 а - Атакующий создает направленный «шторм» ложных DNS-ответов от имени одного из корневых DNS-серверов и при этом провоцирует атакуемый DNS-сервер, посылая DNS-запрос

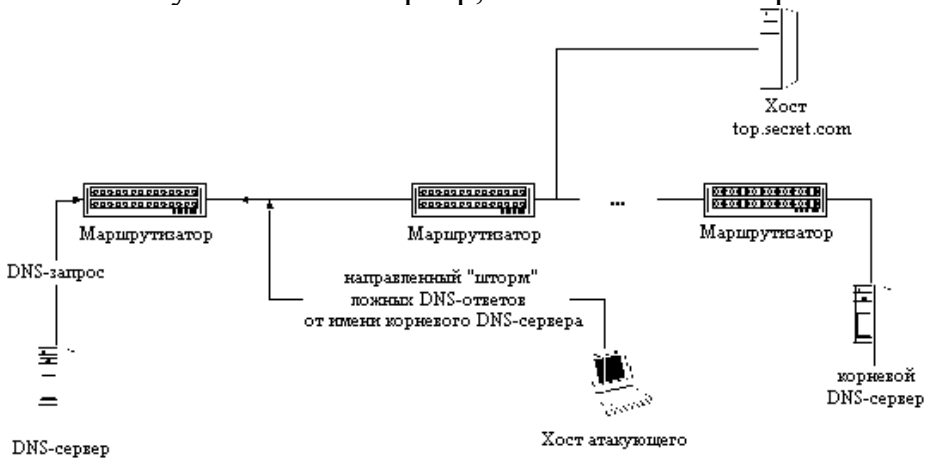


Рис16.7 б - DNS-сервер передает DNS-запрос на корневой DNS-сервер и немедленно получает ложный DNS-ответ от атакующего

Рис. 16.7 - Внедрение в Internet ложного сервера путем создания направленного «шторма» ложных DNS-ответов на атакуемый DNS-сервер

Использование в сети Internet службы удаленного поиска DNS позволяет атакующему организовать в Internet удаленную атаку на любой хост, пользующийся услугами данной службы, и может пробить серьезную брешь в безопасности этой и так отнюдь не безопасной глобальной сети.

16.4 Навязывание хосту ложного маршрута с использованием протокола ICMP с целью создания в сети Internet ложного маршрутизатора

В сети Internet используется управляющий протокол ICMP, одной из функций которого является удаленное управление маршрутизацией на хостах внутри сегмента сети. Удаленное управление маршрутизацией необходимо для предотвращения возможной передачи сообщений по неоптимальному маршруту. В сети Internet удаленное управление маршрутизацией реализовано в виде передачи с маршрутизатора на хост управляющего ICMP-сообщения: Redirect Message. Исследование протокола ICMP показало, что сообщение Redirect бывает двух типов:

- Первый тип сообщения носит название Redirect Net и уведомляет хост о необходимости смены адреса маршрутизатора, то есть default-маршрута.
- Второй тип - Redirect Host - информирует хост о необходимости создания нового маршрута к указанной в сообщении системе и внесения ее в таблицу маршрутизации. Для этого в сообщении указывается IP-адрес хоста, для которого необходима смена маршрута (адрес будет занесен в поле Destination), и новый IP-адрес маршрутизатора, на который необходимо направлять пакеты, адресованные данному хосту (этот адрес заносится в поле Gateway).

Необходимо обратить внимание на важное ограничение, накладываемое на IP-адрес нового маршрутизатора: он должен быть в пределах адресов данной подсети!

Что касается управляющего сообщения ICMP Redirect Host, то единственным идентифицирующим его параметром является IP-адрес отправителя, который должен совпадать с IP-адресом маршрутизатора, так как это сообщение может передаваться только маршрутизатором. Особенность протокола ICMP состоит в том, что он не предусматривает никакой дополнительной аутентификации источников сообщений. Таким образом, ICMP-сообщения передаются на хост маршрутизатором однонаправлено, без создания виртуального соединения.

Следовательно, ничто не мешает атакующему послать ложное ICMP-сообщение о смене маршрута от имени маршрутизатора. Приведенные выше факты позволяют осуществить типовую удаленную атаку «Внедрение в РВС ложного объекта путем навязывания ложного маршрута”.

Для осуществления этой удаленной атаки необходимо подготовить ложное ICMP Redirect Host сообщение, в котором указать конечный IP-адрес маршрута (адрес хоста, маршрут к которому будет изменен) и IP-адрес ложного маршрутизатора. Далее это сообщение передается на атакуемый хост от имени маршрутизатора. Для этого в IP-заголовке в поле адреса отправителя указывается IP-адрес маршрутизатора. В принципе, можно предложить два варианта данной удаленной атаки.

В первом случае атакующий находится в том же сегменте сети, что и цель атаки. Тогда, послав ложное ICMP-сообщение, он в качестве IP-адреса нового маршрутизатора может указать либо свой IP-адрес, либо любой из адресов данной подсети. Это даст атакующему возможность изменить маршрут передачи сообщений, направляемых атакованным хостом на определенный IP-адрес, и получить контроль над трафиком между атакуемым хостом и интересующим атакующего сервером. После этого атака перейдет во вторую стадию, связанную с приемом, анализом и передачей пакетов, получаемых от «обманутого» хоста.

Рассмотрим функциональную схему осуществления этой удаленной атаки (рис 16.8):

- передача на атакуемый хост ложного ICMP Redirect Host сообщения;
- отправление ARP-ответа в случае, если пришел ARP-запрос от атакуемого хоста;
- перенаправление пакетов от атакуемого хоста на настоящий маршрутизатор;
- перенаправление пакетов от маршрутизатора на атакуемый хост;
- при приеме пакета возможно воздействие на информацию по схеме «Ложный объект РВС».

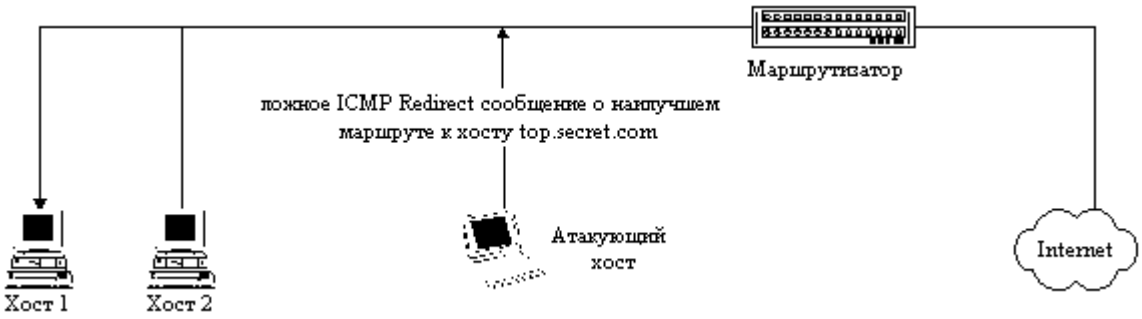


Рис16.8 а - Фаза передачи ложного ICMP Redirect сообщения от имени маршрутизатора.

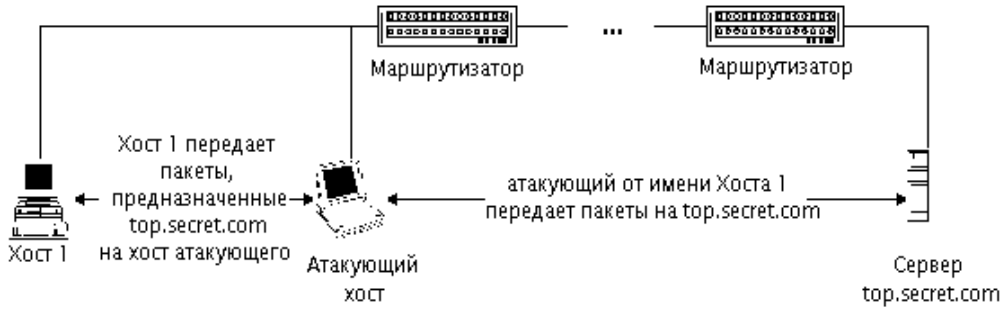


Рис16.8 б - Фаза приема, анализа, воздействия и передачи перехваченной информации на ложном сервере

Рис. 16.8 -Внутрисегментное навязывание хосту ложного маршрута при использовании протокола ICMP

В случае осуществления второго варианта удаленной атаки атакующий находится в другом сегменте относительно цели атаки. Тогда, в случае передачи на атакуемый хост ложного ICMP Redirect сообщения, сам атакующий уже не сможет получить контроль над трафиком, так как адрес нового маршрутизатора должен находиться в пределах подсети атакуемого хоста, поэтому использование данного варианта этой удаленной атаки не позволит атакующему получить доступ к передаваемой по каналу связи информации. Однако, в этом случае атака достигает другой цели: нарушается работоспособность хоста.

Атакующий с любого хоста в Internet может послать подобное сообщение на атакуемый хост и в случае, если сетевая ОС на данном хосте не проигнорирует данное сообщение, то связь между данным хостом и указанным в ложном ICMP-сообщении сервером будет нарушена. Это произойдет из-за того, что все пакеты, направляемые хостом на этот сервер, будут отправлены на IP-адрес несуществующего маршрутизатора. Схема этой атаки приведена на рис. 16.9.

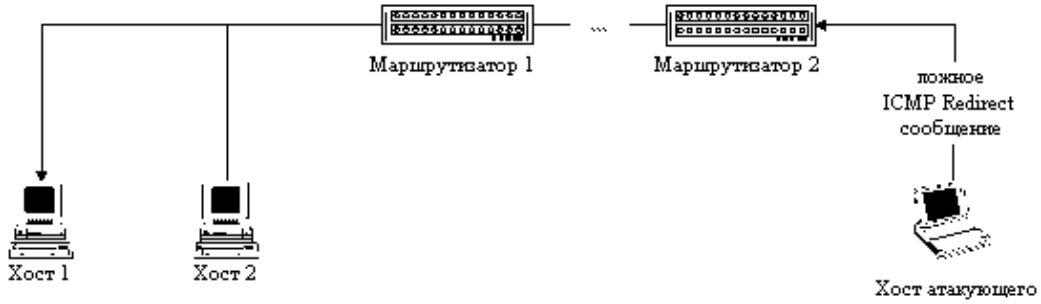


Рис. 16.9 а - Передача атакующим на хост 1 ложного ICMP Redirect сообщения от имени маршрутизатора 1

Рис. 16.9 б - Дезинформация хоста 1. Его таблица маршрутизации содержит информацию о ложном маршруте к хосту top.secret.com

Рис. 16.9 - Межсегментное навязывание хосту ложного маршрута при использовании протокола ICMP, приводящее к отказу в обслуживании

Оба варианта рассмотренной удаленной атаки удается осуществить (как межсегментно, так и внутрисегментно) на ОС Linux 1.2.8, Windows 95 и Windows NT 4.0. Остальные сетевые ОС (Linux 2.0.0 и защищенный по классу B1 UNIX), игнорировали данное ICMP Redirect сообщение (что, не

правда ли, кажется вполне логичным с точки зрения обеспечения безопасности).

16.5 Подмена одного из субъектов TCP-соединения в сети Internet

Протокол TCP (Transmission Control Protocol) является одним из базовых протоколов транспортного уровня сети Internet. Этот протокол позволяет исправлять ошибки, которые могут возникнуть в процессе передачи пакетов, и является протоколом с установлением логического соединения - виртуального канала. По этому каналу передаются и принимаются пакеты с регистрацией их последовательности, осуществляется управление потоком пакетов, организовывается повторная передача искаженных пакетов, а в конце сеанса канал разрывается. При этом протокол TCP является единственным базовым протоколом из семейства TCP/IP, имеющим дополнительную систему идентификации сообщений и соединения. Именно поэтому протоколы прикладного уровня FTP и TELNET, предоставляющие пользователям удаленный доступ на хосты Internet, реализованы на базе протокола TCP.

Для идентификации TCP-пакета в TCP-заголовке существуют два 32-разрядных идентификатора, которые также играют роль счетчика пакетов. Их названия - **Sequence Number** и **Acknowledgment Number**. Также нас будет интересовать поле, называемое **Control Bits**.

Это поле размером 6 бит может содержать следующие командные биты (слева направо):

- **URG**: Urgent Pointer field significant,
- **ACK**: Acknowledgment field significant,
- **PSH**: Push Function,
- **RST**: Reset the connection,
- **SYN**: Synchronize sequence numbers,
- **FIN**: No more data from sender.

Далее рассмотрим схему создания TCP-соединения (рис 16.10).

Предположим, что хосту **A** необходимо создать TCP-соединение с хостом **B**. Тогда **A** посылает на **B** следующее сообщение:

1. **A → B:** SYN, ISSa

Это означает, что в передаваемом **A** сообщении установлен бит SYN (*synchronize sequence number*), а в поле *Sequence Number* установлено начальное 32-битное значение *ISSa* (*Initial Sequence Number*).

2. B отвечает:

B → A: SYN, ACK, ISSb, ACK(ISSa+1)

В ответ на полученный от **A** запрос **B** отвечает сообщением, в котором установлен бит SYN и установлен бит ACK; в поле *Sequence Number* хостом

B устанавливается свое начальное значение счетчика - *ISSb*; поле *Acknowledgment Number* содержит значение *ISSa*, полученное в первом пакете от хоста **A** и увеличенное на единицу.

3. A, завершая рукопожатие (handshake), посылает:

A → B: ACK, ISSa+1, ACK(ISSb+1)

В этом пакете установлен бит *ACK*; поле *Sequence Number* содержит *ISSa + 1*; поле *Acknowledgment Number* содержит значение ISSb + 1. Посылкой этого пакета на хост **B** заканчивается трехступенчатый handshake, и TCP-соединение между хостами **A** и **B** считается установленным.

4. Теперь хост **A** может посылать пакеты с данными на хост **B** по только что созданному виртуальному TCP-каналу:

A → B: ACK, ISSa+1, ACK(ISSb+1); DATA

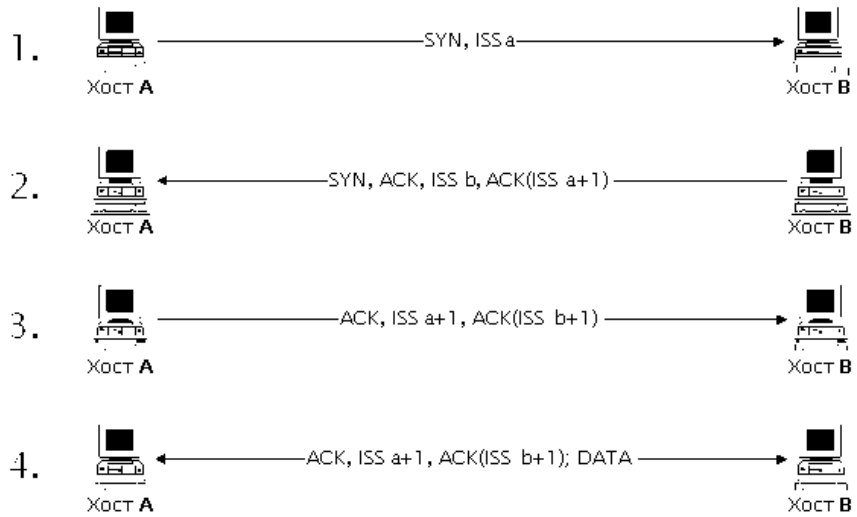


Рис. 16.10 - Схема создания TCP-соединения

Из рассмотренной выше схемы создания TCP-соединения видно, что единственными идентификаторами TCP-абонентов и TCP-соединения являются два 32-бит-ных параметра *Sequence Number* и *Acknowledgment Number*. Следовательно, для формирования ложного TCP-пакета атакующему необходимо знать текущие идентификаторы для данного соединения - *ISSa* и *ISSb*.

Проблема возможной подмены TCP-сообщения становится еще более важной, так как анализ протоколов FTP и TELNET, реализованных на базе протокола TCP, показал, что проблема идентификации FTP- и TELNET-пакетов целиком возлагается данными протоколами на транспортный уровень, то есть на TCP. Это означает, что атакующему достаточно, подобрав соответствующие текущие значения идентификаторов TCP-пакета для данного TCP-соединения (например, данное соединение может представлять собой FTP- или TELNET-подклю-чение), послать пакет с *любого хоста в*

сети Internet от имени одного из участников данного соединения (например, от имени клиента), и данный пакет будет воспринят как верный! К тому же, так как FTP и TELNET ***не проверяют*** IP-адреса отправителей, от которых им приходят сообщения, то в ответ на полученный ложный пакет, *FTP- или TELNET-сервер отправит ответ на указанный в ложном пакете настоящий IP-адрес атакующего, то есть атакующий начнет работу с FTP- или TELNET-сервером со своего IP-адреса, но с правами легально подключившегося пользователя, который, в свою очередь, потеряет связь с сервером из-за рассогласования счетчиков.*

16.6 Нарушение работоспособности хоста в сети Internet при использовании направленного «шторма» ложных TCP-запросов на создание соединения, либо при переполнении очереди запросов

Из рассмотренной в предыдущем пункте схемы создания TCP-соединения следует, что на каждый полученный TCP-запрос на создание соединения операционная система должна сгенерировать начальное значение идентификатора ISN и отослать его в ответ на запросивший хост. При этом, так как в сети Internet (стандарта IP v.4) не предусмотрен контроль за IP-адресом отправителя сообщения, то невозможно отследить истинный маршрут, пройденный IP-пакетом, и, следовательно, у конечных абонентов сети нет возможности ограничить число возможных запросов, принимаемых в единицу времени от одного хоста. Поэтому возможно осуществление типовой УА «Отказ в обслуживании», которая будет заключаться в передаче на атакуемый хост как можно большего числа ложных TCP-запросов на создание соединения от имени любого хоста в сети (рис. 16.11).

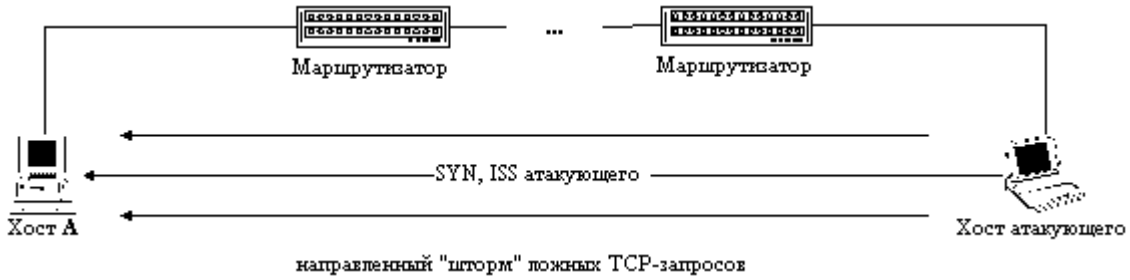


Рис. 16.11 - Нарушение работоспособности хоста в Internet, использующее направленный шторм ложных TCP-запросов на создание соединения

При этом атакуемая сетевая ОС в зависимости от вычислительной мощности компьютера либо - в худшем случае - практически зависает, либо - в лучшем случае - перестает реагировать на легальные запросы на подключение (отказ в обслуживании). Это происходит из-за того, что для всей массы полученных ложных запросов система должна, во-первых, сохранить в памяти полученную в каждом запросе информацию и, во-вторых, выработать и отослать ответ на каждый запрос. Таким образом, все

ресурсы системы «съедаются» ложными запросами: переполняется очередь запросов и система занимается только их обработкой. Эффективность данной удаленной атаки тем выше, чем больше пропускная способность канала между атакующим и целью атаки, и тем меньше, чем больше вычислительная мощь атакуемого компьютера (число и быстродействие процессоров, объем ОЗУ и т. д.).

Другая разновидность атаки «Отказ в обслуживании» состоит в передаче на атакуемый хост нескольких десятков (сотен) запросов на подключение к серверу, что может привести к временному (до 10 минут) переполнению очереди запросов на сервере. Это происходит из-за того, что некоторые сетевые ОС устроены так, чтобы обрабатывать только первые несколько запросов на подключение, а остальные - игнорировать. То есть при получении N запросов на подключение, ОС сервера ставит их в очередь и генерирует соответственно N ответов. Далее, в течение определенного промежутка времени, сервер будет дожидаться от предполагаемого клиента сообщения, завершающего handshake и подтверждающего создание виртуального канала с сервером. Если атакующий пришлет на сервер количество запросов на подключение, равное максимальному числу одновременно обрабатываемых запросов на сервере, то в течение тайм-аута остальные запросы на подключение будут игнорироваться и к серверу будет невозможно подключиться.

Необходимо отметить, что в существующем стандарте сети Internet IP v4 нет приемлемых способов надежно обезопасить свои системы от этой удаленной атаки. К счастью, атакующий в результате осуществления описанной атаки не сможет получить несанкционированный доступ к вашей информации. Он сможет лишь «съесть» вычислительные ресурсы вашей системы и нарушить ее связь с внешним миром. Остается надеяться, что нарушение работоспособности вашего хоста просто никому не нужно.

17. ОБЕСПЕЧЕНИЕ БЕЗОПАСНОСТИ СИСТЕМ ВХОДЯЩИХ В СОСТАВ ГЛОБАЛЬНЫХ КОМПЬЮТЕРНЫХ СЕТЕЙ

17.1 Межсетевые экраны (firewall)

Межсетевой экран (firewall) *- это устройство контроля доступа в сеть, предназначенное для блокировки всего трафика, за исключением разрешенных данных.* Прохождение трафика на межсетевом экране можно настраивать по службам, IP-адресам отправителя и получателя, по идентификаторам пользователей, запрашивающих службу. Межсетевые экраны позволяют осуществлять централизованное управление безопасностью. В одной конфигурации администратор может настроить разрешенный входящий трафик для всех внутренних систем организации. Этим оно отличается от маршрутизатора, функцией которого является доставка трафика в пункт назначения в максимально короткие сроки.

Межсетевые экраны, представляют собой *программные пакеты, базирующиеся на операционных системах общего назначения (таких как Windows NT и Unix) или на аппаратной платформе межсетевых экранов. Набор правил политики определяет, каким образом трафик передается из одной сети в другую. Если в правиле отсутствует явное разрешение на пропуск трафика, межсетевой экран отклоняет или аннулирует пакеты.* Правила политики безопасности усиливаются посредством использования модулей доступа.

Типы межсетевых экранов:

- *межсетевые экраны прикладного уровня;*
- *межсетевые экраны с пакетной фильтрацией;*
- *гибридные межсетевые экраны.*

17.1.1 Межсетевые экраны прикладного уровня

В межсетевом экране прикладного уровня каждому разрешаемому протоколу должен соответствовать свой собственный модуль доступа. Лучшими модулями доступа считаются те, которые построены специально для разрешаемого протокола.

Например, модуль доступа FTP предназначен для протокола FTP и может определять, соответствует ли проходящий трафик этому протоколу и разрешен ли этот трафик правилами политики безопасности. При использовании межсетевого экрана прикладного уровня все соединения проходят через него (см. рис. 17.1).

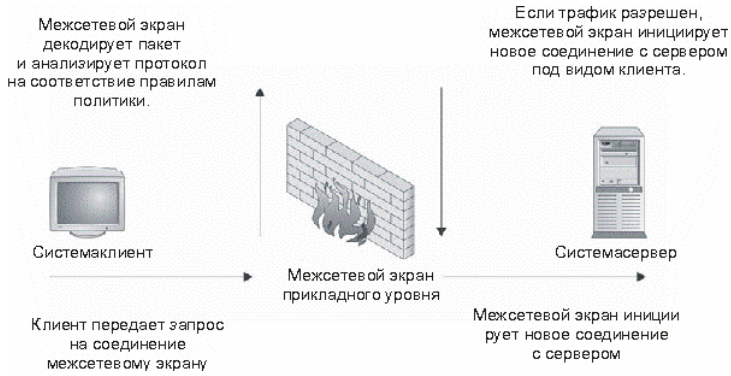


Рис. 17.1 - Соединения модуля доступа межсетевого экрана прикладного уровня

Как показано на рисунке, соединение начинается на системе-клиенте и поступает на внутренний интерфейс межсетевого экрана. Межсетевой экран принимает соединение, анализирует содержимое пакета и используемый протокол и определяет, соответствует ли данный трафик правилам политики безопасности. Если это так, то межсетевой экран инициирует новое соединение между своим внешним интерфейсом и системой-сервером.

Межсетевые экраны прикладного уровня используют модули доступа для входящих подключений. Модуль доступа в межсетевом экране принимает входящее подключение и обрабатывает команды перед отправкой трафика получателю. Таким образом, межсетевой экран защищает системы от атак, выполняемых посредством приложений.

17.1.2 Межсетевые экраны с пакетной фильтрацией

*Правила политики в **межсетевых экранах с пакетной фильтрацией** устанавливаются посредством использования фильтров пакетов. Фильтры изучают пакеты и определяют, является ли трафик разрешенным, согласно правилам политики и состоянию протокола (проверка с учетом состояния). Если протокол приложения функционирует через TCP, определить состояние относительно просто, так как TCP сам по себе поддерживает состояния.* Это означает, что когда протокол находится в определенном состоянии, *разрешена передача только определенных пакетов.*

При использовании межсетевого экрана с пакетной фильтрацией соединения не прерываются на межсетевом экране (см. рис. 16.2), а направляются непосредственно к конечной системе. При поступлении пакетов межсетевой экран выясняет, разрешен ли данный пакет и состояние соединения правилами политики. Если это так, пакет передается по своему маршруту. В противном случае пакет отклоняется или аннулируется.

Рис. 17.2 - Передача трафика через межсетевой экран с фильтрацией пакетов

Межсетевые экраны с фильтрацией пакетов не используют модули доступа для каждого протокола и поэтому могут использоваться с любым протоколом, работающим через IP. Некоторые протоколы требуют распознавания межсетевым экраном выполняемых ими действий. Например, FTP будет использовать одно соединение для начального входа и команд, а другое - для передачи файлов. Соединения, используемые для передачи файлов, устанавливаются как часть соединения FTP, и поэтому межсетевой экран должен уметь считывать трафик и определять порты, которые будут использоваться новым соединением. Если межсетевой экран не поддерживает эту функцию, передача файлов невозможна.

17.1.3 Гибридные межсетевые экраны

Как и многие другие устройства, межсетевые экраны изменяются и совершенствуются с течением времени, т. е. эволюционируют. Так технология модуля доступа Generic Services Proxy (GSP) разработана для поддержки модулями доступа прикладного уровня других протоколов, необходимых системе безопасности и при работе сетевых администраторов. В действительности GSP обеспечивает работу межсетевых экранов прикладного уровня в качестве экранов с пакетной фильтрацией.

Производители межсетевых экранов с пакетной фильтрацией также добавили некоторые модули доступа в свои продукты для обеспечения более высокого уровня безопасности некоторых широко распространенных протоколов. В то время как базовая функциональность межсетевых экранов обоих типов осталась прежней, (что является причиной большинства «слабых мест» этих устройств), сегодня на рынке присутствуют гибридные межсетевые экраны. Практически невозможно найти межсетевой экран, функционирование которого построено исключительно на прикладном уровне или фильтрации пакетов. Это позволяет администраторам, отвечающим за безопасность, настраивать устройство для работы в конкретных условиях.

17.1.4 Пример конфигурирования межсетевого экрана

Стандартная архитектура использования **межсетевого экрана** показана на рис. 17.3. В данной архитектуре используется один межсетевой экран для защиты внутренней сети, так и любых других систем, доступных из интернета. Эти системы располагаются в отдельной сети.

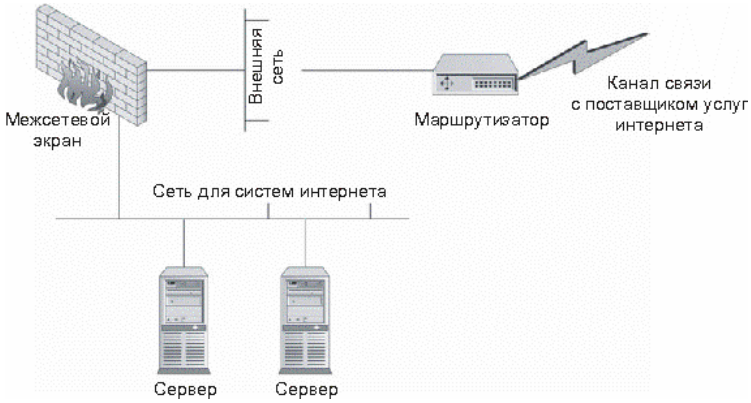


Рис. 17.3 - Использование межсетевого экрана

Таблица 17.1. Правила межсетевого экрана для архитектуры на рис. 17.3

Номер	Исходный IP	Конечный IP	Служба	Действие
1	Любой	Веб-сервер	HTTP	Принятие
2	Любой	Почтовый сервер	SMTP	Принятие
3	Почтовый сервер	Любой	SMTP	Принятие
4	Внутренняя сеть	Любой	HTTP, HTTPS, FTP, Telnet	Принятие
5	Внутренняя DNS	Любой	DNS	Принятие
6	Любой	Любой	Любая	Сброс

Таблица 17.2. Краткий список номеров портов (для протокола TCP)

Протокол : номер порта	Протокол : номер порта
HTTP: 80, 8080	FTP: 21 для команд, 20 для данных
SSH: 22	TFTP: 69/UDP
POP3: 110	ICQ: 5190
SMTP: 25	telnet: 23
IMAP: 143	DNS: 53 (обычно UDP)

17.2 Организация и эксплуатация виртуальных частных сетей (VPN)

Частные сети состоят из каналов связи, арендуемых у различных телефонных компаний и поставщиков услуг интернета. Эти каналы связи характеризуются тем, что они соединяют только два объекта, будучи отделенными от другого трафика, так как арендуемые каналы обеспечивают двустороннюю связь между двумя сайтами.

Частные сети обладают множеством преимуществ:

- *Информация сохраняется в секрете.*
- *Удаленные сайты могут осуществлять обмен информацией незамедлительно.*
- *Удаленные пользователи не ощущают себя изолированными от системы, к которой они осуществляют доступ.*

К сожалению, этот тип сетей обладает одним большим недостатком - высокой стоимостью. С увеличением числа пользователей интернета многие организации перешли на использование виртуальных частных сетей (Virtual Private Network -VPN). Виртуальные частные сети обеспечивают многие преимущества частных сетей за меньшую цену.

17.2.1 Определение виртуальных частных сетей

Значительная часть Internet трафика передается в открытом виде, и любой пользователь, наблюдающий за этим трафиком, сможет его распознать. Это относится к большей части почтового и веб-трафика, а также сеансам связи через протоколы telnet и FTP. Трафик Secure Shell (SSH) и Hyper text Transfer Protocol Secure (HTTPS) является шифруемым трафиком, и его не сможет просмотреть пользователь, отслеживающий пакеты. Тем не менее, трафик типа SSH и HTTPS не образует виртуальную частную сеть VPN.

Виртуальные частные сети обладают следующими характеристиками:

- *Трафик шифруется для обеспечения защиты от прослушивания.*
- *Осуществляется аутентификация удаленного сайта.*
- *Виртуальные частные сети обеспечивают поддержку множества протоколов.*
- *Соединение обеспечивает связь только между двумя конкретными абонентами.*

Так как SSH и HTTPS не способны поддерживать несколько протоколов, то же самое относится и к реальным виртуальным частным сетям. VPN-пакеты смешиваются с потоком обычного трафика в интернете и существуют отдельно по той причине, что данный трафик может считываться только конечными точками соединения.

VPN соединяет два конкретных объекта, образуя таким образом уникальный канал связи между двумя абонентами. Каждая из конечных точек VPN может единовременно поддерживать несколько соединений VPN с другими конечными точками, однако каждая из точек является отдельной от других, и трафик разделяется посредством шифрования.

Виртуальные частные сети, по методу использования, подразделяются на два типа:

- *пользовательские VPN;*
- *узловые VPN.*

17.2.2 Пользовательские VPN

Пользовательские VPN *представляют собой виртуальные частные сети, построенные между отдельной пользовательской системой и узлом или сетью организации* (Часто пользовательские VPN используются сотрудниками, находящимися в командировке или работающими из дома).

Сервер VPN может являться межсетевым экраном организации либо быть отдельным VPN-сервером. Пользователь подключается к интернету через телефонное подключение к локальному поставщику услуг, через канал DSL или кабельный модем и инициирует VPN-соединение с узлом организации через интернет. Узел организации запрашивает у пользователя аутентификационные данные и, в случае успешной аутентификации, позволяет пользователю осуществить доступ ко внутренней сети организации, как если бы пользователь находился внутри узла и физически располагался внутри сети.

Пользовательские VPN позволяют организациям ограничивать доступ удаленных пользователей к системам или файлам. Это ограничение должно базироваться на политике организации и зависит от возможностей продукта VPN.

Рис. 17.4 - Конфигурация пользовательской VPN

Пользовательские VPN обладают двумя основными преимуществами:

1. Сотрудники, находящиеся в командировке, могут осуществлять доступ к электронной почте, файлам и внутренним системам в

любое время без необходимости в осуществлении дорогостоящих междугородних и международных телефонных вызовов для соединения с серверами.

2. Сотрудники, работающие из дома, могут осуществлять доступ к службам сети, как и сотрудники, работающие в организации, без аренды дорогостоящих выделенных каналов.

Оба эти преимущества можно приписать к экономии денежных средств. Экономия может заключаться в отказе от использования дорогостоящих междугородних и международных соединений, арендуемых каналов связи и т.д.

Самой большой проблемой безопасности при использовании VPN сотрудником является одновременное соединение с другими сайтами интернета. Как правило, программное обеспечение VPN на компьютере пользователя определяет, должен ли трафик передаваться через VPN, либо его необходимо отправить на какой-либо другой сайт в открытом виде. Если на компьютер пользователя была произведена атака с использованием «троянского коня», возможно, что некий внешний нелегальный пользователь использует компьютер сотрудника для подключения к внутренней сети организации. Атаки данного типа осуществляются довольно сложно, но они совершенно реальны.

17.2.3 Узловые VPN

Узловые виртуальные частные сети используются организациями для подключения к удаленным узлам без применения дорогостоящих выделенных каналов или для соединения двух различных организаций, между которыми необходима связь для осуществления информационного обмена, связанного с деятельностью этих организаций. Как правило, VPN соединяет один межсетевой экран или пограничный маршрутизатор с другим аналогичным устройством (см. рис. 17.5).

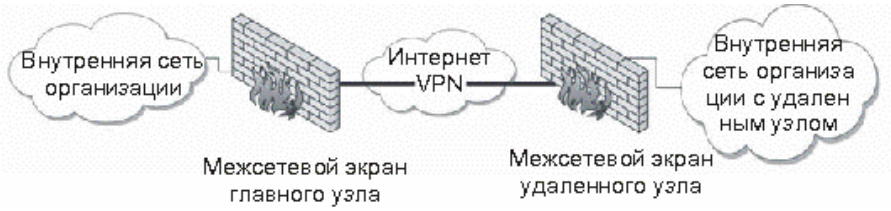


Рис. 17.5 - Межузловое соединение VPN, проходящее через интернет

Чтобы инициировать соединение, один из узлов осуществляет попытку передать трафик другому узлу. Вследствие этого на обоих противоположных узлах соединения VPN инициируется VPN. Оба конечных узла определяют параметры соединения в зависимости от политик, имеющихся на узлах. Оба сайта будут аутентифицировать друг друга посредством некоторого общего предопределенного секрета либо с помощью сертификата с открытым

ключом. Некоторые организации используют узловые VPN в качестве резервных каналов связи для арендуемых каналов.

Основным преимуществом узловой VPN является экономичность. Организация с небольшими, удаленными друг от друга офисами может создать виртуальную частную сеть, соединяющую все удаленные офисы с центральным узлом (или даже друг с другом) со значительно меньшими затратами. Сетевая инфраструктура также может быть применена значительно быстрее, так как в удаленных офисах могут использоваться локальные ISP для каналов ISDN или DSL.

Проблемы, связанные с узловыми VPN. Узловые VPN расширяют периметр безопасности организации, добавляя новые удаленные узлы или даже удаленные организации. Если уровень безопасности удаленного узла невелик, VPN может позволить злоумышленнику получить доступ к центральному узлу и другим частям внутренней сети организации. Следовательно, необходимо применять строгие политики и реализовывать функции аудита для обеспечения безопасности организации в целом. В случаях, когда две организации используют узловую VPN для соединения своих сетей, очень важную роль играют политики безопасности, установленные по обе стороны соединения. В данной ситуации обе организации должны определить, какие данные могут передаваться через VPN, а какие - нет, и соответствующим образом настроить политики на своих межсетевых экранах.

17.2.4 Понятие стандартных технологий функционирования VPN

Сеть VPN состоит из четырех ключевых компонентов/

1. Сервер VPN. Сервер VPN представляет собой компьютер, выступающий в роли конечного узла соединения VPN. Данный сервер должен обладать характеристиками, достаточными для поддержки ожидаемой нагрузки. Большая часть производителей программного обеспечения VPN должна предоставлять рекомендации по поводу производительности процессора и конфигурации памяти, в зависимости от числа единовременных VPN-соединений.

2. Алгоритмы шифрования. Выбор алгоритма не имеет принципиального значения, если он будет стандартным и в достаточной степени мощным. Гораздо больше влияет на общий уровень безопасности реализация системы. Неправильно реализованная система может сделать бесполезным самый мощный алгоритм шифрования. Приняв во внимание сказанное выше, давайте изучим риски, связанные с использованием VPN.

3. Система аутентификации. Система аутентификации VPN должна быть двухфакторной. Пользователи могут проходить аутентификацию с использованием того, что они знают, того, что у них есть или с помощью данных о том, кем они являются. Хорошей комбинацией средств аутентификации являются смарт-карты в паре с персональным

идентификационным номером или паролем. Производители программного обеспечения, как правило, предоставляют организациям на выбор несколько систем аутентификации. В данном перечне присутствуют ведущие производители смарт-карт.

4. Протокол VPN. Протокол VPN определяет, каким образом система VPN взаимодействует с другими системами в интернете, а также уровень защищенности трафика. Протокол VPN оказывает влияние на общий уровень безопасности системы. Причиной этому является тот факт, что протокол VPN используется для обмена ключами шифрования между двумя конечными узлами. Если этот обмен не защищен, злоумышленник может перехватить ключи и затем расшифровать трафик, сведя на нет все преимущества VPN. В настоящее время стандартным протоколом для VPN является IPSec. Этот протокол представляет собой дополнение к IP, осуществляющее инкапсуляцию и шифрование заголовка TCP и полезной информации, содержащейся в пакете. IPSec также поддерживает обмен ключами, удаленную аутентификацию сайтов и согласование алгоритмов (как алгоритма шифрования, так и хэш-функции). IPSec использует UDP-порт 500 для начального согласования, после чего используется IP-протокол 50 для всего трафика. Для правильного функционирования VPN эти протоколы должны быть разрешены.

Эти компоненты реализуют соответствие требованиям по безопасности, производительности и способности к взаимодействию. То, насколько правильно реализована архитектура VPN, зависит от правильности определения требований.

Определение требований по безопасности в VPN должно включать в себя следующие аспекты:

- *Количество времени, в течение которого необходимо обеспечивать защиту информации.*
- *Число одновременных соединений пользователей.*
- *Ожидаемые типы соединений пользователей (сотрудники, работающие из дома или находящиеся в поездке).*
- *Число соединений с удаленным сервером.*
- *Типы сетей VPN, которым понадобится соединение.*
- *Ожидаемый объем входящего и исходящего трафика на удаленных узлах.*
- *Политика безопасности, определяющая настройки безопасности.*

При разработке системы также может оказаться полезным указать дополнительные требования, связанные с местоположением сотрудников, находящихся в поездке (имеются в виду узлы в других организациях или в номерах отелей), а также типы служб, которые будут работать через VPN.

17.2.5 Типы систем VPN

На настоящее время можно выделить три типа систем на основе которых организуются VPN:

- *аппаратные системы;*
- *программные системы;*
- *веб-системы.*

Аппаратные системы VPN*, как правило, базируются на аппаратной платформе, используемой в качестве VPN-сервера. На этой платформе выполняется программное обеспечение производителя, а также, возможно, некоторое специальное программное обеспечение, предназначенное для улучшения возможностей шифрования.* В большинстве случаев для построения VPN на системе удаленного пользователя необходимо наличие соответствующего программного обеспечения.

Аппаратная система VPN имеет два преимущества.

- ***Скорость.*** Оборудование, как правило, оптимизировано для поддержки VPN, посредством чего обеспечивается преимущество в скорости по сравнению с компьютерными системами общего назначения. За счет этого достигается возможность поддержки большего числа одновременных VPN-соединений.
- ***Безопасность.*** Если аппаратная платформа специально разработана для приложения VPN, из ее системы удалены все лишние программы и процессы. За счет этого снижается степень подверженности атакам по сравнению с компьютерной системой общего назначения, в которой работают другие процессы. Это не значит, что компьютер общего назначения не может быть должным образом защищен. Как правило, использование компьютера общего назначения требует дополнительных усилий по настройке безопасности.

Программные VPN *работают на компьютерных системах общего назначения. Они могут быть установлены на выделенной для VPN системе либо совместно с другим программным обеспечением, таким как межсетевой экран.* При загрузке программного обеспечения необходимо обеспечить достаточную мощность аппаратной платформы для поддержки VPN. Так как VPN-продукт устанавливается на компьютеры, имеющиеся в организации, руководство организации должно позаботиться о соответствии компьютеров предъявляемым требованиям.

Веб-системы. Главным недостатком большинства пользовательских систем VPN является потребность в установке программного обеспечения на систему-клиент. Указанные проблемы привели к тому, что *некоторые производители VPN стали рассматривать веб-браузеры в качестве VPN-*

клиентов и реализовывать этот подход на практике. Он заключается в том, что пользователь с помощью браузера подключается к VPN через SSL. SSL обеспечивает шифрование трафика, а подтверждение подлинности пользователя выполняется с помощью средств аутентификации, встроенных в систему. Для предоставления пользователю необходимых услуг используется несколько различных механизмов. Среди них можно выделить надстройки браузера и виртуальные машины Java.

В то время как стоимость поддержки и обслуживания несомненно ниже, на момент написания этой книги ни одна из бесклиентных систем VPN не обеспечивает полную функциональность. Этим сетям VPN присущи ограничения, заключающиеся в наборе используемых приложений и методе подключения пользователей к внутренним системам.

17.3 Системы предотвращения вторжений (IDS)

17.3.1 Общие понятия о функционировании IDS

Обнаружение вторжений *- это еще одна задача, выполняемая сотрудниками, ответственными за безопасность информации в организации, при обеспечении защиты от атак, это активный процесс, при котором происходит обнаружение хакера при его попытках проникнуть в систему.*

Системы обнаружения вторжений (Intrusion detection system - IDS) *обнаруживают несанкционированные попытки проникновения в зашищаемый периметр.* Обнаружение вторжений помогает при превентивной идентификации активных угроз посредством оповещений и предупреждений о том, что злоумышленник осуществляет сбор информации, необходимой для проведения атаки.

Базовая концепция системы обнаружения вторжений заключается в необходимости определения периметра защиты компьютерной системы или сети.

Периметр защиты сети *представляет собой виртуальный периметр, внутри которого находятся компьютерные системы. Этот периметр может определяться межсетевыми экранами, точками разделения соединений или настольными компьютерами с модемами.* Данный периметр может быть расширен для содержания домашних компьютеров сотрудников, которым разрешено соединяться друг с другом, или партнеров по бизнесу, которым разрешено подключаться к сети. С появлением в деловом взаимодействии беспроводных сетей периметр защиты организации расширяется до размера беспроводной сети. В случае если компания имеет части информационных ресурсов доступных напрямую из глобальной сети периметр защиты дополняется димилитаризированной зоной (DMZ). Суть DMZ заключается в том, что она не входит непосредственно ни во внутреннюю, ни во внешнюю сеть, и доступ к ней

может осуществляться только по заранее заданным правилам межсетевого экрана. В DMZ нет пользователей — там располагаются только серверы.

Демилитаризованная зона (Demilitarized Zone — DMZ) *служит для предотвращения доступа из внешней сети к ресурсам и компьютерам внутренней сети за счет выноса из локальной сети в особую зону всех сервисов, требующих доступа извне.*

Сигнализация, оповещающая о проникновении злоумышленика, предназначена для обнаружения любых попыток входа в защищаемую область т. е. система обнаружения вторжений IDS предназначена для разграничения авторизованного входа и несанкционированного проникновения.

Цели использования IDS определяют требования для политики IDS. Потенциально целями применения IDS являются следующие:

- ***Обнаружение атак.*** Распознавание атак является одной из главных целей использования IDS. Система IDS запрограммирована на поиск определенных типов событий, которые служат признаками атак. В качестве простого примера приведем соединение через TCP-порт 80 (HTTP), за которым следует URL, содержащий расширение .bat. Это может быть признаком того, что злоумышленник пытается использовать уязвимость на веб-сервере IIS.
- ***Предотвращение атак.*** При обнаружении атаки IDS должна выполнить действия по нейтрализации угрозы.
- ***Обнаружение нарушений политики.*** Целью системы IDS, настроенной на отслеживание политики, является отслеживание выполнения или невыполнения политики организации. В самом простом случае NIDS можно настроить на отслеживание всего веб-трафика вне сети. Такая конфигурация позволяет отслеживать любое несоответствие политикам использования Интернета.
- ***Принуждение к использованию политик безопасности.*** Применение системы IDS в качестве средства принудительного использования политики выводит конфигурацию мониторинга политики на более высокий уровень. При отслеживании политики IDS настраивается на выполнение действий при нарушении политики.
- ***Принуждение к следованию политикам соединений.*** Использования принудительного блокирования незапрошенных или запрещенных соединений.
- ***Сбор доказательств.*** Система IDS может оказаться полезной после обнаружения инцидента. В этом случае с помощью IDS можно собрать доказательства. Сетевую IDS можно настроить на отслеживание определенных соединений и ведение полноценного журнала по учету трафика.

Существуют два основных типа IDS:

- ***узловые (Host IDS HIDS)*** - располагается на отдельном узле и отслеживает признаки атак на данный узел.
- ***сетевые (Network IDS - NIDS)*** - находится на отдельной системе, отслеживающей сетевой трафик на наличие признаков атак, проводимых в подконтрольном сегменте сети.

На рисунке 17.6 показаны два типа IDS, которые могут присутствовать в сетевой среде.

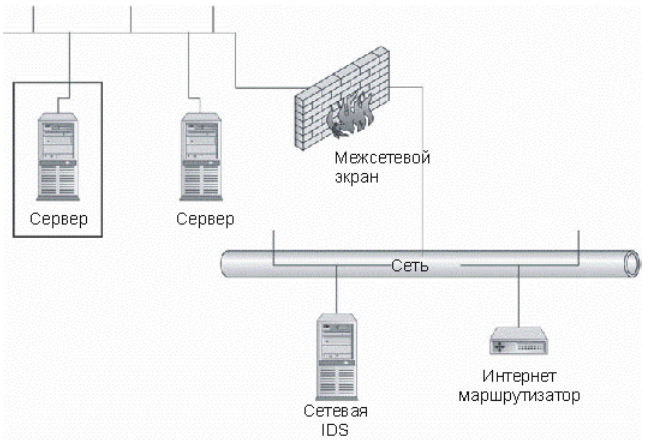


Рис. 17.6 - Примеры размещения IDS в сетевой среде

17.3.2 Узловые IDS

Узловые IDS (Host intrusion detection system - HIDS) *представляют собой систему датчиков, загружаемых на различные сервера организации и управляемых центральным диспетчером.* Датчики отслеживают различные типы событий (более детальное рассмотрение этих событий приводится в следующем разделе) и предпринимают определенные действия на сервере либо передают уведомления. Датчики HIDS отслеживают события, связанные с сервером, на котором они загружены. Сенсор HIDS позволяет определить, была ли атака успешной, если атака имела место на той же платформе, на которой установлен датчик.

Существует пять основных типов датчиков HIDS.

1. Анализаторы журналов. Большая часть анализаторов журналов настроена на отслеживание записей журналов, которые могут означать событие, связанное с безопасностью системы. Анализаторы журналов по своей природе являются реактивными системами. Иными словами, они реагируют на событие уже после того, как оно произошло. Таким образом, журнал будет содержать сведения о том, что проникновение в систему выполнено. В большинстве случаев анализаторы журналов не способны предотвратить осуществляемую атаку на систему.

2. Датчики признаков. Системы, основанные на сопоставлении признаков, обеспечивают возможность отслеживания атак во время их

выполнения в системе, поэтому они могут выдавать дополнительные уведомления о проведении злоумышленных действий. Тем не менее, атака будет успешно или безуспешно завершена перед вступлением в действие датчика HIDS, поэтому датчики этого типа считаются реактивными. Датчик признаков HIDS является полезным при отслеживании авторизованных пользователей внутри информационных систем.

3. Анализаторы системных вызовов. Анализаторы системных вызовов осуществляют анализ вызовов между приложениями и операционной системой для идентификации событий, связанных с безопасностью. Датчики HIDS данного типа размещают программную спайку между операционной системой и приложениями. Когда приложению требуется выполнить действие, его вызов операционной системы анализируется и сопоставляется с базой данных признаков. Эти признаки являются примерами различных типов поведения, которые являют собой атакующие действия, или объектом интереса для администратора IDS. Анализаторы системных вызовов отличаются от анализаторов журналов и датчиков признаков HIDS тем, что они могут предотвращать действия. Если приложение генерирует вызов, соответствующий, например, признаку атаки на переполнение буфера, датчик позволяет предотвратить этот вызов и сохранить систему в безопасности.

4. Анализаторы поведения приложений. Анализаторы поведения приложений аналогичны анализаторам системных вызовов в том, что они применяются в виде программной спайки между приложениями и операционной системой. В анализаторах поведения датчик проверяет вызов на предмет того, разрешено ли приложению выполнять данное действие, вместо определения соответствие вызова признакам атак. Например, веб-серверу обычно разрешается принимать сетевые соединения через порт 80, считывать файлы в веб-каталоге и передавать эти файлы по соединениям через порт 80. Если веб-сервер попытается записать или считать файлы из другого места или открыть новые сетевые соединения, датчик обнаружит несоответствующее норме поведение сервера и заблокирует действие.

5. Контролеры целостности файлов. Контролеры целостности файлов отслеживают изменения в файлах. Это осуществляется посредством использования криптографической контрольной суммы или цифровой подписи файла. Конечная цифровая подпись файла будет изменена, если произойдет изменение хотя бы малой части исходного файла (это могут быть атрибуты файла, такие как время и дата создания). Алгоритмы, используемые для выполнения этого процесса, разрабатывались с целью максимального снижения возможности для внесения изменений в файл с сохранением прежней подписи.

17.3.3 Сетевые IDS

Сетевые IDS (Network intrusion detection system - NIDS) представляет собой *программный процесс, работающий на специально выделенной системе. NIDS переключает сетевую карту в режим работы, при котором сетевой адаптер пропускает весь сетевой трафик (а не только трафик, направленный на данную систему) в программное обеспечение NIDS. После этого происходит анализ трафика с использованием набора правил и признаков атак для определения того, представляет ли этот трафик какой-либо интерес. Если это так, то генерируется соответствующее событие.*

На данный момент большинство систем NIDS базируется на признаках атак. Это означает, что в системы встроен набор признаков атак, с которыми сопоставляется трафик в канале связи. Если происходит атака, признак которой отсутствует в системе обнаружения вторжений, система NIDS не замечает эту атаку.

NIDS-системы позволяют указывать интересуемый трафик по адресу источника, конечному адресу, порту источника или конечному порту. Это дает возможность отслеживания трафика, не соответствующего признакам атак.

Рис. 17.7 - Конфигурация NIDS с двумя сетевыми картами

Среди преимуществ использования NIDS можно выделить следующие моменты.

- *NIDS можно полностью скрыть в сети таким образом, что злоумышленник не будет знать о том, что за ним ведется наблюдение.*
- *Одна система NIDS может использоваться для мониторинга трафика с большим числом потенциальных систем-целей.*
- *NIDS может осуществлять перехват содержимого всех пакетов, направляющихся на систему-цель.*

Среди недостатков NIDS необходимо отметить следующие аспекты.

- *Система NIDS может только выдавать сигнал тревоги, если трафик соответствует предустановленным правилам или признакам.*
- *NIDS может упустить нужный интересуемый трафик из-за использования широкой полосы пропускания или альтернативных маршрутов.*
- *Система NIDS не может определить, была ли атака успешной.*
- *Система NIDS не может просматривать зашифрованный трафик.*
- *В коммутируемых сетях (в отличие от сетей с общими носителями) требуются специальные конфигурации, без которых NIDS будет проверять не весь трафик.*

17.3.4 Использование IDS

Пример использования IDS приведен на рис. 17.8.

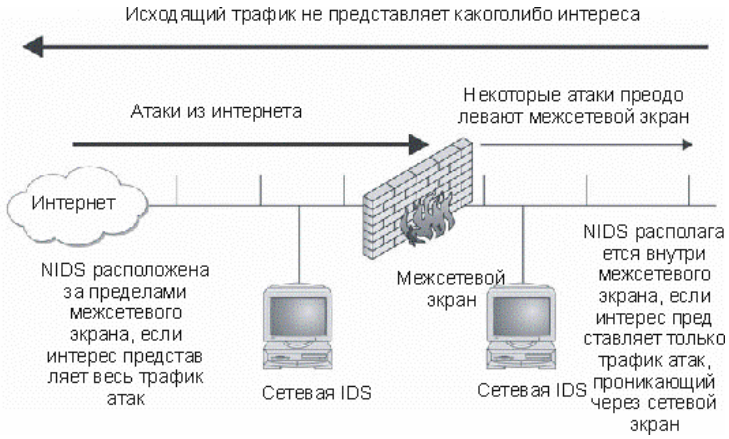


Рис. 17.8 - Пример выбора объекта мониторинга

Таблица 17.3 – События, отслеживаемые при наличии политики IDS

Политика	NIDS	HIDS
Обнаружение атак	Весь трафик, поступающий на потенциально атакуемые системы (сетевые экраны, веб-серверы, серверы приложений и т.д.)	Неудачные попытки входа. Попытки соединения. Удачный вход с удаленных систем.
Предотвращение атак	То же, что и для обнаружения атак	То же, что и для обнаружения атак.
Обнаружение нарушений политики	Весь трафик HTTP, формируемый на системах клиентах. Весь трафик FTP, формируемый на системах клиентах	Успешные HTTP-соединения. Успешные FTP соединения. Загружаемые файлы.

Политика	NIDS	HIDS
Принуждение к использованию политик	То же, что и для обнаружения нарушений политики	То же, что и для обнаружения нарушения политики.
Принуждение к соответствию политикам соединений	Весь трафик, нарушающий принудительно используемую политику соединения	Успешные соединения с запрещенных адресов или по запрещенным портам.
Сбор доказательств	Содержимое всего трафика, формируемого на системе-цели или атакующей системе	Все успешные подключения, исходящие с атакующей системы. Все неудачные соединения с атакующих систем. Все нажатия клавиш из интерактивных сеансов на атакующих системах.

При обнаружении вторжения IDS должна выработать методы противодействия вторжению.

Рассматривают следующие виды действий при обнаружении вторжений:

- ***Пассивная обработка*** - это наиболее распространенный тип действий, предпринимаемых при обнаружении вторжения. Причина этому проста - пассивные ответные действия обеспечивают меньшую вероятность повреждения легитимного трафика, являясь, в то же время, наиболее простыми для автоматического применения. Как правило, пассивные ответные действия осуществляют сбор большего числа информации или передают уведомления лицам, имеющим право на принятие более жестких мер.
- ***Активная обработка события*** *позволяет наиболее быстро предпринять возможные меры для снижения уровня вредоносного действия события.* Однако если недостаточно серьезно отнестись к логическому программированию действий в различных ситуациях и не провести должного тестирования набора правил, активная обработка событий может вызвать повреждение системы или полный отказ в обслуживании легитимных пользователей. Среди активной обработки событий различают следующие.
- ***Прерывание соединений, сеансов или процессов.*** Вероятно, самым простым действием для понимания является прерывание события. Оно может осуществляться посредством прерывания соединения, используемого атакующим злоумышленником (это возможно только в том случае, если событие использует TCP-соединение), с закрытием сеанса пользователя или завершением процесса, вызвавшего неполадку.
- ***Определение того, какой объект подлежит уничтожению***, выполняется посредством изучения события. Если процесс

использует слишком много системных ресурсов, лучше всего завершить его. Если пользователь пытается использовать конкретную уязвимость или осуществить нелегальный доступ к файлам, то рекомендуется закрыть сеанс этого пользователя. Если злоумышленник использует сетевое соединение в попытках изучения уязвимостей системы, то следует закрыть соединение.

Таблица 17.4. - Примеры ответных действий, определяемые политикой IDS

Политика	Пассивные ответные действия	Активные ответные действия
Обнаружение атак	Ведение журналов. Ведение дополнительных журналов. Уведомление	Нет ответного активного действия.
Предотвращение атак	Ведение журналов. Уведомление.	Закрытие соединения. Завершение процесса. Возможна перенастройка маршрутизатора или межсетевого экрана.
Обнаружение нарушений политики	Ведение журналов. Уведомление.	Нет ответного активного действия.
Принудительное использование политик	Ведение журналов. Уведомление.	Закрытие соединения. Возможно перенастройка прокси.
Принудительное использование политик соединения	Ведение журналов. Уведомление.	Закрытие соединения. Возможно перенастройка маршрутизатора или межсетевого экрана.
Сбор доказательств	Ведение журналов. Ведение дополнительных журналов. Уведомление.	Обманные действия. Возможно закрытие соединения.

18. ОБЕСПЕЧЕНИЕ БЕЗОПАСТНОГО ВЗАИМОДЕЙСТВИЯ В ГЛОБАЛЬНЫХ КОМПЬЮТЕРНЫХ СЕТЯХ

18.1 Аутефекация и управление сертификатами

18.1.1 Цифровые подписи

Цифровые подписи *- это форма шифрования, обеспечивающая аутентификацию (подтверждение подлинности) цифровой информации.*

С помощью цифровой подписи можно повысить уровень этой защиты и обезопасить информацию от изменения после получения и дешифрования.

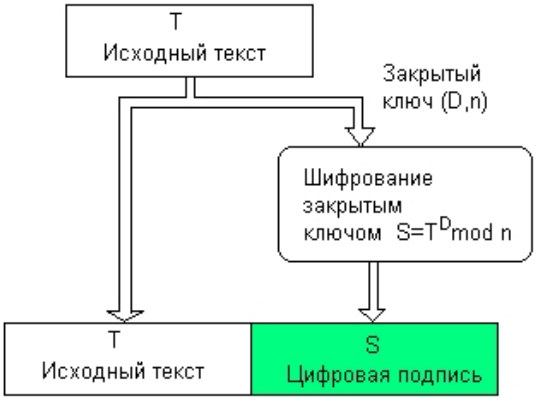


Рис. 18.1 - Схема формирования цифровой подписи по алгоритму RSA

До настоящего времени наиболее часто для построения схемы цифровой подписи использовался алгоритм RSA. В основе этого алгоритма лежит концепция Диффи-Хеллмана. Она заключается в том, что каждый пользователь сети имеет свой закрытый ключ, необходимый для формирования подписи; соответствующий этому секретному ключу открытый ключ, предназначенный для проверки подписи, известен всем другим пользователям сети.

На рис. 18.1 показана схема формирования цифровой подписи по алгоритму RSA. Подписанное сообщение состоит из двух частей: незашифрованной части, в которой содержится исходный текст *T*, и зашифрованной части, представляющей собой цифровую подпись.

Если результат расшифровки цифровой подписи совпадает с открытой частью сообщения, то считается, что документ подлинный, не претерпел никаких изменений в процессе передачи, а автором его является именно тот человек, который передал свой открытый ключ получателю. Если сообщение снабжено цифровой подписью, то получатель может быть уверен, что оно не было изменено или подделано по пути.

Цифровые подписи применяются к тексту до того, как он шифруется. Если помимо снабжения текста электронного документа цифровой подписью надо обеспечить его конфиденциальность, то вначале к тексту применяют

цифровую подпись, а затем шифруют все вместе: и текст, и цифровую подпись (рис. 18.2).

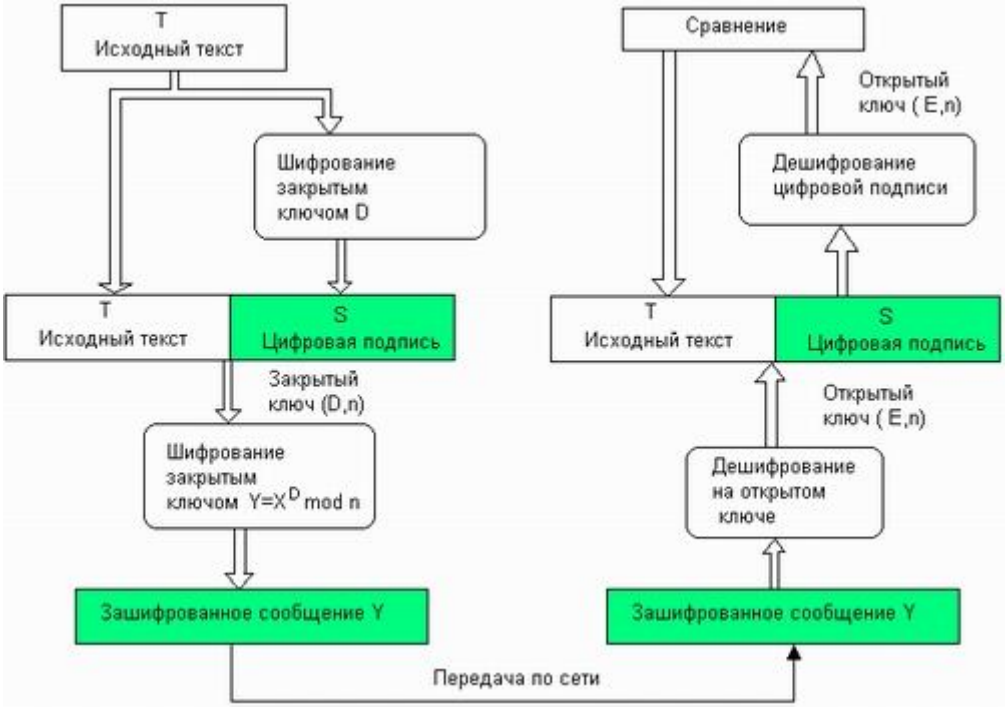


Рис. 18.2 - Обеспечение конфиденциальности документа с цифровой подписью

18.1.2 Управление ключами и сертификация ключей

Управление ключами является самой неприятной задачей при использовании любой системы шифрования. Ключи представляют собой самые важные объекты во всей системе, так как если злоумышленник получает ключ, у него появляется возможность расшифровывать все данные, зашифрованные с помощью этого ключа. Управление ключами заключается не только в защите их при использовании. Данная задача предусматривает создание надежных ключей, безопасное распространение ключей среди удаленных пользователей, обеспечение корректности ключей, отмену в случае их раскрытия или истечения срока действия.

Если ключи некоторым образом передаются в удаленное место расположения, они должны проверяться при получении на предмет того, не подверглись ли они вмешательству в процессе передачи. Это можно делать вручную либо использовать некоторую форму цифровой подписи.

Открытые ключи предназначены для публикации или передачи другим пользователям и должны сертифицироваться как принадлежащие владельцу ключевой пары. ***Сертификация*** *осуществляется с* ***помощью центрального бюро сертификатов (Certificate Authority - CA)****. В данном случае CA предоставляет цифровую подпись на открытом ключе, и благодаря этому CA с доверием воспринимает тот факт, что открытый ключ принадлежит владельцу ключевой пары* (см. рис. 18.3).

Цифровой сертификат *представляет собой цифровой документ (небольшой файл), заверяющий подлинность и статус владельца для пользователя или компьютерной системы.*

Бюро сертификатов (Certificate Authority - CA) *- объединение, включающее сервер сертификатов и создающее сертификаты.*

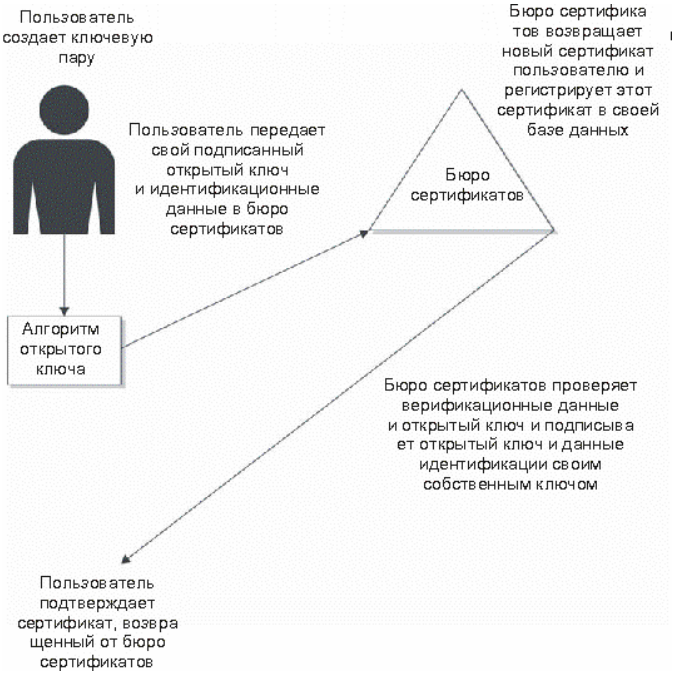


Рис. 18.3 - Сертификация открытого ключа в бюро сертификатов

18.1.3 Концепция доверия в информационной системе

Концепция доверия является основополагающим принципом информационной безопасности и шифрования в частности. Для работы шифрования необходима уверенность в том, что ключ шифра не будет раскрыт, и что используемый алгоритм шифрования является достаточно мощным. В случае с аутентификацией и цифровыми подписями необходима также уверенность в том, что открытый ключ на самом деле принадлежит тому, кто его использует.

18.1.3.1 Иерархическая модель доверия

Иерархическая модель доверия наиболее проста для восприятия. Говоря простым языком, в данном случае вы доверяете человеку, который находится выше в иерархической цепи, так как от него было получено соответствующее указание о необходимости доверия. На рисунке 18.4 изображена схема этой модели.

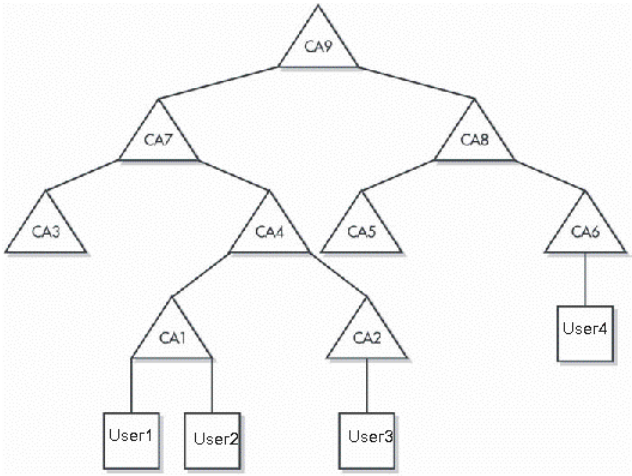


Рис. 18.4 - Иерархическая модель доверия

Как видно из рисунка, пользователи *User1* и *User2* располагаются под *CA1*. Следовательно, если *CA1* говорит, что сертификат открытого ключа принадлежит пользователю *User1*, пользователь *User2* будет верить этому. На практике *User2* передает пользователю *User1* свой сертификат открытого ключа, подписанный *CA1*. Пользователь *User1* проверяет подпись *CA1* с использованием открытого ключа *CA1*. Так как *CA1* находится в иерархии выше, чем *User1*, то *User1* доверяет *CA1* и, следовательно, доверяет сертификату пользователя *User2*.

Если пользователю *User1* нужно проверить информацию от пользователя *User3*, все несколько усложняется. *CA1* не знает о пользователе *User3*, в отличие от *CA2*. Тем не менее, пользователь *User1* не доверяет *CA2*, так как это бюро сертификатов напрямую не принадлежит цепочке пользователя *User1*. Следующий уровень вверх по цепочке - *CA4*. Пользователь *User1* может верифицировать информацию от пользователя *User3* посредством проверки с помощью *CA4* следующим образом.

- Пользователь *User1* смотрит на сертификат пользователя *User3*. Он подписан в *CA2*.
- Пользователь *User1* получает сертификат пользователя *CA2*. Он подписан в *CA4*.
- Так как пользователь *User1* доверяет *CA4*, открытый ключ *CA4* может использоваться для верификации сертификата *CA2*.
- Как только сертификат *CA2* верифицирован, пользователь *User1* может верифицировать сертификат пользователя *User3*.
- Как только будет верифицирован сертификат пользователя *User3*, пользователь *User1* может использовать открытый ключ пользователя *User3* для верификации данных.

18.1.3.2 Сетевая модель доверия

Сеть с доверием представляет собой альтернативную модель доверия. Эта концепция была впервые использована в технологии Pretty Good Privacy (PGP).

Сетевая модель доверия *заключается в том, что каждый пользователь сертифицирует свой сертификат и передает его известным ассоциированным объектам. Эти объекты могут подписать сертификат другого пользователя, так как он известен (см. рис. 18.5).*

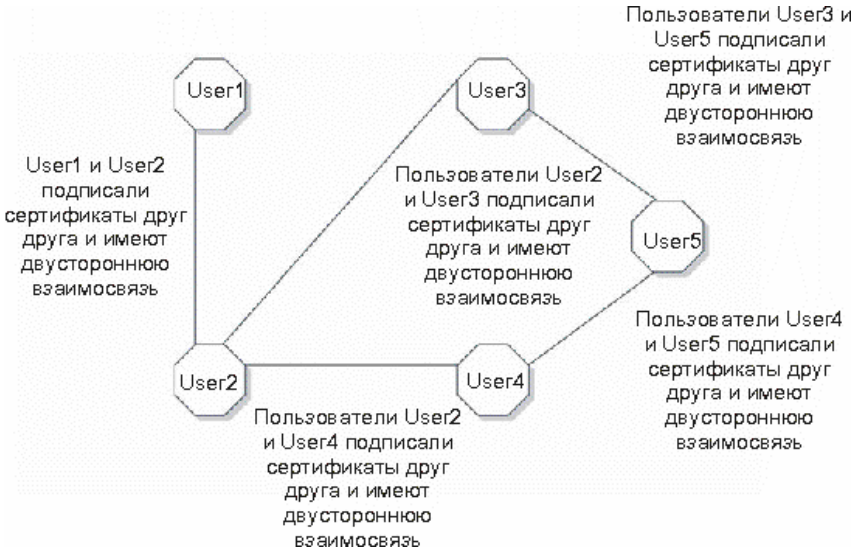


Рис. 18.5 - Сетевая модель доверия

В данной модели не существует центрального бюро сертификатов. Если пользователю *User1* требуется верифицировать информацию, поступающую от пользователя *User2*, он запрашивает сертификат пользователя *User2*. Так как пользователь *User1* знает пользователя *User2*, то доверяет сертификату и даже может его подписать.

Теперь рассмотрим ситуацию, в которой User1 получает информацию от *User3*. Пользователь *User3* не известен пользователю *User1*, но у пользователя User3 есть сертификат, подписанный пользователем *User2*. Таким образом, рассматриваемая модель распространяется на всю компьютерную сеть. Единственным решением, которое должно приниматься в процессе работы, является число переходов, которому доверяет пользователь. Как правило, это число равно 3 или 4. Кроме того, может возникнуть ситуация, в которой для установления доверия другому пользователю есть два пути. Например, *User2* может использовать два пути установления доверия с пользователем *User5*: один через пользователя *User3* и другой через пользователя User4. Так как оба пользователя *User3* и *User4* сертифицируют пользователя *User5*, пользователь *User2* может быть уверен в сертификате пользователя *User5*.

Главной проблемой, связанной с данной моделью доверия*, является недостаток масштабируемости. Так как модель сети состоит из*

двусторонних взаимоотношений, каждый пользователь должен иметь некоторое число таких взаимосвязей, чтобы пользоваться в сети каким-либо доверием. На практике такие взаимосвязи могут отсутствовать, так как большинство пользователей работают с небольшим числом связей и редко выходят на уровень трех или четырех переходов.

18.1.4 Аутентификация с использованием протоколов открытого ключа

Протоколы открытых ключей позволяют устанавливать авторизованные шифруемые связи между узлами внутренних сетей и в интернете.

Существуют три модели аутентификации, проводимой в этих протоколах; они используются как по отдельности, так и в комбинации.

- ***Аутентификация клиента***. Позволяет серверу Windows 2000 VPN или веб-серверу IIS идентифицировать пользователя с использованием стандартных методов шифрования на открытом ключе. Осуществляет проверку подлинности сертификата клиента и общего ID, а также проверку того, что эти данные сгенерированы бюро сертификатов, корневой сертификат которого установлен в перечне доверенных CA. Эта проверка очень важна, если сервером является банк, который передает конфиденциальную финансовую информацию клиенту и должен подтвердить личность получателя. На рисунке 18.6 отображен процесс аутентификации.

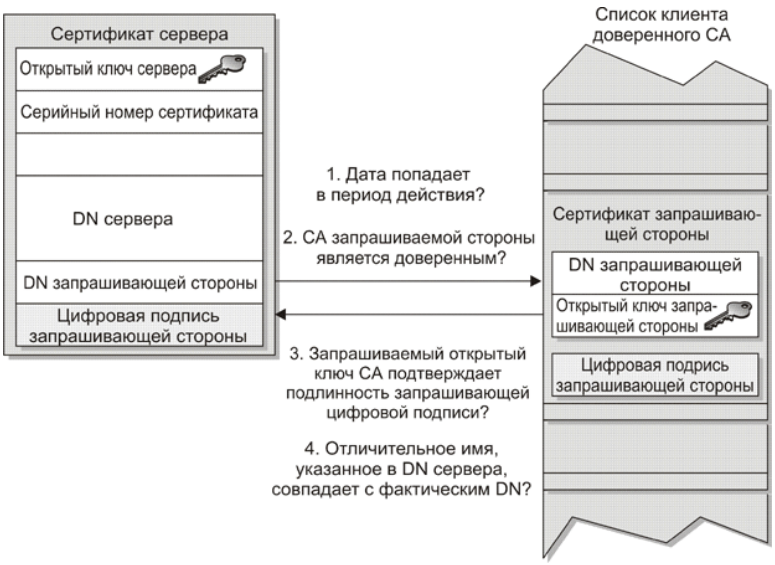


Рис. 18.6 -Аутентификация сервером сертификата клиента

- ***Аутентификация сервера***. Позволяет клиенту VPN или браузеру клиента SSL/TLS подтверждать идентичность сервера, проверяя правильность сертификата сервера и идентификатора ID, а также то, что сертификаты выпущены бюро сертификатов (CA), корневой сертификат которого присутствует в перечне доверенных CA

клиента. Это подтверждение имеет важное значение для пользователя веб-сайта, который отправляет номер кредитной карты через сеть и хочет удостовериться в том, что это именно тот сервер, который ему нужен.

- ***Взаимная аутентификация***. Позволяет клиенту и серверу авторизовать друг друга единовременно. Взаимная аутентификация требует, чтобы клиент и сервер имели цифровые сертификаты и соответствующие корневые сертификаты CA в перечнях доверенных CA.

18.2 Протокол конфиденциального обмена данными SSL

Протокол SSL спроектирован для *обеспечения конфиденциальности обмена между двумя прикладными процессами клиента и сервера* (см. http://www.netscape.com). Он предоставляет возможность аутентификации сервера и, опционно, клиента. SSL требует применения надежного транспортного протокола (например, TCP).

Преимуществом SSL является то, что он независим от прикладного протокола. Протоколы приложения, такие как HTTP, FTP, TELNET и т.д. могут работать поверх протокола SSL совершенно прозрачно. Протокол SSL может согласовывать алгоритм шифрования и ключ сессии, а также аутентифицировать сервер до того как приложение примет или передаст первый байт данных.

Все протокольные прикладные данные в SSL передаются зашифрованными с гарантией конфиденциальности.

Протокол SSL предоставляет «безопасный канал», который имеет три основные свойства.

- *Канал является частным.* Шифрование используется для всех сообщений после простого диалога, который служит для определения секретного ключа.
- *Канал аутентифицирован.* Серверная сторона диалога всегда аутентифицируется, в то время как клиентская — аутентифицируется опционно.
- Канал надежен. Транспортировка сообщений включает в себя проверку целостности (с привлечением MAC – Message Authentication Code).

Сеанс SSL между клиентом и сервером устанавливается следующим образом.

1. *Клиент открывает сокет и запрашивает подключение к серверу.*
2. *Сервер аутентифицирует клиента (либо по паролю, либо посредством сертификата, отправляемого клиентом).*

3. *После установки соединения сервер передает браузеру свой открытый ключ посредством отправки сертификата сервера, выпущенного доверенным бюро сертификатов.*
4. *Клиент аутентифицирует сертификат.*
5. *Клиент и сервер осуществляют обмен настроечной информацией для определения типа и силы шифрования, используемых в сеансе соединения.*
6. *Клиент создает сеансовый ключ, используемый для шифрования данных.*
7. *Клиент шифрует сеансовый ключ с помощью открытого ключа сервера (полученного из сертификата сервера) и отправляет его серверу. Секретный ключ, с помощью которого можно расшифровать сеансовый ключ, находится только на сервере.*
8. *Сервер расшифровывает сеансовый ключ и использует его для создания безопасной сессии, через которую будет осуществляться обмен данными с клиентом.*

В несколько упрощенном варианте диалог SSL представлен на рис. 18.7.

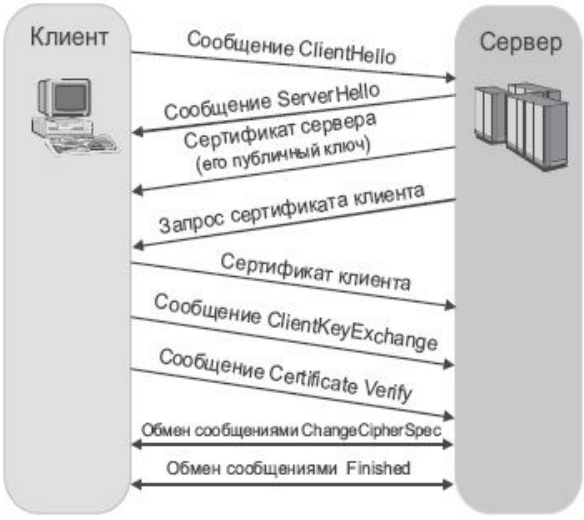


Рис. 18.7 - Алгоритм работы SSL

Необходимым условием успешной реализации этих шагов является заранее установленный на клиенте корневой сертификат, полученный от доверенного бюро сертификатов. При использовании сертификата, полученного от коммерческого CA, корневой сертификат которого уже имеется в Microsoft Internet Explorer и Netscape Communicator (например, Verisign), не нужно беспокоиться об этом. При использовании сертификатов клиентов серверу необходимо установить клиентский корневой сертификат, выпущенный клиентским бюро сертификатов.

*Протокол диалога SSL имеет две основные фазы. Первая фаза используется для установления конфиденциального канала коммуникаций. Вторая служит для аутентификации клиента (*смотри также http://book.itep.ru/6/ssl_65.htm).

Фаза 1.

Первая фаза является фазой инициализации соединения, когда оба партнера посылают сообщения **hello**. Клиент инициирует диалог посылкой сообщения **CLIENT-HELLO**. Сервер, получив это сообщение, обрабатывает его и откликается сообщением **SERVER-HELLO**.

К этому моменту, как клиент, так и сервер имеют достаточно информации, чтобы знать, нужен ли новый **мастерный ключ**. Когда новый мастерный ключ не нужен, клиент и сервер немедленно переходят в **фазу 2**.

Когда нужен новый мастерный ключ, сообщение **SERVER-HELLO** будет содержать достаточно данных, чтобы клиент мог сформировать такой ключ. Сюда входит:

- подписанный сертификат сервера,
- список базовых шифров (см. ниже),
- идентификатор соединения (последний представляет собой случайное число, сформированное сервером и используемое на протяжении сессии).

Клиент генерирует мастерный ключ и посылает сообщение **CLIENT-MASTER-KEY** (или сообщение **ERROR**, если информация сервера указывает, что клиент и сервер не могут согласовать базовый шифр).

Здесь следует заметить, что каждая оконечная точка SSL использует пару шифров для каждого соединения (т.е. всего 4 шифра). На каждой конечной точке, один шифр используется для исходящих коммуникаций и один — для входящих. Когда клиент или сервер генерирует ключ сессии, они в действительности формируют два ключа, **SERVER-READ-KEY** (известный также как **CLIENT-WRITE-KEY**) и **SERVER-WRITE-KEY** (известный также как **CLIENT-READ-KEY**). Мастерный ключ используется клиентом и сервером для генерации различных ключей сессий.

Наконец, после того как мастерный ключ определен, сервер посылает клиенту сообщение **SERVER-VERIFY**. Этот заключительный шаг аутентифицирует сервер, так как только сервер, который имеет соответствующий общедоступный ключ, может знать мастерный ключ.

Фаза 2.

Вторая фаза является фазой аутентификации. Сервер уже аутентифицирован клиентом на первой фазе, по этой причине здесь осуществляется аутентификация клиента. При типичном сценарии серверу необходимо получить что-то от клиента, и он посылает запрос. Клиент пришлет позитивный отклик, если располагает необходимой информацией, или пришлет сообщение об ошибке, если нет. Эта спецификация протокола

не определяет семантику сообщения **ERROR**, посылаемого в ответ на запрос сервера (например, конкретная реализация может игнорировать ошибку, закрыть соединение, и т.д. и, тем не менее, соответствовать данной спецификации). Когда один партнер выполнил аутентификацию другого партнера, он посылает сообщение **finished**. В случае клиента сообщение **CLIENT-FINISHED** содержит зашифрованную форму идентификатора **CONNECTION-ID**, которую должен верифицировать сервер. Если верификация терпит неудачу, сервер посылает сообщение **ERROR**.

Раз партнер послал сообщение **finished** он должен продолжить воспринимать сообщения до тех пор, пока не получит сообщение **finished** от партнера. Как только оба партнера послали и получили сообщения **finished**, протокол диалога SSL закончил свою работу. С этого момента начинает работать прикладной протокол.

18.3 Обеспечение безопасности беспроводных сетей

В беспроводных локальных сетях главным образом используется группа стандартов 802.11x (a, b, g и т. д.) технологии Wi-Fi. Эти стандарты позволяют соединять рабочие станции каналами с пропускной способностью до 54 Мбит/с с использованием беспроводной точки доступа, которая подключается к кабельной сети или напрямую к другой рабочей станции (см. рис. 18.8).

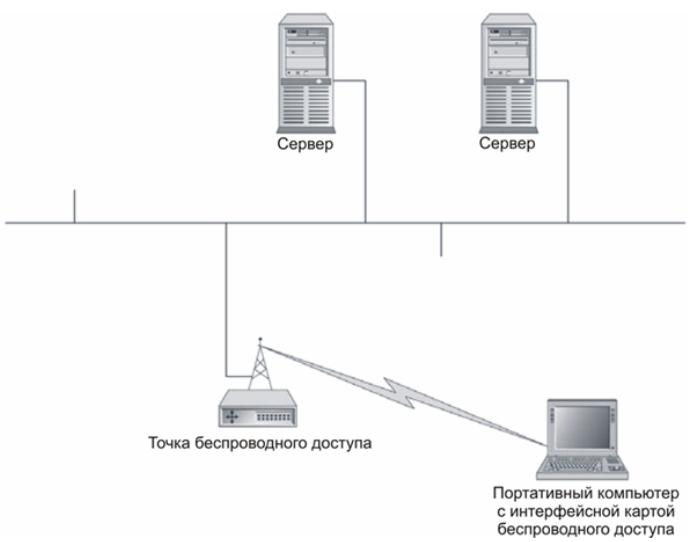


Рис. 18.8 - Типичная архитектура беспроводной сети

Так как беспроводные сети используют воздух и пространство для передачи и приема информации (сигналы являются открытыми для любого лица, находящегося в зоне действия), безопасность передачи данных является очень важным аспектом безопасности всей системы в целом. Без обеспечения должной защиты конфиденциальности и целостности информации при ее передаче между рабочими станциями и точками доступа нельзя быть уверенным в том, что информация не будет перехвачена

злоумышленником, и что рабочие станции и точки доступа не будут подменены посторонним лицом.

18.3.1 Угрозы безопасности беспроводных соединений

18.3.1.1 Обнаружение беспроводных сетей

Обнаружить WLAN очень легко. Действительно, именно для этой цели был разработан ряд средств. Одной из таких утилит является NetStumber (http://www.netstumber.com/); она работает в операционных системах семейства Windows и может использоваться совместно со спутниковым навигатором (ресивером глобальной системы позиционирования, GPS) для обнаружения беспроводных сетей WLAN. Данная утилита идентифицирует SSID сети WLAN, а также определяет, используется ли в ней WEP. Существуют и другие средства, идентифицирующие рабочие станции, подключенные к точке доступа, а также их MAC-адреса например, Kismet (http://www.kismetwireless.net/).

18.3.1.2 Прослушивание

Беспроводные сети по своей природе позволяют соединять с физической сетью компьютеры, находящиеся на некотором расстоянии от нее, как если бы эти компьютеры находились непосредственно в сети. Такой подход позволит подключиться к беспроводной сети организации, располагающейся в здании, человеку, сидящему в машине на стоянке рядом с ним (см. рис. 18.10).

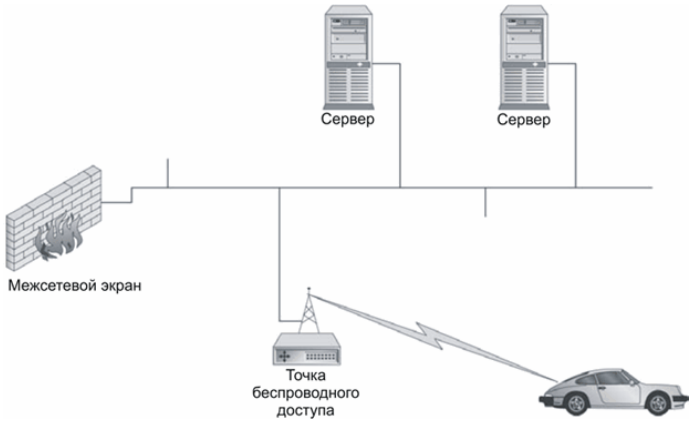


Рис. 18.10 - Прослушивание сети WLAN

18.3.1.3 Активные атаки

Несмотря на то, что прослушивание сети представляет серьезную опасность, активные атаки могут быть еще более опасными. Основной риск, связанный с беспроводными сетями, состоит с том, что злоумышленник может успешно преодолеть периметр сетевой защиты организации. Не следует полагать, что атаки с использованием уязвимостей - это

единственный способ злонамеренного воздействия злоумышленников. Если хакер прослушивает сеть, он может также перехватить пароли и пользовательские идентификаторы. Основные атаки проводимые на WLAN связанны с перехватом информации передаваемой по сети за счет низкой криптостойкости алгоритмов шифрования WEP.

18.3.2 Протокол WEP

Стандарт 802.11x определяет протокол Wired Equivalent Privacy (WEP) для защиты информации при ее передаче через WLAN.

WEP предусматривает обеспечение трех основных аспектов обеспечивающих безопасность.

- ***Аутентификация.*** Служба аутентификации WEP используется для аутентификации рабочих станций на точках доступа. В аутентификации открытых систем рабочая станция рассматривается как аутентифицированная, если она отправляет ответный пакет с MAC-адресом в процессе начального обмена данными с точкой доступа. В реальных условиях данная форма аутентификации не обеспечивает доказательства того, что к точке доступа подключается именно конкретная рабочая станция, а не какой-либо другой компьютер.
- ***Конфиденциальность.*** Механизм обеспечения конфиденциальности базируется на RC4. RC4 - это стандартный мощный алгоритм шифрования, поэтому атаковать его достаточно сложно. WEP определяет систему на базе RC4, обеспечивающую управление ключами, и другие дополнительные службы, необходимые для функционирования алгоритма. WEP поддерживает ключи длиной 40 бит и 128 бит (непосредственный ключ комбинируется с вектором инициализации алгоритма). К сожалению, WEP не определяет механизм управления ключами. Это означает, что многие инсталляции WEP базируются на использовании статических ключей. Действительно, часто на всех рабочих станциях сети используются одни и те же ключи.
- ***Целостность.*** Спецификация протокола WEP включает контроль целостности для каждого пакета. Используемая проверка целостности представляет собой циклическую 32-битную проверку избыточности (CRC). CRC вычисляется для каждого пакета перед его шифрованием, после чего данные в комбинации с CRC шифруются и отправляются в пункт назначения. Несмотря на то что CRC с криптографической точки зрения небезопасна, она защищается шифрованием. Используемая здесь система шифрования может быть достаточно надежной, если алгоритм шифрования обладает достаточной мощностью. Однако недостатки WEP представляют угрозу и для целостности пакетов.

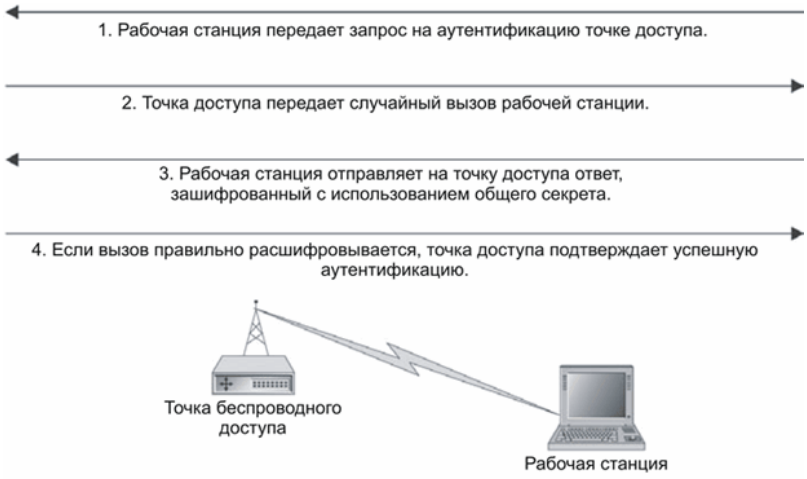


Рис. 18.11 - Аутентификационный обмен WEP

WEP предусматривает использование службы аутентификации. К сожалению, эта служба осуществляет только аутентификацию рабочей станции относительно AP. Она не обеспечивает взаимную аутентификацию, поэтому рабочая станция не получает доказательства того, что AP действительно является авторизованной точкой доступа в данной сети. Таким образом, использование WEP не предотвращает перехват данных или атаки через посредника (см. рис. 18.12).

Рис. 18.12 - Атака на WEP через посредника

18.3.3 Протокол 802.1X - контроль доступа в сеть по портам

Протокол 802.1X разработан в качестве надстройки для всех протоколов контроля доступа 2 уровня, включая Ethernet и WLAN. Так как данный протокол был разработан в то время, когда создатели WLAN искали решения проблем, связанных с WEP, он пришелся как нельзя кстати.

Протокол предназначен для обеспечения обобщенного механизма аутентификации при доступе в сеть и предусматривает следующий набор элементов:

1. **Аутентификатор**. Сетевое устройство, осуществляющее поиск других объектов для аутентификации; для WLAN это может быть AP.

2. **Соискатель**. Объект, которому требуется доступ. В случае с WLAN это может быть рабочая станция.
3. **Сервер аутентификации**. Источник служб аутентификации. 802.1X разрешает централизацию этой функции, поэтому данный сервер является, например, сервером RADIUS.
4. **Сетевая точка доступа**. Точка присоединения рабочей станции к сети. По сути, это порт на коммутаторе или концентраторе. В беспроводной технологии является связью между рабочей станции и точкой доступа.
5. **Процесс доступа через порт (PAE)**. PAE - это процесс, выполняющий протоколы аутентификации. PAE есть как у аутентификатора, так и у соискателя.
6. **Расширяемый протокол аутентификации (EAP)**. Протокол EAP (определен в стандарте RFC 2284) представляет собой протокол, используемый при обмене аутентификационными данными. Поверх EAP могут работать и другие протоколы аутентификации более высокого уровня.

Использование протокола 802.1X позволяет применить более надежный механизм аутентификации, нежели возможности, доступные в 802.11x. При использовании совместно с сервером RADIUS становится возможным централизованное управление пользователями.

18.4 Обеспечение безопасности элекгонной почты

18.4.1 Риски, связанные с использованием электронной почты

Электронная почта — один из наиболее широко используемых видов сервиса, как в корпоративных сетях, так и в Интернет. Она является не просто способом доставки сообщений, а важнейшим средством коммуникации, распределения информации и управления различными процессами в бизнесе. Роль электронной почты становится очевидной, если рассмотреть функции, которые выполняет почта:

- Обеспечивает внутренний и внешний информационный обмен;
- Является компонентом системы документооборота;
- Формирует транспортный протокол корпоративных приложений;
- Является средством образования инфраструктуры электронной коммерции.

Благодаря выполнению этих функций электронная почта решает одну из важнейших на настоящий момент задач — **формирует единое информационное пространство**. В первую очередь это касается создания общей коммуникационной инфраструктуры, которая упрощает обмен информацией между отдельными людьми, подразделениями одной компании и различными организациями.

Электронная почта обладает рядом преимуществ по сравнению с обычными способами передачи сообщений (традиционная почта или факсимильная связь). К ним относятся следующие.

1. Оперативность и легкость использования.
2. Доступность практически в любом месте.
3. Универсальность форматов писем и вложений.
4. Дешевизна сервиса.
5. Надежность и скорость инфраструктуры доставки.
6. Использование для обработки электронной почты прикладного специального программного обеспечения.

Электронная почта обладает многочисленными достоинствами, но именно из-за этих достоинств возникают основные риски, связанные с ее использованием. К примеру, доступность электронной почты превращается в недостаток, когда пользователи начинают применять почту для рассылки спама, легкость в использовании и бесконтрольность приводит к утечкам информации, возможность пересылки разных форматов документов — к распространению вирусов и т.д.

В конечном итоге любой из этих рисков может привести к серьезным последствиям для компании. Это и потеря эффективности работы, и снижение качества услуг информационных систем, и разглашение конфиденциальной информации. Недостаточное внимание к данной проблеме грозит значительными потерями в бизнесе, а в некоторых случаях даже привлечением к юридической ответственности в связи с нарушением законодательства.

Компания подвергается данным рискам в силу ряда свойств электронной почты. Например, благодаря применению MIME-стандарта электронная почта может переносить большие объемы информации различных форматов данных в виде прикрепленных к сообщениям файлов. Такой возможностью сразу воспользовались злоумышленники.

Достоинство электронной почты превратилось в угрозу, поскольку электронная почта стала представлять собой практически идеальную среду для переноса различного рода «опасных» вложений, а именно компьютерных вирусов, вредоносных программ, «троянских» программ и т.п.

Если надлежащий контроль за использованием электронной почты не обеспечен, это может привести к чрезвычайно серьезным последствиям и даже нанести непоправимый ущерб. Избавиться от данного риска можно лишь путем блокировки писем с «опасными» вложениями, а также антивирусной проверки прикрепленных файлов. На практике же оптимальным средством может оказаться **блокировка определенных типов файлов**. Это, как правило, исполняемые файлы (exe, com, bat) и файлы, содержащие макросы и OLE-объекты (файлы, созданные в приложениях MS Office).

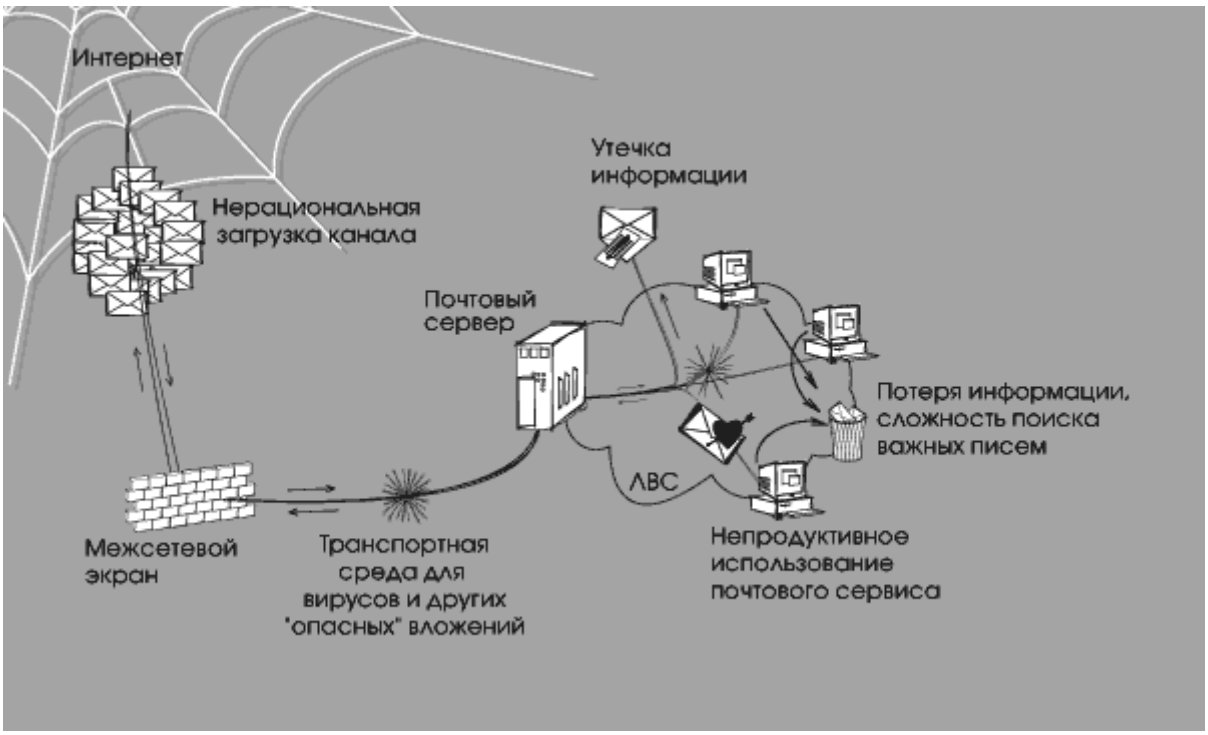


Рис. 18.12 - Негативное воздействие различных факторов на незащищенную почтовую систему

Серьезную опасность для корпоративной сети представляют различного рода **атаки с целью «засорения» почтовой системы.** Это, в первую очередь, пересылка в качестве вложений в сообщениях электронной почты файлов больших объемов или многократно заархивированных файлов. «Открытие» таких файлов или попытка «развернуть» архив может привести к «зависанию» системы. При этом одинаково опасны как умышленные атаки этого типа, например, «отказ в обслуживании» (Denial of Service) и «почтовые бомбы» (mail-bombs), так и «неумышленные», когда пользователи отправляют электронные письма с вложениями большого объема, просто не подумав о том, к каким последствиям может привести открытие подобного файла на компьютере адресата.

Действенный способ избавиться от «засорения» почтовой системы и ее перегрузки — фильтрация по объему передаваемых данных, по количеству вложений в сообщения электронной почты и глубине вложенности архивированных файлов.

Другой особенностью электронной почты является ее доступность и простота в использовании. Во многом результатом этого стало широкое и повсеместное применение этого вида сервиса Интернет. Стихийность развития и отсутствие единых правил функционирования почтового сервиса привели к неконтролируемому использованию электронной почты, а в связи с этим, и к возникновению целого ряда рисков, связанных с неуправляемой циркуляцией электронной почты в сети.

Отсутствие контроля над почтовым потоком, как правило, становится причиной того, что сотрудники компании **используют электронную почту в**

целях, не связанных с деятельностью компании (например, для обмена видео-файлами и графикой, частной переписки, ведения собственного бизнеса с использованием почтовых ресурсов компании, рассылки резюме в различные организации и т.п.).

Кроме того, к такому же результату может привести **непродуктивное использование почтовых ресурсов** в трудовой деятельности сотрудников (например, чрезмерное увлечение почтовой перепиской в случаях, когда необходимости в такой переписке нет, использование электронной почты не по назначению и т.п.). Причиной этого, как правило, является отсутствие в компании правил, регламентирующих использование системы электронной почты. Последствиями непродуктивного использования почтового сервиса являются снижение производительности труда в компании, а также излишние финансовые затраты. Сэкономить средства в значительной степени поможет **проведение анализа эффективности использования системы электронной почты, который основывается на базе статистических данных о функционировании системы. Подобную статистику возможно получить лишь в случае ведения архива электронной почты.** Обработка информации, содержащейся в архиве, позволяет получать отчеты о различных параметрах электронной почты, ее объемах и структуре, представить наглядную картину использования почтового трафика сотрудниками компании, а это, в свою очередь, поможет предотвратить использование электронной почты, несвязанное с деятельностью компании, и повысить эффективность работы корпоративной почтовой системы.

Передача в электронных письмах графических, видео и звуковых файлов, которые, как правило, имеют **большой объем** даже если такая передача предусмотрена условиями ведения бизнеса, приводит к значительной перегрузке сети, соответственно к дополнительным финансовым затратам на ее обслуживание. Избежать этого, а значит и добиться значительной экономии средств компании, поможет, так называемая, **отложенная доставка писем**, которая позволяет доставлять сообщения больших объемов в то время, когда загрузка сети не имеет критического значения (например, в ночное время, в выходные дни и т.п.).

К «засорению» трафика также ведет рассылка **спама**. Как правило, это письма, содержащие навязчивые предложения самых разнообразных услуг, товаров и т.п. Такого рода почта является «группой риска» с точки зрения переноса вирусов. Большое количество ненужной почты загружает каналы, «замусоривает» почтовые ящики, отнимает время на удаление ненужных писем и повышает вероятность случайного удаления нужных. Конечно рассылка, например, сообщений рекламного характера, напрямую не преследует цели «засорить» почтовую систему организации, однако косвенно приводит к негативным последствиям. Использование списков рассылки, в которую могут входить все пользователи одной корпоративной сети, и получение одновременно всеми этими пользователями сообщений рекламного характера грозит компании снижением производительности ее

сетевых ресурсов. **Блокировка спама**, в первую очередь, связана с контекстным анализом сообщений, то есть проверкой электронной почты на наличие ключевых слов и выражений, которые обычно употребляются в сообщениях рекламного характера.

Переписка с внешними корреспондентами представляет наибольшую угрозу из-за особенностей электронной почты (невозможность контролировать маршрут передачи писем, а также их копирование и перенаправление, осуществлять аутентификацию отправителя/получателя, возвращать письма после их отправления). Кроме того невозможен либо затруднен контроль количества отправляемых копий письма. Содержимое сообщения может быть прочитано в процессе передачи его по Интернету, поскольку заголовки и содержимое электронных писем часто передаются в открытом виде.

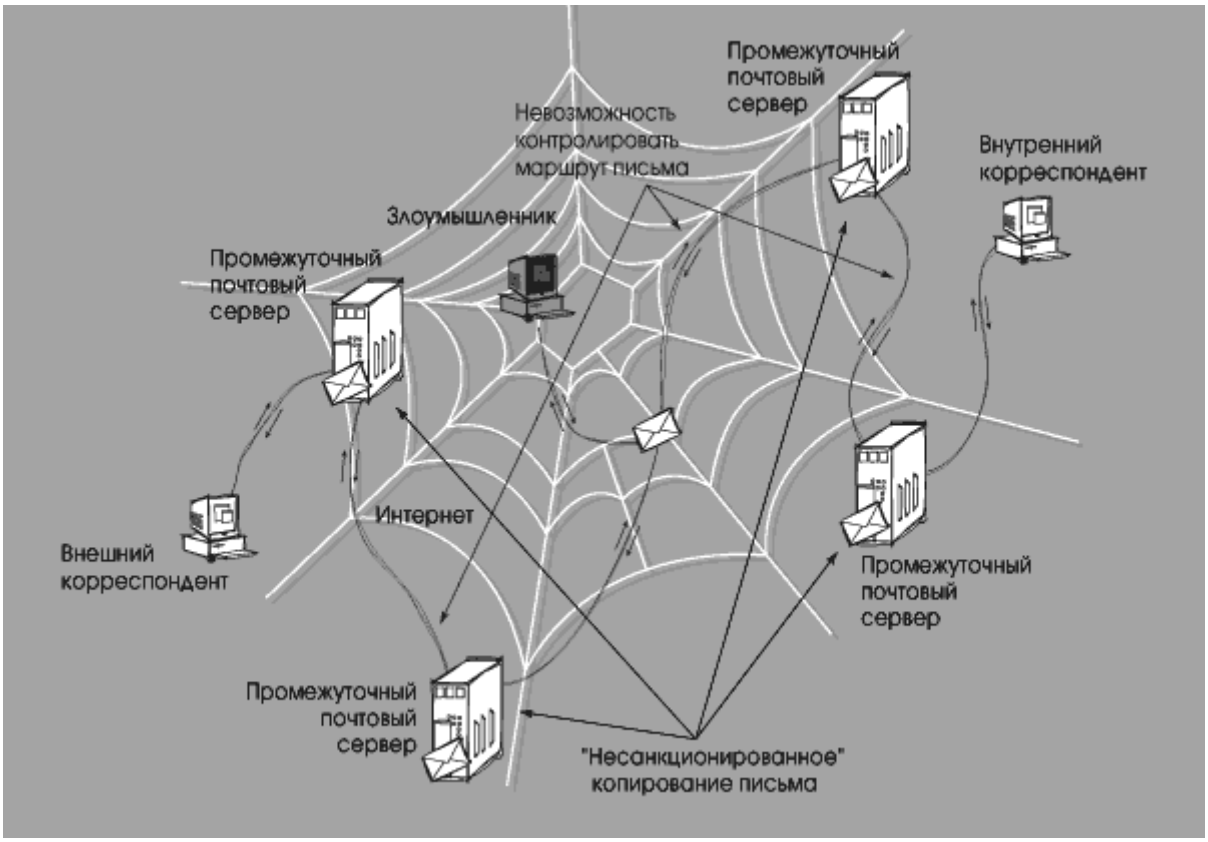


Рис. 18.13 - Проблемы, возникающие при пересылке электронной почты через Интернет

Другой проблемой, связанной с особенностями электронной почты, является то, что электронная почта позволяет неконтролируемое накопление информации в архивах и практически неуничтожима. В противоположность бытующему мнению, удалить электронную почту непросто. Резервные копии сообщений могут оставаться на персональных компьютерах отправителя и получателя или в сети компаний, где они работают. Если электронное почтовое сообщение отправлено через коммерческую службу или через Интернет, то оно будет передаваться через несколько различных серверов.

Каждый сервер в цепочке между отправителем и получателем может сохранить копию сообщения в своих архивах. Даже методичное выяснение местонахождения каждой копии электронного письма с последующим его удалением не дает никакой гарантии того, что сообщение не осталось на жестком диске компьютера или сервера.

Все эти особенности, а также простота копирования электронного сообщения и невозможность проконтролировать данную операцию приводят к тому, что сотрудник может передать корпоративную информацию любому количеству людей как внутри, так и за пределами компании анонимно и без соответствующего разрешения, сразу или по истечении какого-либо времени. При этом, такая информация может представлять собой служебную информацию компании это грозит серьезным нарушением конфиденциальности и может привести к неприятным для компании последствиям.

Чтобы обеспечить **защиту от утечки конфиденциальной информации** из сети, необходимо осуществлять контроль адресатов, **фильтрацию** передаваемых данных на наличие в текстах сообщений или в прикрепленных к электронному письму файлах **слов и выражений, имеющих отношение к «закрытой» тематике**, осуществлять разграничение доступа различных категорий пользователей к архивам электронной почты и т.п.

Одно из основных отличий электронной почты состоит в формальном к ней отношении (по сравнению с другими видами коммерческих коммуникаций):

- Во-первых, большинство пользователей относятся к электронной почте как к чему-то временному, то есть поступают с ней по принципу «прочитал и выкинул». При таком отношении существует риск **случайного удаления значимой информации**. Кроме того, существует опасность **потери переписки с важным клиентом**. Все эти проблемы решаются путем **создания в организации архива электронной почты**.
- Во-вторых, такое отношение к электронной почте приводит к тому, что из-за кажущейся недолговечности электронных сообщений люди часто используют их для того, чтобы выразить чувства и мнения в выражениях, которые они никогда не позволили бы себе употребить в традиционных письмах. Публикация таких писем в сети может нанести серьезный ущерб репутации компании или явиться причиной юридических исков к ней.

Еще одна область связана с возможностью привлечения к юридической ответственности компании и ее сотрудников — **за нарушение авторского права**. Защищенные этим правом материалы могут содержаться или в сообщении электронной почты, или в присоединенных файлах. К подобным материалам относятся графическая, аудио, видео и различная текстовая информация, т. е. любая информация, которая может быть представлена в

электронном виде и передана по компьютерным сетям. Копирование или распространение этих материалов без предварительного согласия автора или владельца авторских прав является нарушением закона. Если компания допускает, чтобы материалы почты, защищенные авторским правом, использовались сотрудниками, не имеющими на это полномочий, то она может быть привлечена к ответственности за прямое или косвенное пособничество нарушению авторского права.

18.4.2 Средства обеспечения безопасности электронной почты

Учитывая описанные выше риски, связанные с использованием электронной почты, организациям необходимо принять соответствующие меры для защиты от них.

Подход к защите должен быть всесторонним и комплексным — необходимо сочетать организационные меры с использованием соответствующих технических средств.

К организационным мерам относятся разработка и внедрение в компании политики использования электронной почты. Технические средства должны обеспечить выполнение данной политики как за счет мониторинга почтового трафика, так и за счет адекватного реагирования на нарушения.

Очень важно отметить, что политика использования электронной почты первична по отношению к средствам ее реализации, поскольку составляет основу для формирования комплекса мер по защите информационной системы от вышеперечисленных рисков. Сначала необходимо сформулировать политику, составить правила использования электронной почты, определить, как созданная система должна реагировать на определенные нарушения данной политики и только затем переводить правила на компьютерный язык того средства, которое используется для контроля выполнения положений политики использования электронной почты.

К таким техническим средствам относится специальное программное обеспечение, называемое **система контроля содержимого электронной почты** (content security software). В функции таких систем входит контроль почтового трафика и ведение архива переписки по электронной почте. К данным системам предъявляются следующие требования:

- Проведение текстового анализа;
- Фильтрация передаваемых данных:
  1. по размеру и объему данных;
  2. по количеству вложений в сообщения электронной почты;
  3. по типу файлов (вложенных в электронную почту);
  4. по адресу электронной почты;
- Контроль использования почтовых ресурсов и разграничение доступа к ним различных категорий пользователей;

- Отложенную доставку сообщений электронной почты по расписанию;
- Ведение полнофункционального архива электронной почты.

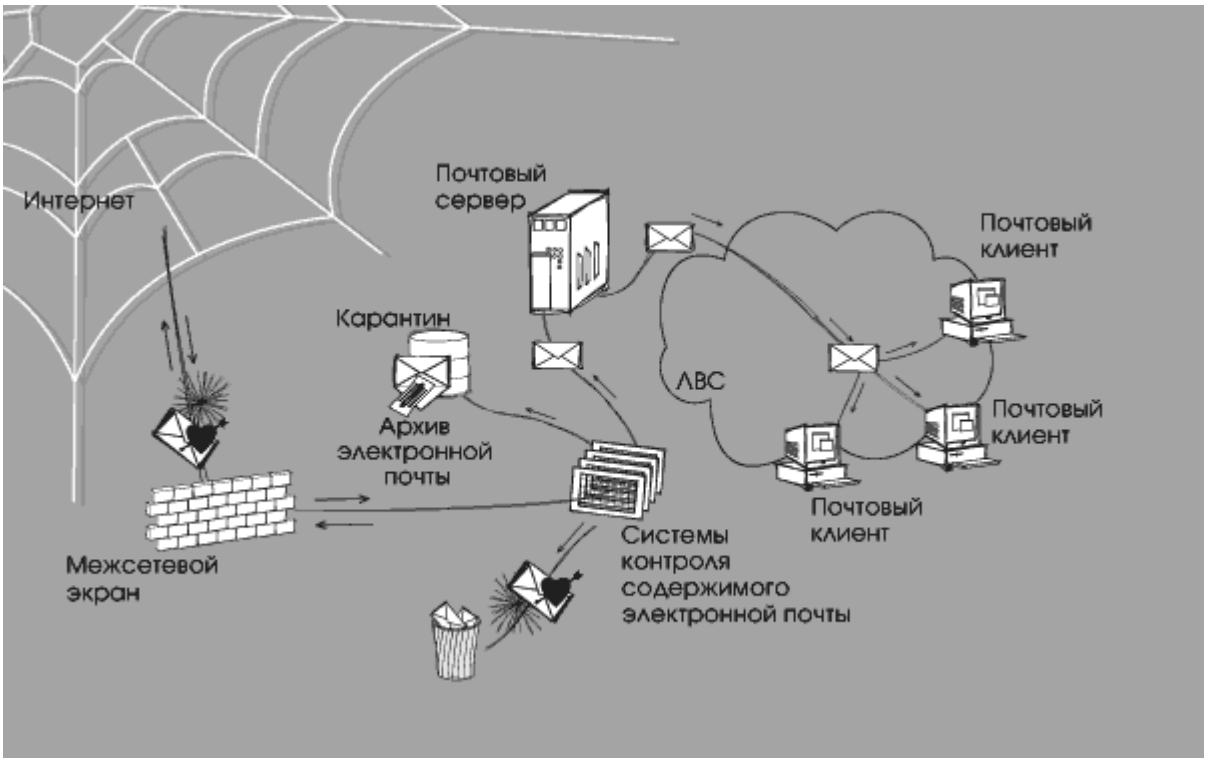


Рис. 18.14 - Решение проблем защиты почтовой системы

Выполнение этих требований обеспечивается применением в средствах защиты определенных механизмов. К таким механизмам могут относиться:

1. Рекурсивная декомпозиция (специальный алгоритм, применяемый для разбора сообщений электронной почты на составляющие компоненты с последующим анализом их содержимого);
2. Эвристическое определение кодировок текстов;
3. Определение типа файлов по сигнатуре;
4. Полнотекстовый поиск по архиву электронной почты и т.п.

18.4.3 Политика использования электронной почты

Средство защиты — система контроля содержимого электронной почты, само по себе никаких задач по обеспечению безопасности не решает. Это всего лишь «машина», которая помогает человеку в решении данной проблемы. Поэтому задачу по обеспечению безопасности необходимо такой «машине» поставить. Это означает, что должен быть выработан специальный набор правил, который в дальнейшем будет переведен на язык машины. **Данный набор правил называется «политикой использования электронной почты».**

Во многих организациях такие правила существуют уже длительное время. Как и всякая ограничительная мера, они создают определенные

неудобства пользователям системы, а если пользователю что-то неудобно, он либо перестает этим пользоваться, либо старается обойти препятствия. Поэтому такого рода политики, не подкрепленные техническими средствами контроля за их выполнением, постепенно теряют силу. Программные системы, ориентированные на фильтрацию почты, следует позиционировать именно как инструмент для внедрения и контроля исполнения этих правил.

Таким образом, ***политика использования электронной почты — это закрепленные в письменном виде и доведенные до сотрудников инструкции и другие документы, которые регламентируют их деятельность и процессы, связанные с использованием системы электронной почты.*** Данные документы и инструкции обладают юридическим статусом и, как правило, предоставляются для ознакомления сотрудникам организации.

Политика использования электронной почты является важнейшим элементом общекорпоративной политики информационной безопасности и неотделима от нее.

Политика должна соответствовать следующим критериям:

- Быть лаконично изложенной и понятной всем сотрудникам компании, простота написания не должна привести к потере юридического статуса документа;
- Исходить из необходимости защиты информации в процессе экономической деятельности компании;
- Быть согласованной с другими организационными политиками компании (регламентирующими финансовую, экономическую, юридическую и другие сферы деятельности компании);
- Иметь законную силу, т.е. политика, как документ, должна быть одобрена и подписана всеми должностными лицами руководящего звена компании, а ее выполнение должно быть детально продумано;
- Не противоречить федеральным и местным законам;
- Определять меры воздействия на сотрудников, нарушивших положения данной политики;
- Соблюдать баланс между степенью защищенности информации и продуктивностью деятельности компании;
- Детально определять мероприятия по обеспечению политики использования электронной почты в компании.

Политика использования электронной почты, как правило, рассматривается с двух сторон — как официально оформленный юридический документ и как материал, который описывает технику реализации политики.

Как документ политика должна включать:

- Положение, что электронная почта является собственностью компании и может быть использована только в рабочих целях.

- Указание на то, что применение корпоративной системы электронной почты не должно противоречить законодательству РФ и положениям политики безопасности.
- Инструкции и рекомендации по использованию и хранению электронной почты.
- Предупреждение о потенциальной ответственности сотрудников компании за злоупотребления при использовании электронной почты в личных целях и возможном использовании электронной почты в судебных и служебных разбирательствах.
- Письменное подтверждение того, что сотрудники компании ознакомлены с политикой использования электронной почты и согласны с ее положениями.

С технической точки зрения политика устанавливает правила использования электронной почты, то есть определяет следующее:

Что контролируется	Прохождение каких сообщений входящей, исходящей или внутренней электронной почты должно быть разрешено или запрещено.
На кого распространяется	Категории лиц, которым разрешено или запрещено отправлять исходящие или получать входящие сообщения электронной почты.
Как реагирует система	Что необходимо делать с теми или иными сообщениями электронной почты, которые удовлетворяют или не удовлетворяют критериям, определенным правилами использования электронной почты.

18.4.4 Системы контроля содержимого электронной почты

Внедрение политики использования электронной почты требует от руководства компании понимания, что наличие только документально оформленной политики не гарантирует ее выполнения. Необходимо создание в компании соответствующих условий реализации данной политики. При этом важнейшим условием является наличие в корпоративной сети программно-технических средств контроля выполнения положений и требований политики. К таким средствам относятся системы контроля содержимого электронной почты.

Системы контроля содержимого электронной почты — это программное обеспечение, способное анализировать содержание письма по различным компонентам и структуре в целях реализации политики использования электронной почты.

К особенностям данных продуктов относятся:

1. Применение при анализе содержания специально разработанной политики использования электронных писем.

2. Способность осуществлять «рекурсивную декомпозицию» электронных писем.
3. Возможность распознавания реальных форматов файлов вне зависимости от различных способов их маскировки (искажение расширения файлов, архивирование файлов и т.п.).
4. Анализ множества параметров сообщения электронной почты.
5. Ведение архива электронной почты
6. Анализ содержимого сообщения электронной почты и прикрепленных файлов на наличие запрещенных к использованию слов и выражений.

Системы контроля электронной почты помимо основной своей задачи мониторинга почтового трафика способны выполнять и другие функции. Практика показала, что в настоящее время такие системы используются в качестве:

1. Средств управления почтовым потоком.
2. Средств управления доступом.
3. Средств администрирования и хранения электронной почты.
4. Средств аудита контента (важнейшую функцию которого осуществляет архив электронной почты).
5. Основы системы документооборота.

18.4.5 Требования к системам контроля содержимого электронной почты

Спектр возможностей всех категорий систем контроля содержимого электронной почты достаточно широк и существенно меняется в зависимости от производителя. Однако ко всем системам предъявляются наиболее общие требования, которые позволяют решать задачи, связанные с контролем почтового трафика.

Основныме требования предьявляемые к таким системам - полнота и адекватность

1. ***Полнота*** *— это способность систем контроля обеспечить наиболее глубокую проверку сообщений электронной почты.* Это предполагает, что фильтрация должна производиться по всем компонентам письма. При этом ни один из объектов, входящих в структуру электронного сообщения, не должен быть «оставлен без внимания». Условия проверки писем должны учитывать все проблемы, риски и угрозы, которые могут существовать в организации, использующей систему электронной почты.

2. ***Адекватность*** *— это способность систем контроля содержимого как можно более полно воплощать словесно сформулированную политику использования электронной почты, иметь все необходимые средства реализации написанных людьми правил в понятные системе условия фильтрации.*

К другим наиболее общим требованиям относятся:

3. ***Текстовый анализ электронной почты*** - анализ ключевых слов и выражений с помощью встроенных словарей. Данная возможность позволяет обнаружить и своевременно предотвратить утечку конфиденциальной информации, установить наличие запрещенного содержания, остановить рассылку спама, а также передачу других материалов, запрещенных политикой безопасности. При этом качественный анализ текста должен предполагать морфологический анализ слов, то есть система должна иметь возможность генерировать и определять всевозможные грамматические конструкции слова. Эта функция приобретает большое значение в связи с особенностями русского языка, в котором слова имеют сложные грамматические конструкции.

4. ***Контроль отправителей и получателей сообщений электронной почты.*** Данная возможность позволяет фильтровать почтовый трафик, тем самым реализуя некоторые функции межсетевого экрана в почтовой системе.

5. ***Разбор электронных писем на составляющие их компоненты*** (MIME-заголовки, тело письма, прикрепленные файлы и т.п.), устранение «опасных» вложений и последующий сбор компонентов письма воедино, причем с возможностью добавлять к сообщению электронной почты необходимые для администраторов безопасности элементы (например, предупреждения о наличии вирусов или «запрещенного» текста в содержании письма).

6. ***Блокировка или задержка сообщений большого размера*** *до того момента, когда канал связи будет менее всего загружен (например, в нерабочее время).* Циркуляция в почтовой сети компании таких сообщений может привести к перегрузке сети, а блокировка или отложенная доставка позволит этого избежать.

7. ***Распознавание графических, видео и звуковых файлов.*** Как правило, такие файлы имеют большой размер, и их циркуляция может привести к потере производительности сетевых ресурсов. Поэтому способность распознавать и задерживать данные типы файлов позволяет предотвратить снижение эффективности работы компании.

8. ***Обработка сжатых/архивных файлов.*** Это дает возможность проверять сжатые файлы на содержание в них запрещенных материалов.

9. ***Распознавание исполняемых файлов.*** Как правило, такие файлы имеют большой размер и редко имеют отношение к коммерческой деятельности компании. Кроме того, исполняемые файлы являются основным источником заражения вирусами, передаваемыми с электронной почтой. Поэтому способность распознавать и задерживать данные типы файлов позволяет предотвратить снижение эффективности работы компании и избежать заражения системы.

10. ***Контроль и блокирование спама.*** Циркуляция спама приводит к перегрузке сети и потере рабочего времени сотрудников. Функция контроля и блокирования спама позволяет сберечь сетевые ресурсы и предотвратить

снижение эффективности работы компании. Основными способами защиты от спама являются: проверка имен доменов и IP-адресов источников рассылки спама по спискам, запрос на указанный адрес отправителя (блокировка в случае отсутствия ответа), текстовый анализ спам-сообщения на наличие характерных слов и выражений в заголовках электронной почты (from/subject), проверка заголовков на соответствие спецификации RFC-822 и т.п.

11. ***Способность определять число вложений в сообщениях электронной почты.*** Пересылка электронного письма с большим количеством вложений может привести к перегрузке сети, поэтому контроль за соблюдением определенных политикой информационной безопасности ограничений на количество вложений обеспечивает сохранение ресурсов корпоративной сети.

12. ***Контроль и блокирование программ-закладок*** (cookies), вредоносного мобильного кода (Java, ActiveX, JavaScript, VBScript и т.д.), а также файлов, осуществляющих автоматическую рассылку (так называемые «Automatic Mail-to»). Эти виды вложений являются крайне опасными и приводят к утечке информации из корпоративной сети.

13. ***Категоризация ресурсов почтовой системы компании*** («административный», «отдел кадров», «финансы» и т.д.) и разграничение доступа сотрудников компании к различным категориям ресурсов сети (в т.ч. и в зависимости от времени суток).

14. ***Реализация различных вариантов реагирования***, в том числе: удаление или временная блокировка сообщения; задержка сообщения и помещение его в карантин для последующего анализа; «лечение» зараженного вирусом файла; уведомление администратора безопасности или любого другого адресата о нарушении политики безопасности и т.п.

15. ***Возможность модификации данных***, которая предусматривает, например, удаление неприемлемых вложений и замену их на тексты заданного содержания. Такая возможность позволит администратору удалять из писем прикрепленные файлы, тип которых запрещен политикой безопасности компании. К таким типам могут относиться исполняемые, видео и звуковые файлы, не имеющие отношения к деятельности компании. А это, в конечном итоге, позволит избежать заражения сети вирусами и добиться от сотрудников продуктивного использования почтового сервиса.

16. ***Ведение полнофункционального архива электронной почты****, способного обеспечить хранение в режиме on-line большого количества электронной почты с высоким уровнем доступности данных.* На основании хранящейся в архиве информации, возможно проводить дальнейший анализ почтового потока компании, корректировать работу системы, осуществлять анализ инцидентов, связанный с злоупотреблением сотрудниками компании почтовым сервисом и т.п.

На Рис. 18.15 представлена последовательность работы типичной системы контроля содержимого электронной почты. Схема обработки сообщения, как правило, включает в себя следующие этапы: рекурсивная декомпозиция электронного письма; анализ содержимого электронного письма; «категоризация» электронного письма (отнесение к определенной категории); действие над письмом по результатам присвоения категории.

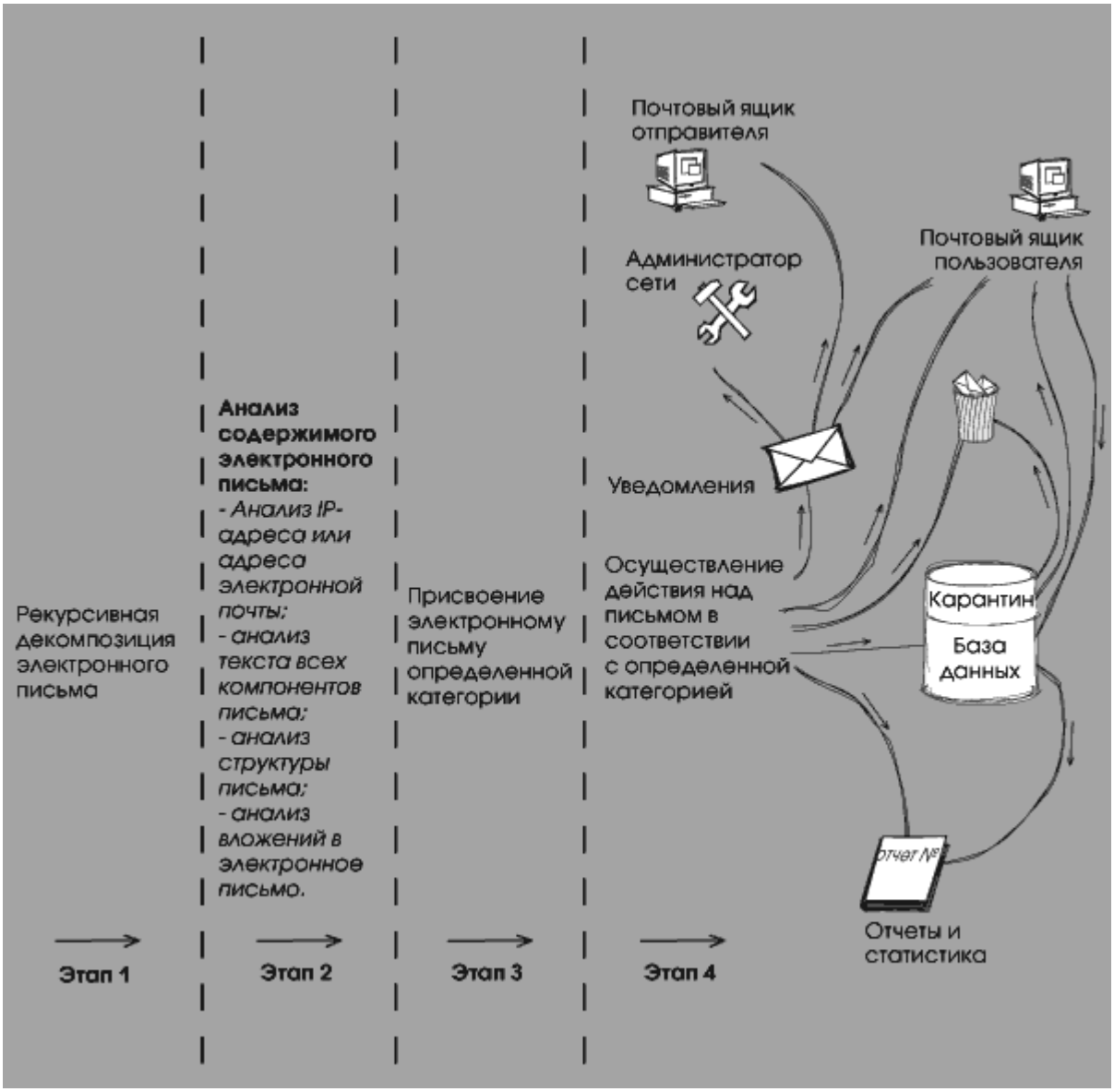


Рис. 18.15 - Схема обработки сообщения системой контроля содержимого электронной почты

18.4.6 Принципы функционирования систем контроля содержимого электронной почты

Каждое попадающее в систему электронное письмо должно проверяться на соответствие заданным условиям. При этом, по меньшей мере, должны выполняться следующие условия отбора писем:

- условия на почтовые заголовки;

- условия на структуру письма (наличие, количество и структура вложений);
- условия на типы вложений (MS Office, исполняемые, архивы и т.п.);
- условия на содержимое (текст) писем и вложений;
- условия на результат обработки письма.

Кроме того, система должна позволять анализировать почтовые сообщения по всем их составляющим: атрибутам конверта, заголовкам сообщения, MIME-заголовкам, телу сообщения, присоединенным файлам.

18.4.6.1 Категоризация писем и фильтрация спама

Рассмотрим вопрос категоризации писем. Важно отметить, что гибкость при фильтрации почтовых сообщений особенно необходима, когда это касается такой проблемы, как спам. Одним из главных критериев выбора системы контроля содержимого электронной почты в настоящее время является как раз ее способность как можно более качественно справляться с данной проблемой.

Существует четыре основные методики определения, какое письмо относится к спаму, а какое нет.

1. *Выявление спама по наличию в письме определенных признаков*, таких как наличие ключевых слов или словосочетаний, характерное написание темы письма (например, все заглавные буквы и большое количество восклицательных знаков), а также специфическая адресная информация.

2. *Определение адреса отправителя и его принадлежности к, так называемым, «черным спискам» почтовых серверов Open Relay Black List (ORBL).* В эти списки заносятся те серверы, которые замечены в массовых рассылках спама и идея состоит в том, чтобы вообще не принимать и не транслировать почту, исходящую с этих серверов.

3. *Совместное использование методик по фильтрации по определенным признакам и проыеркой «черных списко».* По продуктивности мало чем отличается от двух первых. Результаты тестирования хорошо настроенного фильтра с применением обеих методик показывают, что из 100% спам-сообщений обнаруживается только 79,7%. При этом был выявлен значительный процент ложных срабатываний, а это значит, что к спаму были отнесены обычные письма (1,2% от задержанных писем), а это грозит для компании потерей важной информации. Некачественное разделение спама и обычных писем обусловлено, в том числе и некоторой «однобокостью» стандартных фильтров. При отбраковке писем учитываются «плохие» признаки и не учитываются «хорошие», характерные для полезной переписки.

4. *автоматическая настройка фильтров согласно особенностям индивидуальной переписки, а при обработке учет признаков как «плохих»,*

так и «хороших» фильтров. Эта четвертая методика, предложенная американским программистом и предпринимателем Полом Грэмом. Методика основывается на теории вероятностей и использует для фильтрации спама статистический алгоритм Байеса. По имеющимся оценкам, этот метод борьбы со спамом является весьма эффективным. Так, в процессе испытания через фильтр были пропущены 8000 писем, половина из которых являлась спамом. В результате система не смогла распознать лишь 0,5% спам-сообщений, а количество ошибочных срабатываний фильтра оказалось нулевым.

Требование полного разбора письма при решении задачи категоризации следует дополнить требованием устойчивости.

- *Во-первых, система должна быть устойчивой по отношению к обработке писем с некорректной стрктурой.* Структура письма подчиняется определенным правилам. Разбор письма на составляющие основан на применении этих правил к конкретному письму. Возможны случаи, когда почтовая программа автора письма формирует письмо с нарушением этих правил. В этом случае письмо не может быть корректно разобрано..
- *Во-вторых, система должна надежно определять типы файлов-вложений.* Под «надежностью» имеется в виду определение, не основанное на имени файла, а также на информации, вписываемой в письмо почтовым клиентом при прикреплении файла (mime-type). Такая информация может быть недостоверна либо в результате сознательных попыток обмануть систему контроля, либо в результате неправильных настроек почтовой программы отправителя. Бессмысленно запрещать пересылку файлов типа JPEG, если файл picture.jpg после переименования в page.txt пройдет незамеченным.
- *В-третьих, система болжна обеспечивать полноту проводимых проверок, то есть высокое количество и разнообразие критериев анализа электронной почты.* При этом система должна осуществлять фильтрацию по любым атрибутам сообщений, по объему сообщений и вложенных файлов, по количеству и типу вложений, по глубине вложенности, а также уметь анализировать содержимое прикрепленных файлов вне зависимости от того, являются ли эти файлы сжатыми или архивными. Существенным преимуществом многих продуктов является возможность создания собственного сценария обработки сообщений электронной почты.

При анализе текста нужно иметь возможность работать с нормализованными словоформами и т.д.

18.4.6.2 Реализациия политики использования

Рассмтрим не отдельные правила, а все множество правил, составляющих политику. Любая реалистичная политика состоит из целого множества правил, которые, естественно, объединяются в группы. Очевидно, что правила для исходящей почты отличаются от правил для входящей, правила для руководства компании — от правил для рядовых сотрудников и т.д. Более того, поскольку правила применяются к письму в определенной последовательности, хотелось бы, чтобы эта последовательность была логичной и могла зависеть от результатов анализа письма. Все это вместе приводит к требованию «прозрачности»: правила, заданные в системе, должны «читаться» как правила, написанные на естественном языке, понятном человеку.

Все сказанное выше относилось к анализу письма. Однако сам по себе анализ ничего не дает. По его результатам письмо должно быть отнесено к какой-нибудь категории (безопасное, важное, неразрешенное и т.п.). Если такая категоризация проведена, то можно говорить о каких-либо действиях по отношению к проанализированному письму, например, доставить его адресату, заблокировать, и т.д. Другими словами, необходима возможность задавать системе правила, по которым она обрабатывает письма.

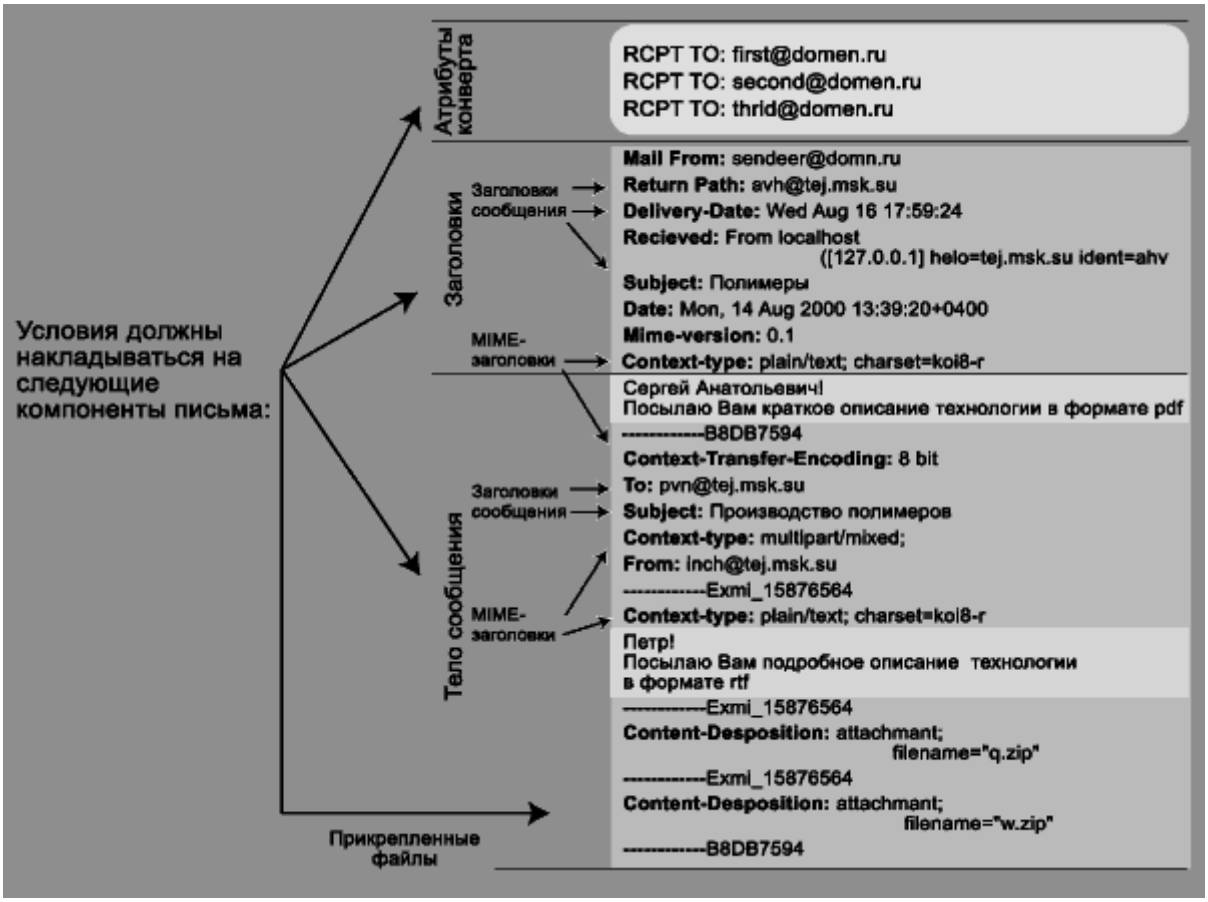


Рис. 18.16 - Фильтрация по всем компонентам письма

Любое правило можно представить себе как связку «условие + действие». Какие же действия нужны для того, чтобы обеспечить реализацию разумной политики?

Само по себе отнесение письма к определенной категории уже может рассматриваться как неявное действие. На этом действии следует остановиться подробнее. Дело в том, что жесткая категоризация как основа для принятия решений по электронному письму оказывается весьма непрактичной. Действительно, пусть мы выделили категорию писем «письмо, отправленное на запрещенный адрес» для того, чтобы блокировать доставку. С другой стороны, у нас может быть категория «письма руководства компании», которые надо отправлять безусловно. Что делать с письмом президента, отправленным по «запрещенному» адресу? Здравый смысл подсказывает, что приоритет должен быть отдан категории «письма руководства компании», что, безусловно, и будет сделано в системе с жесткой категоризацией. Однако будет потеряна существенная информация о письме. Разумный выход из таких ситуаций заключается в возможности относить письма сразу к нескольким категориям. Такая «свободная категоризация» позволит системе гибко реагировать на самые различные комбинации данных, содержащихся в письмах.

На Рис. 18.17 показана схема действий, поддерживаемых правилами фильтрации почтовых сообщений типичной системы контроля содержимого электронной почты.

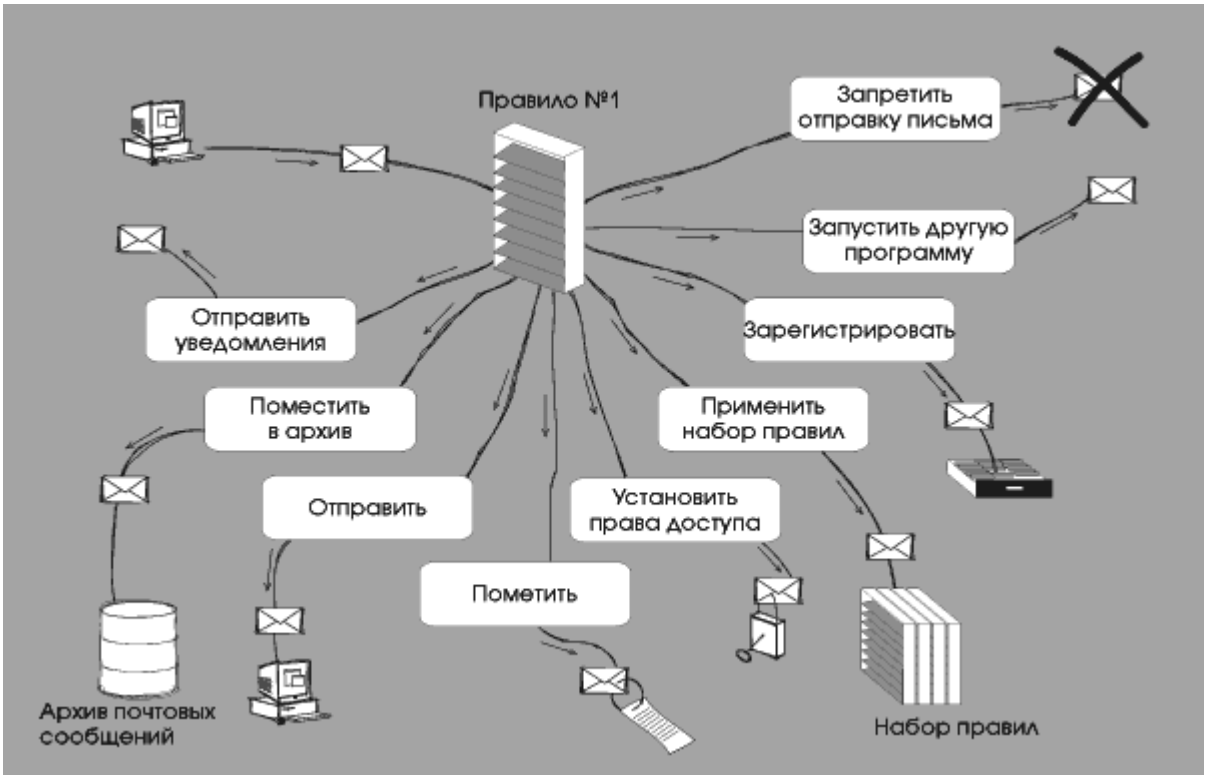


Рис. 18.17 - Схема реагирования типичной системы контроля содержимого электронной почты

18.4.6.3 Долговременное хранение и архивирование

В последнее время большое значение для обеспечения безопасности информационных систем приобрело наличие в компании архива почтовых сообщений. Некоторые разработчики систем контекстного анализа предусматривают прикрепление к своим продуктам специальных модулей архивирования. Именно наличие архива электронной почты и определяет в настоящее время полнофункциональность продуктов этой категории. При этом ведение архива — это не просто автоматическая архивация почтовых сообщений в файл, а способность регистрации сообщений и учета необходимой информации на протяжении всего жизненного цикла сообщения, возможность получения любых выборок и статистики из архива по запросам, созданным с использованием любых критериев.

Кроме того, долговременный архив предоставляет возможность ретроспективного анализа почтовых потоков и не только позволяет найти виновных в нарушении принятых в компании правил по прошествии определенного времени, но и дает материал для построения объективной и обоснованной политики использования электронной почты.

18.4.6.4 Контекстный контроль содержимого

Отличительным признаком средств контекстного анализа является способность накопления статистики и генерации отчетов. Многие продукты имеют в своем арсенале только встроенные формы отчетов, другие способны осуществлять только просмотр статистики работы конкретного пользователя системы электронной почты.

Одним из основных критериев оценки систем контекстного анализа для российского рынка является поддержка продуктом различных кодировок кириллицы (CP1251, CP866, ISO8859-5, KOI8-R, MAC), что дает возможность анализа русскоязычных текстов. Кроме того, «проклятие» множественных кодировок тяготеет над российскими информационными системами. Все осложняется тем, что разные части письма, включая почтовые заголовки, могут быть написаны в разных кодировках. Вдобавок эти кодировки не всегда указаны или не всегда указаны верно.

Рассмотрим вопрос, касающийся архитектуры систем контроля содержимого электронной почты. В подобных продуктах уникальной особенностью является открытая архитектура, которая позволяет разработчикам расширять функциональные возможности системы, интегрируя в нее дополнительные модули и не затрагивая ее ядра. Это дает возможность постоянно наращивать способности системы контроля содержимого по защите электронной почты и одновременно с этим экономить значительные средства, которые могут потребоваться на модернизацию всей системы.

ЧАСТЬ 6. ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ В ОПЕРАЦИОННЫХ СИСТЕМАХ И ПРОГРАММНОМ ОБЕСПЕЧЕНИИ

19. СРЕДСТВА УПРАВЛЕНИЯ БЕЗОПАСНОСТЬЮ В АРХИТЕКТУРЕ ОПЕРАЦИОННЫХ СИСТЕМ WINDOWS

19.1 Средства управления безопасностью

Для управления системой безопасности в ОС Windows имеются разнообразные и удобные инструментальные средства. В частности, потребуется умение управлять учетными записями пользователей при помощи панели «Пользователи и пароли». Кроме того понадобится контролировать привилегии пользователей при помощи панели «Назначение прав пользователям». Рекомендуется также освоить работу с утилитой просмотра данных маркера доступа процесса WhoAml.exe, утилитами просмотра и редактирования списков контроля доступа (cacls.exe, ShowACLs.exe, SubInACL,exe, SvcACL.exe), утилитой просмотра маркера доступа процесса PuList.exe и рядом других.

19.1.1 Система управления доступом

Подсистема защиты данных является одной из наиболее важных. В центре системы безопасности ОС Windows находится система контроля доступа.

С каждым процессом или потоком, то есть активным компонентом (**субъектом**), связан маркер доступа, а у каждого защищаемого объекта (например, файла) имеется дескриптор защиты. Проверка прав доступа обычно осуществляется в момент открытия объекта и заключается в сопоставлении прав **субъекта** списку прав доступа, который хранится в составе дескриптора защиты **объекта**

Защищаемые объекты Windows включают:

- файлы,
- устройства,
- каналы,
- события,
- мьютексы,
- семафоры,
- разделы общей памяти,
- разделы реестра
- и другое.

*Сущность, от которой нужно защищать объекты, называется **«субъектом»**.* Субъектами в Windows являются процессы и потоки, запускаемые конкретными пользователями. *Субъект безопасности - активная системная составляющая, а объект - пассивная.*

Помимо дискреционного доступа Windows поддерживает управление ***привилегированным*** **доступом**. Это означает, что в системе имеется пользователь-администратор с неограниченными правами.

Кроме того, для упрощения администрирования пользователи Windows объединены в **группы**. Пользователь, как член группы, облекается, таким образом, набором полномочий, необходимых для его деятельности, и играет определенную роль. Подобная стратегия называется **управление *ролевым* доступом**.

Ключевая цель системы защиты Windows *- следить за тем, кто и к каким объектам осуществляет доступ.* Система защиты хранит информацию, относящуюся к безопасности для каждого пользователя, группы пользователей и объекта. Модель защиты ОС Windows требует, чтобы субъект на этапе открытия объекта указывал, какие операции он собирается выполнять в отношении этого объекта.

Единообразие контроля доступа *к различным объектам (процессам, файлам, семафорам и др.) обеспечивается тем, что с каждым процессом (потоком) связан маркер доступа, а с каждым объектом - дескриптор защиты. Маркер доступа в качестве параметра имеет идентификатор пользователя, а дескриптор защиты - списки прав доступа. ОС может контролировать попытки доступа, которые прямо или косвенно производятся процессами и потоками, инициированными пользователем.*

19.1.2 Пользователи и группы пользователей

Каждый пользователь (и каждая группа пользователей) системы должен иметь учетную запись (account) в базе данных системы безопасности. Учетные записи идентифицируются именем пользователя и хранятся в базе данных SAM (Security Account Manager) в разделе HKLM/SAM реестра.

Учетная запись пользователя содержат набор сведений о пользователе, такие, как имя, пароль (или реквизиты), комментарии и адрес.

Наиболее важными элементами учетной записи пользователя являются:

1. список привилегий пользователя в отношении данной системы,
2. список групп, в которых состоит пользователь,
3. идентификатор безопасности **SID** (Security IDentifier).

Идентификаторы безопасности генерируются при создании учетной записи. Они (а не имена пользователей, которые могут не быть уникальными) служат основой для идентификации субъектов внутренними процессами ОС Windows.

Учетные записи групп, созданные для упрощения администрирования, содержат список учетных записей пользователей, а также включают сведения, аналогичные сведениям учетной записи пользователя (SID группы, привилегии члена группы и др.).

SID пользователя (и группы) является уникальным внутренним идентификатором и представляют собой структуру переменной длины с коротким заголовком, за которым следует длинное случайное число. Это числовое значение формируется из ряда параметров, причем утверждается, что вероятность появления двух одинаковых SID практически равна нулю. В частности, если удалить пользователя в системе, а затем создать его под тем же именем, то SID вновь созданного пользователя будет уже другим.

Узнать свой идентификатор безопасности пользователь легко может при помощи утилит *whoami* или *getsid* из ресурсов Windows.

Система хранит идентификаторы безопасности в бинарной форме, однако существует и текстовая форма представления SID. Текстовая форма используется для вывода текущего значения SID, а также для интерактивного ввода (например, в реестр).

В текстовой форме каждый идентификатор безопасности имеет определенный формат. Вначале находится префикс S, за которым следует группа чисел, разделенных дефисами. Например, SID администратора системы имеет вид: *S-1-5-<домен>-500*, а SID группы everyone, в которую входят все пользователи, включая анонимных и гостей - *S-1-1-0*.

19.1.3 Объекты. Дескриптор защиты

В ОС Windows все типы объектов защищены одинаковым образом. С каждым объектом связан ***дескриптор защиты*** *(security descriptor)*. Связь объекта с дескриптором происходит в момент создания объекта.

Дескриптор защиты (см. рис. 19.1) содержит

- SID владельца объекта,
- SID групп для данного объекта
- два указателя на списки DACL (Discretionary ACL) и SACL (System ACL) контроля доступа.

DACL и SACL содержат разрешающие и запрещающие доступ списки пользователей и групп, а также списки пользователей, чьи попытки доступа к данному объекту подлежат аудиту.

Структура каждого ACL списка проста. Это набор записей ACE (Access Control Entry), каждая запись содержит SID и перечень прав, предоставленных субъекту с этим SID.

На примере, изображенном на рис. 19.1, владелец файла Александр имеет право на все операции с данным файлом, всем остальным обычно дается только право на чтение, а Павлу запрещены все операции. Таким образом, ***список DACL*** *описывает все права доступа к объекту*. Если этого списка нет, то все пользователи имеют все права; если этот список существует, но он пустой, права имеет только его владелец.

Рис. 19.1 - Структура дескриптора защиты для файла

В списке ACL есть записи ACE двух типов:

- *разрешающие доступ.* Разрешающая запись содержит SID пользователя или группы и битовый массив (access mask), определяющий набор операций, которые процессы, запускаемые этим пользователем, могут выполнять с данным объектом.
- *запрещающие доступ*. Запрещающая запись действует аналогично, но в этом случае процесс не может выполнять перечисленные операции.

Кроме списка DACL дескриптор защиты включает также список SASL, который имеет такую же структуру, что и DACL, то есть состоит из таких же ACE записей, только вместо операций, регламентирующих доступ к объекту, в нем перечислены операции, подлежащие аудиту.

В примере на рис. 19.1 операции с файлом процессов, запускаемых пользователем Сергеем, описанные в соответствующем битовом массиве будут регистрироваться в системном журнале.

19.1.4 Субъекты безопасности. Процессы, потоки. Маркер доступа

Так же как и объекты, субъекты должны иметь отличительные признаки - контекст пользователя, для того, чтобы система могла контролировать их действия. Сведения о контексте пользователя хранятся в маркере (употребляются также термины «токен», «жетон») доступа.

При интерактивном входе в систему пользователь обычно вводит свое имя и пароль. Система (процедура Winlogon) по имени находит соответствующую учетную запись, извлекает из нее необходимую информацию о пользователе, формирует список привилегий,

ассоциированных с пользователем и его группами, и все это объединяет в структуру данных, которая называется маркером доступа.

Маркер также хранит некоторые параметры сессии, например, время окончания действия маркера. Таким образом, именно маркер является той визитной карточкой, которую субъект должен предъявить, чтобы осуществить доступ к какому-либо объекту.

Основные компоненты маркера доступы показаны на рис. 19.2.

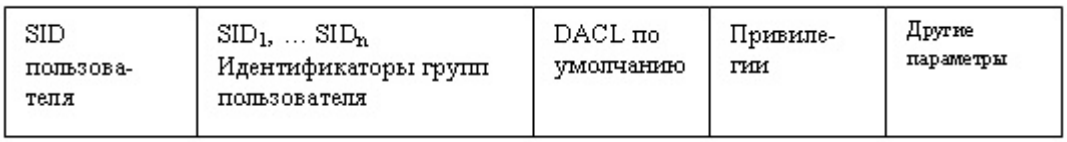


Рис. 19.2 - Основные компоненты маркера доступа

Включая в маркер информацию о защите, в частности, DACL, Windows упрощает создание объектов со стандартными атрибутами защиты. Как уже говорилось, если процесс не позаботится о том, чтобы явным образом указать атрибуты безопасности объекта, на основании списка DACL, присутствующего в маркере, будут сформированы права доступа к объекту по умолчанию.

19.1.5 Проверка прав доступа

После формализации атрибутов защиты субъектов и объектов можно перечислить основные этапы проверки прав доступа см. рис. 19.3.

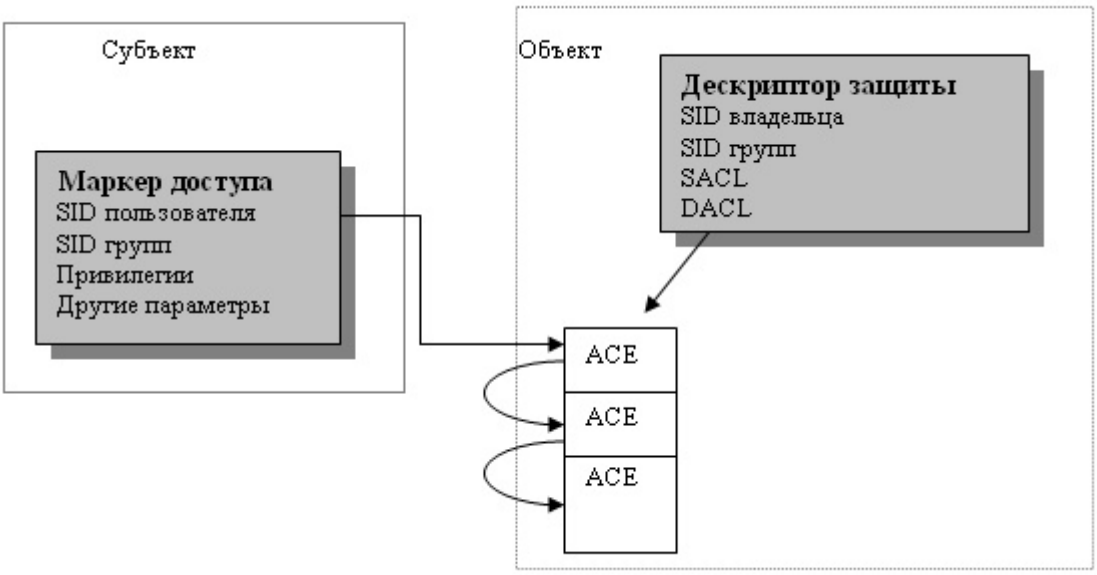


Рис. 19.3. - Пример проверки прав доступа к защищенному объекту

Этапов проверки довольно много. Наиболее важные этапы из них:

- Если SID субъекта совпадает с SID владельца объекта и запрашиваются стандартные права доступа, то доступ предоставляется независимо от содержимого DACL.
- Далее система последовательно сравнивает SID каждого ACE из DACL с SID маркера. Если обнаруживается соответствие, выполняется сравнение маски доступа с проверяемыми правами. Для запрещающих ACE даже при частичном совпадении прав доступ немедленно отклоняется. Для успешной проверки разрешающих элементов необходимо совпадение всех прав.

Очевидно, что для процедуры проверки важен порядок расположения ACE в DACL. Поэтому Microsoft предлагает так называемый предпочтительный порядок размещения ACE. Например, для ускорения рекомендуется размещать запрещающие элементы перед разрешающими.

19.2 Основные компоненты системы безопасности

Система контроля дискреционного доступа - центральная концепция защиты ОС Windows, однако перечень задач, решаемых для обеспечения безопасности, этим не исчерпывается. В данном разделе будут проанализированы структура, политика безопасности и API системы защиты.

Изучение структуры системы защиты помогает понять особенности ее функционирования. Несмотря на слабую документированность ОС Windows по косвенным источникам можно судить об особенностях ее функционирования.

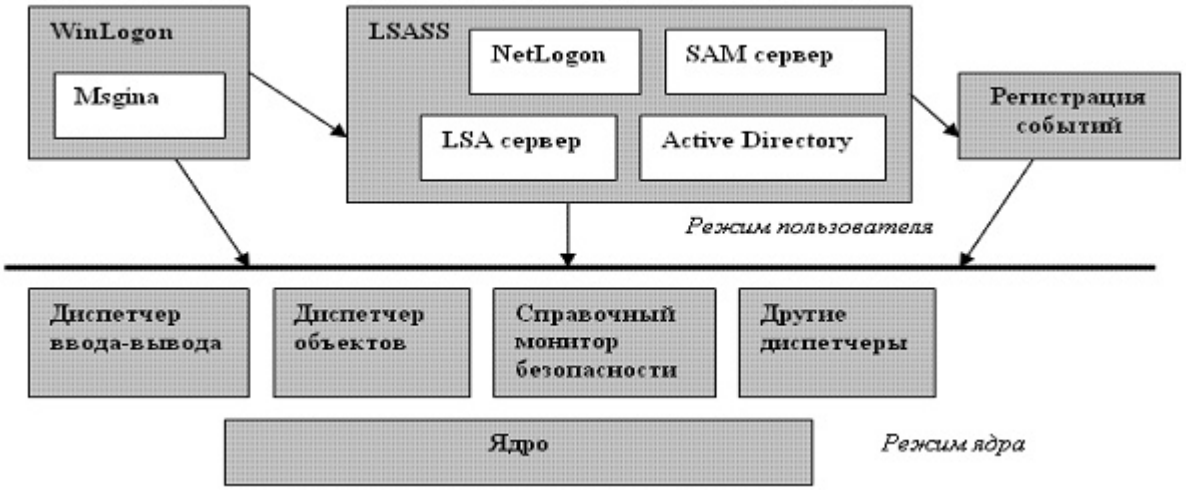


Рис. 19.4 - Структура системы безопасности ОС Windows

Система защиты ОС Windows состоит из следующих компонентов (см. рис. 14.1).

- **Процедура регистрации (Logon Processes),** которая обрабатывает запросы пользователей на вход в систему. Она включают в себя

начальную интерактивную процедуру, отображающую начальный диалог с пользователем на экране, и удаленные процедуры входа, которые позволяют удаленным пользователям получить доступ с рабочей станции сети к серверным процессам Windows NT. Процесс *Winlogon* реализован в файле Winlogon.exe и выполняется как процесс пользовательского режима. Стандартная библиотека аутентификации *Gina* реализована в файле *Msgina.dll*.

- **Подсистема локальной авторизации (Local Security Authority, LSA)**, которая гарантирует, что пользователь имеет разрешение на доступ в систему. Этот компонент - центральный для системы защиты Windows NT. Он порождает маркеры доступа, управляет локальной политикой безопасности и предоставляет интерактивным пользователям аутентификационные услуги. LSA также контролирует политику аудита и ведет журнал, в котором сохраняются сообщения, порождаемые диспетчером доступа. Основная часть функциональности реализована в *Lsasrv.dll*.
- **Менеджер учета (Security Account Manager, SAM)**, который управляет базой данных учета пользователей. Эта база данных содержит информацию обо всех пользователях и группах пользователей. Данная служба реализована в *Samsrv.dll* и выполняется в процессе *Lsass*.
- **Диспетчер доступа (Security Reference Monitor, SRM)**, проверяющий, имеет ли пользователь право на доступ к объекту и на выполнение тех действий, которые он пытается совершить. Этот компонент обеспечивает легализацию доступа и политику аудита, определяемые LSA. Он предоставляет услуги для программ супервизорного и пользовательского режимов, чтобы гарантировать, что пользователи и процессы, осуществляющие попытки доступа к объекту, имеют необходимые права. Данный компонент также порождает сообщения службы аудита, когда это необходимо. Это компонент исполнительной системы: Ntoskrnl.exe.

Все компоненты активно используют базу данных Lsass, содержащую параметры политики безопасности локальной системы, которая хранится в разделе HKLM\SECURITY реестра.

Реализация модели дискреционного контроля доступа связана с наличием в системе одного из ее важнейших компонентов - монитора безопасности.

Монитор безопасности *- особый вид субъекта, который активизируется при каждом доступе и в состоянии отличить легальный доступ от нелегального и не допустить последний.* Монитор безопасности входит в состав диспетчера доступа (SRM), который, согласно описанию, обеспечивает также управление ролевым и привилегированным доступом.

19.2.1 Политика безопасности

Система безопасности ОС Windows отвечает требованиям класса С2 «оранжевой» книги и требованиям стандарта Common Criteria, которые составляют основу *политики безопасности системы*.

Политика безопасности подразумевает ответы на следующие вопросы:

- какую информацию защищать,
- какого рода атаки на безопасность системы могут быть предприняты,
- какие средства использовать для защиты каждого вида информации.

Требования, предъявляемые к системе защиты.

- Каждый пользователь должен быть идентифицирован уникальным входным именем и паролем для входа в систему. Доступ к компьютеру предоставляется лишь после аутентификации. Должны быть предприняты меры предосторожности против попытки против применения фальшивой программы регистрации (механизм безопасной регистрации).
- Система должна быть в состоянии использовать уникальные идентификаторы пользователей, чтобы следить за их действиями. Владелец ресурса (например, файла) должен иметь возможность контролировать доступ к этому ресурсу.
- Управление доверительными отношениями. Необходима поддержка наборов ролей (различных типов учетных записей). Кроме того, в системе должны быть средства для управления привилегированным доступом.
- ОС должна защищать объекты от повторного использования. Перед выделением новому пользователю все объекты, включая память и файлы, должны быть проинициализированы.
- Системный администратор должен иметь возможность учета всех событий, относящихся к безопасности (аудит безопасности).
- Система должна защищать себя от внешнего влияния или навязывания, такого, как модификация загруженной системы или системных файлов, хранимых на диске.

Надо отметить, что, в отличие от большинства операционных систем, ОС Windows была изначально спроектирована с учетом требований безопасности, и это является ее несомненным достоинством. Посмотрим теперь, как в рамках данной архитектуры обеспечивается выполнение требований политики безопасности.

19.2.2 Ролевой доступ. Привилегии

С целью гибкого управления системной безопасностью в ОС Windows реализовано **управление доверительными отношениями (trusted facility management)**, которое требует поддержки набора ролей (различных типов

учетных записей) для разных уровней работы в системе. В системе имеется **управление привилегированным доступом**, то есть функции администрирования доступны только одной группе учетных записей - Administrators (Администраторы.).

В соответствии со своей ролью каждый пользователь обладает определенными ***привилегиями*** **и** ***правами*** на выполнение различных операций в отношении системы в целом, например, право на изменение системного времени или право на создание страничного файла. Аналогичные права в отношении конкретных объектов называются ***разрешениями***. И права, и привилегии назначаются администраторами отдельным пользователям или группам как часть настроек безопасности.

Каждая привилегия имеет следующие представления:

- дружественное имя, отображаемое в пользовательском интерфейсе Windows,
- программное имя, используемое приложениями,
- Luid - внутренний номер привилегии в конкретной системе.

Помимо привилегий в Windows имеются близкие к ним права учетных записей.

Важно, что даже администратор системы по умолчанию обладает далеко не всеми привилегиями. Это связано с **принципом предоставления минимума привилегий**. В каждой новой версии ОС Windows, в соответствии с этим принципом, производится ревизия перечня предоставляемых каждой группе пользователей привилегий, и общая тенденция состоит в уменьшении их количества. С другой стороны общее количество привилегий в системе растет, что позволяет проектировать все более гибкие сценарии доступа.

Назначение и отзыв привилегий - прерогатива **локального администратора безопасности** LSA (Local Security Authority), поэтому, чтобы программно назначать и отзывать привилегии, необходимо применять функции LSA.

Локальная политика безопасности системы *означает наличие набора глобальных сведений о защите, например, о том, какие пользователи имеют право на доступ в систему, а также о том, какими они обладают правами.* Поэтому говорят, что каждая система, в рамках которой действует совокупность пользователей, обладающих определенными привилегиями в отношении данной системы, является объектом политики безопасности. Объект политики используется для контроля базы данных LSA. Каждая система имеет только один объект политики, который создается администратором LSA во время загрузки и защищен от несанкционированного доступа со стороны приложений.

Управление привилегиями пользователей включает в себя задачи перечисления, задания, удаления, выключение привилегий и ряд других.

20. ОБЕСПЕЧЕНИЕ БЕЗОПАСНОСТИ В ОПЕРАЦИОННЫХ СИСТЕМАХ WINDOWS 2000/XP И WINDOWS 2003 SERVER

20.1 Настройка системы Windows 2000/XPpro

В Windows 2000 добавлены серьезные функции безопасности к тем возможностям, которые имели место в серверных версиях Windows NT. Приводимые в данном разделе рекомендации адресованы в основном версии операционных систем Windows 2000 однако в части касающиеся обеспечения безопасности могут быть использованы в операционной системе Windows XP Pro. Операционная система Windows XP Home обладает сущетсвенно урезанной функциональностью в области настройки безопасности и ее настройка здесь и далее не рассмаривается.

Windows 2000/XP не является защищенной системой сразу после установки. Имея это в виду, необходимо произвести настройку некоторых параметров для повышения уровня безопасности, прежде чем система будет готова к работе. Параметры конфигурации подразделяются на параметры локальной политики безопасности и параметры конфигурации системы.

20.1.1 Параметры локальной политики безопасности

В Windows 2000 появилось новое средство - графический пользовательский интерфейс редактирования локальной политики. Чтобы запустить эту утилиту, откройте Control Panel / Administrative Tools / Local Security Policy (Панель управления / Администрирование / Локальная политика безопасности) (см. рис. 20.1). Это средство позволяет настраивать политики учетных записей и локальные политики безопасности. Позже мы обсудим конфигурирование учетной записи. Сейчас давайте сконцентрируем внимание на локальных политиках безопасности.

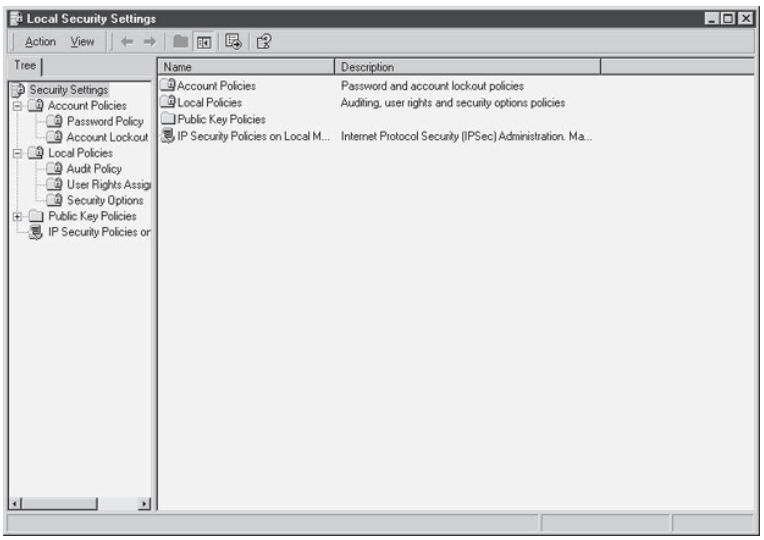


Рис. 20.1 - Графический пользовательский интерфейс управления локальными политиками безопасности

Графический пользовательский интерфейс локальных политик безопасности в действительности является лишь внешней оболочкой процесса внесения изменений в реестр. Следовательно, для внесения изменений в общие параметры реестра больше не требуется использовать программы regedit или regedit32 - лучше использовать утилиту, чем открывать реестр и вносить изменения собственноручно.

На рисунке 20.2 показаны элементы политики безопасности, которые можно настраивать через графический пользовательский интерфейс локальных политик безопасности. В следующих разделах более подробно обсуждаются рекомендуемые изменения для внесения в политику безопасности.

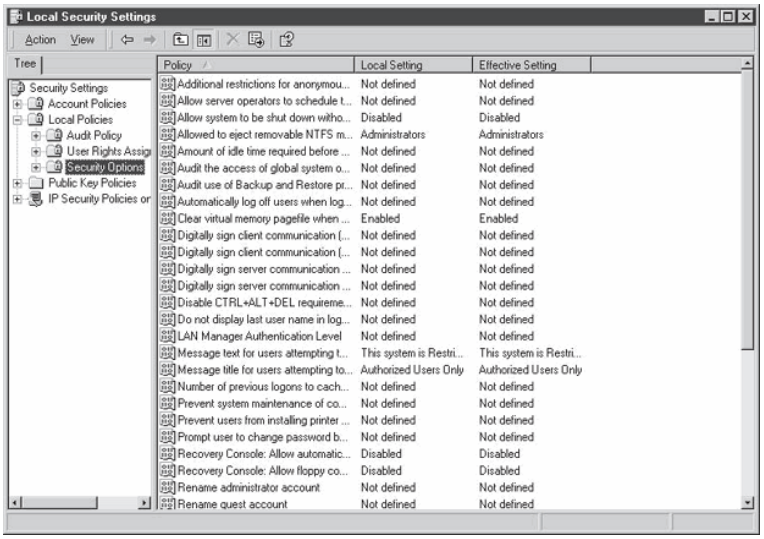


Рис. 20.2 - Настраиваемые элементы локальной политики безопасности

Windows 2000 содержит набор шаблонов, которые используются для конфигурации системы, настройки локальной политики безопасности и параметров управления пользователями в системе. Если вы будете

использовать один из этих шаблонов, убедитесь, что вам понятны изменения, которые будут внесены в систему.

20.1.1.1 Сообщение входа

В Windows 2000 имеются два параметра, настраивающие сообщение входа, отображаемое пользователям:

1. Message Text for Users Attempting to Log On (Текст сообщения для пользователей, пытающихся осуществить вход);
2. Message Title for Users Attempting to Log On (Название сообщения для пользователей, пытающихся осуществить вход).

20.1.1.2 Очистка файла виртуальной памяти при отключении системы

Страничный файл виртуальной памяти содержит важную системную информацию во время работы системы, такую как ключи шифрования или пароли. Чтобы Windows 2000 очищала системный страничный файл при отключении системы, включите параметр Clear Virtual Memory Pagefile When System Shuts Down (Очистить файл подкачки при отключении системы).

20.1.1.3 Разрешение отключения системы без осуществления входа

Пользователи не должны иметь возможность отключать системы, если они не могут осуществлять вход. Следовательно, опция Allow System to Be Shut Down Without Having to Log On (Разрешить отключение системы без входа) должна быть отключена.

20.1.1.4 Уровень аутентификации LAN Manager

Аутентификация LAN Manager - это система аутентификации, позволяющая серверам Windows 2000 работать с клиентами Windows 95 и Windows 98 (а также Windows для рабочих групп). Схемы аутентификации LAN Manager значительно более слабы, нежели система аутентификации NT или Windows 2000 (которая называется NTLM v2) и, таким образом, могут позволить злоумышленнику произвести атаку грубой силой на зашифрованные пароли с использованием гораздо меньших вычислительных мощностей. Чтобы в принудительном порядке использовать аутентификацию NTLM v2, примените следующие параметры.

1. Выберите параметр политики LAN Manager Authentication Level (Уровень аутентификации LAN Manager).
2. Выберите соответствующий уровень в ниспадающем меню.

Устанавливаемое значение зависит от рассматриваемой среды. Существуют шесть уровней:

1. Send LM and NTLM Responses (Отправлять ответы LM и NTLM). Это уровень по умолчанию. Происходит отправка обоих ответов - LAN

Manager и NTLM. В системе никогда не будет использоваться защита сеанса NTLM v2;

2. Send LM и NTLM, Use NTLM v2 If Negotiated (Отправка LM и NTLM, использование NTLM v2 при согласии);
3. Send NTLM Response Only (Отправлять только ответ NTLM);
4. Send NTLM v2 Response Only (Отправлять только ответ NTLM v2);
5. Send NTLM v2 Response Only, Refuse LM (Отправка только ответа NTLM v2, отклонение LM);
6. Send NTLM v2 Response Only, Refuse LM and NTLM (Отправка только ответа NTLM v2, отклонение LM и NTLM).

Перед тем как вносить изменение в эти настройки политики, определите функциональные требования для рассматриваемой сети. Если в сети установлены клиенты Windows 95 или Windows 98, необходимо разрешить ответы LAN Manager.

20.1.1.5 Дополнительные ограничения для анонимных соединений

Этот параметр политики позволяет администратору определить, какие действия разрешены для выполнения через анонимное соединение. Он имеет три опции:

- None, Rely On Default Permissions (Нет ограничений, использовать разрешения по умолчанию);
- Do Not Allow Enumeration of SAM Accounts and Shares (Запретить перечисление учетных записей и общих местоположений в SAM);
- No Access Without Explicit Anonymous Permissions (Запретить доступ без отдельных разрешений на анонимный доступ).

Эти параметры могут предотвратить получение доступа к информации о пользователях системы при работе в недействительных пользователей через недействительные сеансы.

20.1.2 Настройка конфигурации системы

В Windows 2000 включены новые функции безопасности, однако следует понимать, в чем заключаются преимущества и недостатки каждой новой возможности. Сушествует возможность конфигурирования следующих основных групп настроек системы Windows 2000:

- файловые системы;
- параметры сети;
- параметры учетных записей;
- сервис-пакеты и «горячие» обновления.

Политика безопасности организации в обязательном порядке должна предусматривать определенные параметры и требования к конфигурации системы.

20.1.2.1 Файловая система NTFS

Все файловые системы в Windows 2000/XP должны быть преобразованы в NTFS. Так как файловые системы FAT не позволяют использовать разрешения файлов, NTFS лучше с точки зрения безопасности. Если какая-либо из имеющихся файловых систем является системой FAT, можно использовать программу CONVERT, чтобы сменить их на NTFS. Эта программа требует перезагрузки, однако ее можно выполнить с уже имеющейся информацией на диске.

Также следует заметить, что Windows 2000 поставляется с NTFS-5 которая содержит новый набор индивидуальных разрешений:

- проход по папке/выполнение файла;
- просмотр папки/чтение данных;
- чтение атрибутов;
- чтение расширенных атрибутов;
- создание файлов/запись данных;
- создание папок/присоединение данных;
- запись атрибутов;
- запись расширенных атрибутов;
- удаление подпапок и файлов;
- удаление;
- чтение разрешений;
- изменение разрешений;
- присвоение прав владения.

20.1.2.2 Шифрующая файловая система EFS

Одним из недостатков файловой системы NTFS является то, что она защищает файлы только тогда, когда используется с Windows NT или Windows 2000. Если злоумышленник загрузит систему с использованием другой операционной системы (например, DOS), он сможет использовать программу (такую как NTFSDOS) для чтения файлов и, таким образом, обойдет элементы управления доступом NTFS. В Windows 2000 введена файловая система для защиты секретных файлов от атак данного типа.

Шифрующая файловая система (Encrypting File System - EFS) реализована таким образом, чтобы быть незаметной для пользователя. Следовательно, пользователю не требуется инициировать дешифрование или шифрование файла (после применения EFS для файла или каталога). Чтобы активизировать EFS, выберите файл или каталог, который нужно защищать, щелкните правой кнопкой на этом элементе и выберите Properties (Свойства). Нажмите кнопку Advanced (Дополнительно) в окне General (Общие) и выберите Encrypt Contents to Secure Data (Шифровать содержимое для защиты данных).

Когда для файла назначено шифрование, система выбирает ключ для использования в алгоритме симметричного шифрования и шифрует данный

файл. После этого ключ шифруется с использованием открытого ключа одного или нескольких пользователей, которые будут иметь доступ к файлу. Следует заметить, что EFS имеет встроенный механизм, позволяющий осуществлять восстановление зашифрованной информации. По умолчанию из локальной учетной записи администратора всегда можно расшифровать любые файлы EFS.

В зависимости от способа взаимодействия EFS с пользователем и операционными системами некоторые команды будут приводить к расшифровыванию файлов, а другие - нет. Например, команда Ntbackup копирует зашифрованный файл в том виде, в каком он есть. Однако если пользователь выполнит команду Copy, файл будет расшифрован и перезаписан на диск. Если конечным расположением файла является раздел, отличный от NTFS 5.0, или гибкий диск, то файл не будет шифроваться при записи. Кроме того, если файл копируется на другой компьютер, он будет шифроваться заново с использованием другого ключа симметричного алгоритма. Два файла будут выглядеть различным образом на двух компьютерах, даже если их содержимое идентично.

20.1.2.3 Общие местоположения

Windows 2000 создает административные общие местоположения при загрузке. Ими являются C$, D$, IPC$, ADMIN$ и NETLOGON (имеются только на контроллерах доменов). Полный список текущих общих местоположений можно просмотреть в утилите Computer Management (Управление компьютером), выбрав в панели управления значок Administrative Tools (Администрирование) (см. рис. 20.3).

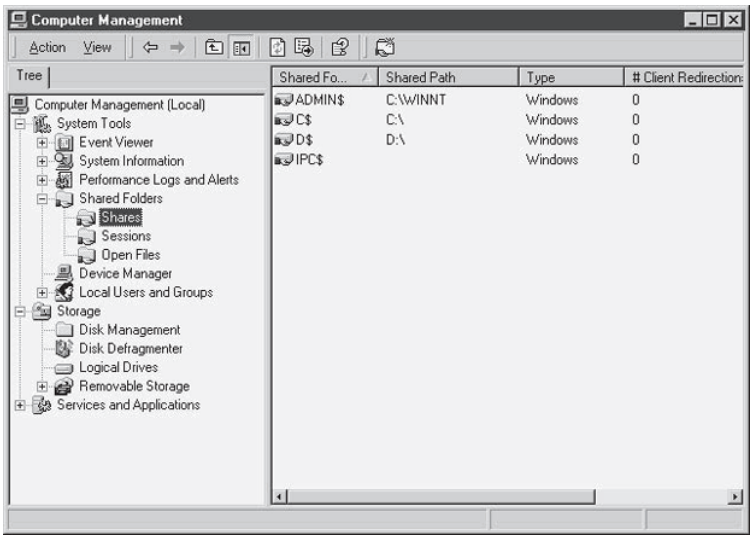


Рис. 20.3 - Имеющиеся общие местоположения, отображаемые в оснастке Computer Management (Управление компьютером)

Несмотря на то, что эти общие местоположения могут использоваться злоумышленниками для осуществления попыток раскрытия пароля

администратора посредством грубой силы, отключать какие-либо из них не рекомендуется.

20.1.2.4 Сеть

В дополнение к стандартным портам Windows (135, 137 и 139) в Windows 2000 используется порт 88 для **Kerberos**, порт 445 для **SMB** через **IP**, порт 464 для **Kerberos kpasswd** и порт 500 (только UDP) для **Internet Key Exchange (IKE)**. Это означает, что если требуется удалить NetBIOS из системы Windows 2000, то вам потребуется отключить опцию File and Print Sharing for Microsoft Networks (Совместный доступ к файлам и принтерам в сетях Microsoft) в данном конкретном интерфейсе. Это можно сделать в окне Network and Dial-up Connections (Сеть и удаленный доступ**)**. Выберите меню Advanced (Дополнительно) и затем выберите Advanced Settings (Дополнительные параметры), чтобы открыть вкладку Adapters and Bindings (Адаптеры и компоненты) - см. рис. 20.4.

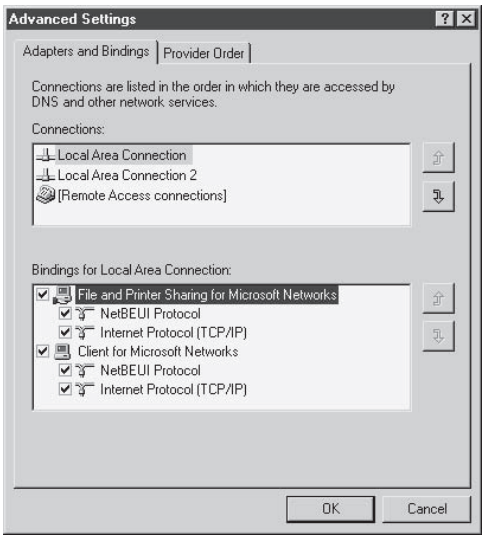


Рис. 20.4 - Удаление компонентов для NetBIOS

Сеть по-прежнему является ключевой частью Windows 2000. Домены Windows 2000 поддерживают централизованное управление пользовательской базой данных. Однако структура **Active Directory** не позволяет использовать иерархическую концепцию. Это означает, что группы могут создаваться над или под другими группами, и домен может быть поделен на организационные единицы с локальным управлением.

Перед развертыванием Windows 2000 или 2003 в организации необходимо четко спланировать структуру доменов.

20.1.2.5 Параметры учетных записей

В Windows 2000 имеются две учетные записи по умолчанию:

- Administrator (Администратор),

- Guest (Гость).

Обе учетные записи можно переименовать с помощью утилиты Local Security Settings (Локальные параметры безопасности). Выберите элементы политики Rename Administrator Account (Переименование учетной записи администратора) и Rename Guest Account (Переименование гостевой учетной записи), чтобы внести изменения. Гостевая учетная запись также должна быть отключена. На всякий случай рекомендуется сменить пароль гостевой учетной записи, указав очень длинный пароль со случайным набором символов.

Каждая рабочая станция и сервер Windows 2000 в организации будут содержать учетную запись Administrator (Администратор), являющуюся локальной по отношению к данному компьютеру и требующую соответствующей защиты. Для защиты этих учетных записей необходимо разработать процедуру создания очень надежного пароля. Пароль должен быть записан, заклеен в конверт и положен на хранение в запираемом кабинете.

Политики паролей и блокировки, описываемые в следующих разделах, могут быть применены посредством оснасток:

- Group Policies (Групповые политики),
- Active Directory.

20.1.2.6 Политика паролей

Политика системных паролей определяется с помощью средства Local Security Settings (Локальные параметры безопасности) - см. рис. 20.5. В этом окне настраиваются параметры паролей и требования к их надежности. Как в случае с любой компьютерной системой, эти параметры должны настраиваться в соответствии с политикой безопасности организации.

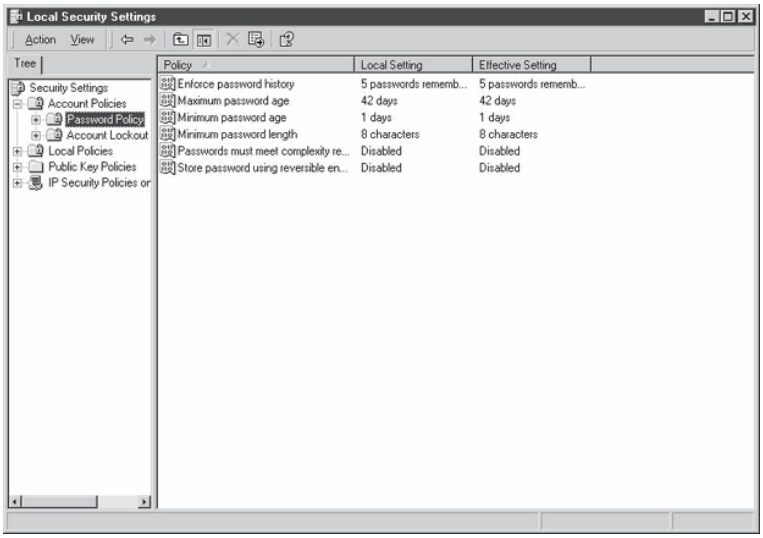


Рис. 20.5 - Использование средства Local Security Settings (Локальные параметры безопасности) для настройки политики паролей

Если включить параметр Passwords Must Meet Complexity Requirements (Пароли должны отвечать требованиям сложности), то будет применен фильтр паролей, установленный по умолчанию (PASSFILT.DLL). Этот фильтр требует, чтобы длина всех паролей составляла не менее шести символов, чтобы пароли не содержали частей имени пользователя и содержали, по крайней мере, какие-либо из следующих элементов: цифры, символы, строчные или прописные буквы.

За исключением случая крайней необходимости не следует включать параметр Store Passwords Using Reversible Encryption (Сохранять пароли с использованием обратимого шифрования).

20.1.2.7 Политика блокировки учетных записей.

Политика блокировки учетных записей также настраивается с использованием средства Local Security Settings (Локальные параметры безопасности) - см. рис. 20.6. Эти параметры должны настраиваться в соответствии с политикой безопасности организации.

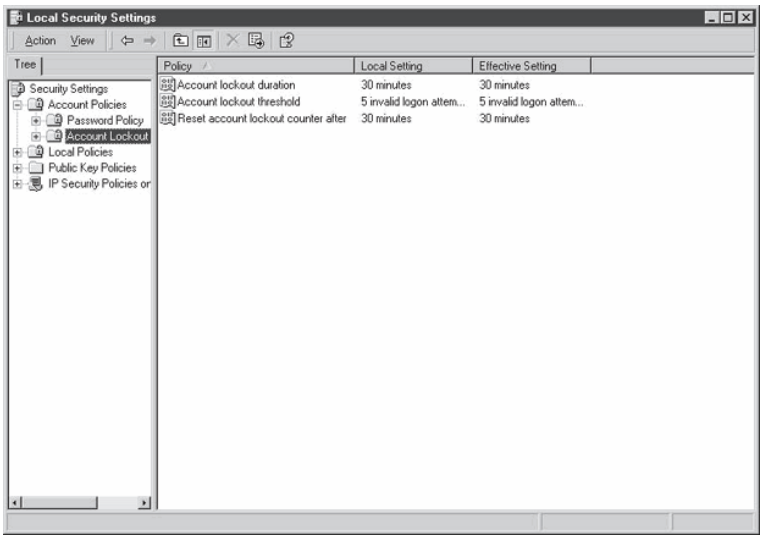


Рис. 20.6 - Использование средства Local Security Settings (Локальные параметры безопасности) для настройки политики блокировки

Политика блокировки учетных записей предназначена для предотвращения атак «грубой силы», направленных на угадывание паролей. Данная возможность также используется для создания условия отказа в обслуживании по отношению ко всему сообществу пользователей. Следовательно, следует принимать во внимания возможные последствия продолжительных блокировок пользователей при настройке данной политики.

Блокировка не распространяется в принудительном порядке на учетную запись администратора. Учетная запись Administrator (Администратор) всегда доступна для входа в систему из системной консоли.

20.1.2.8 Сервис-пакеты и обновления

Сервис-пакеты и горячие обновления для Windows 2000 должны устанавливаться в сети организации после соответствующего тестирования. Своевременная установка сервис-паков позволит устранять найденные уязвимости и гарантировать безопасность системы.

20.2 Особенности настройки Windows 2003 Server

Изначально процесс установки системы идентичен установке Windows 2000. Однако имеются три задачи по настройке, которые необходимо правильно выполнить после установки системы:

1. ограничения на программное обеспечение;
2. служба Terminal Services;
3. настройка Framework .NET.

20.2.1 Политики ограничения программного обеспечения

Единственным отличием локальных политик безопасности Windows 2003 Server и Windows 2000 являются политики ограничения программного обеспечения (Software Restriction Policies) - см. рис. 20.7. Политики ограничения программного обеспечения позволяют осуществлять контроль над тем, какие программы могут выполняться на данном локальном компьютере. Преимуществом этой возможности является то, что администратор может указывать, выполнение каких программ разрешено в системе, и, таким образом, предотвращать выполнение программ, не пользующихся доверием.

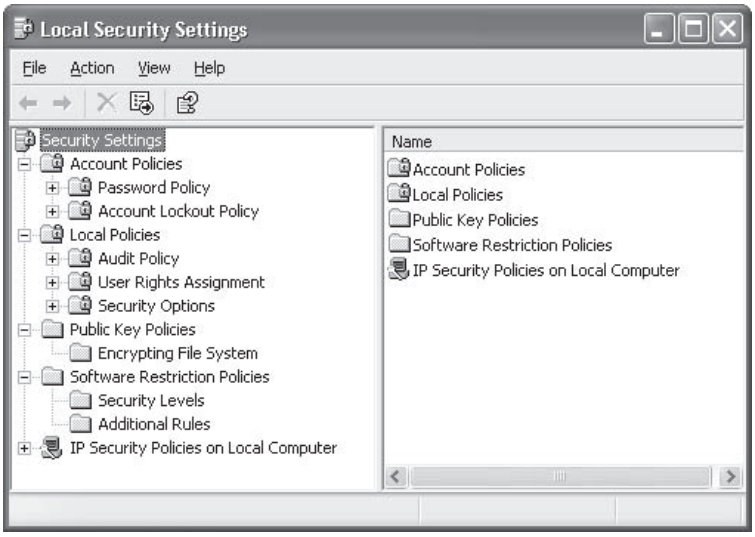


Рис. 20.7 - Политика ограничения программного обеспечения для локальной системы

Вы можете определить уровень безопасности по умолчанию Unrestricted (Без ограничений) (разрешить все, что не запрещено) или Disallowed (Запрещено)

(Запретить все, что не разрешено). Последний вариант лучше с точки зрения безопасности, однако при его использовании могут возникнуть проблемы из-за того, что этот уровень окажется слишком ограничительным. Настоятельно рекомендуется потратить время на то, чтобы проверить эти настройки на тестовой системе, перед тем как применять их на работающих системах.

После установки уровня по умолчанию можно указать исключения в данном уровне безопасности посредством создания правил политики ограничения программного обеспечения для конкретных программ. Исключения могут быть указаны на основе программного обеспечения:

1. хеши;
2. сертификаты;
3. пути (включая путь реестра);
4. зоны интернета.

Некоторые примеры действий, которые можно реализовать посредством политик ограничения программ:

- запрет на запуск определенных типов файлов в каталоге вложений электронной почты используемой почтовой программы;
- ограничение того, какие программы могут запускаться пользователями на серверах терминала.

При этом политики ограничения программ не должны использоваться вместо антивирусного программного обеспечения.

20.2.2 Службы терминала (Terminal Services)

По умолчанию система Windows 2003 Server содержит функцию Remote Desktop for Administration (Удаленный рабочий стол для администрирования) (Terminal Services в режиме Remote Administration [удаленное администрирование] в Windows 2000). Она позволяет создавать до двух удаленных сеансов плюс сеанс консоли. Так как эта возможность разрешает пользователям удаленно управлять серверами с любого клиента сети, необходимо обеспечить ее защиту от несанкционированного использования. Чтобы обеспечить максимальный уровень безопасности, необходимо убедиться в наличии следующих параметров, настраиваемых с помощью опции Properties (Свойства) для конкретного соединения в оснастке Terminal Services Configuration (Настройка службы терминала) - рис. 20.8.

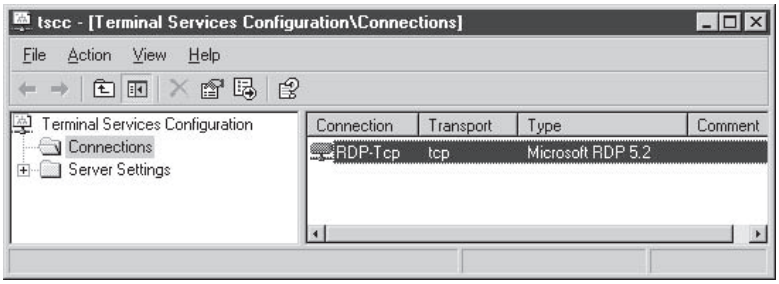


Рис. 20.8 - Настройка службы Terminal Services

Уровень шифрования. В параметре Encryption Level (Уровень шифрования) приводится перечень доступных уровней, используемых для защиты данных, передаваемых между клиентом и сервером. Здесь имеются четыре опции:

1. Low (Низкий). Данные шифруются с использованием 56-битного ключа.
2. Client Compatible (Совместимый с клиентом). Данные шифруются с использованием ключа максимальной длины, поддерживаемого клиентом.
3. High (Высокий). Данные шифруются с использованием 128-битного шифрования. Клиенты, не поддерживающие этот уровень шифрования, не будут иметь возможность подключения (рекомендуется использовать эту опцию).
4. FIPS Compliant (Соответствие FIPS). Данные шифруются в соответствии со стандартом Federal Information Processing Standard 140-1, определяющим соответствующие методы шифрования.

Logon Settings (Параметры входа в систему). Здесь можно указать аутентификационные данные для использования по умолчанию при подключении клиентов к серверу терминала (см. рис. 20.9). По умолчанию используются аутентификационные данные, предоставляемые клиентом. Другая опция позволяет использовать одну учетную запись пользователя для всех соединений. Последняя опция требует от пользователя ввода пароля, даже если предоставлены аутентификационные данные.

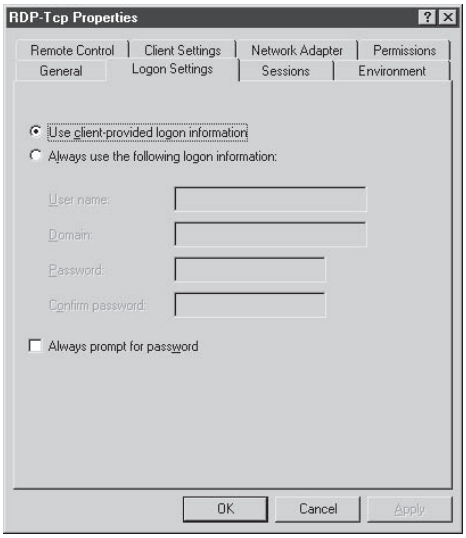


Рис. 20.9 - Вкладка Logon Settings (Параметры входа в систему).

Network Adapter settings (Параметры сетевого адаптера)**.** С помощью этой опции можно определить, какие сетевые адаптеры будет использовать служба. Это относится только к системам с несколькими сетевыми адаптерами.

При правильном администрировании учетных записей пользователей (посредством надежных паролей, блокировки и т. д.) и правильной защите

системы (с использованием межсетевых экранов) данная служба будет относительно защищена.

20.2.3 Настройка средства Framework .NET

Средство .NET Framework Configuration (см. рис. 20.10) позволяет настраивать политику безопасности доступа к коду специально для **Framework .NET**. В данной утилите есть возможность обеспечения защиты и/или удаления управляемых компонентов, установленных на рассматриваемом компьютере. С точки зрения безопасности данное средство может использоваться для контроля за доступом приложений к защищенным ресурсам. Система безопасности использует три уровня политики:

1. Enterprise (Предприятие). Политика безопасности для предприятия в целом. Следует иметь в виду, что нет четких границ между данным уровнем и политикой Machine (Компьютер), так как обе эти политики применяются к каждому компьютеру.
2. Machine (Компьютер). Применяется ко всем программам, выполняемым на данном компьютере.
3. User (Пользователь). Применяется к пользователю, работающему в системе в данный момент.

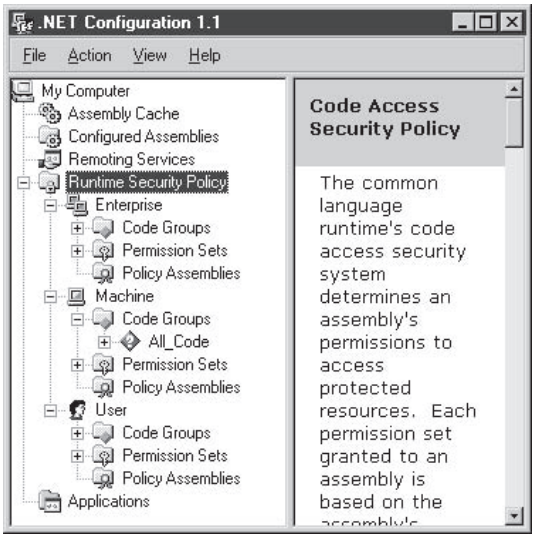


Рис. 20.10 - Утилита настройки .NET

Оценка политик осуществляется в отдельном порядке, и программам предоставляется минимальный набор разрешений, обуславливаемый комбинацией политик. Любой «запрет» имеет преимущество перед «разрешением».

Для получения более подробной информации о модели безопасности программного доступа, необходимо обратится к документации Microsoft .NET Framework SDK.

20.3 Управление пользователями

Управление пользователями в системе Windows 2000/XP является очень важным аспектом безопасности системы и организации в целом. В организации необходимо наличие корректных процедур по определению полномочий каждого нового пользователя. При увольнении сотрудника из организации также необходимо применять соответствующие процедуры, чтобы обеспечить запрет доступа увольняемого сотрудника к системам организации.

20.3.1 Добавление пользователей в систему

При добавлении новых пользователей в систему необходимо следовать процедурам управления пользователями. Эти процедуры должны определять, кто может запрашивать новые учетные записи, и кто может одобрять эти запросы. Новые пользователи добавляются в систему или домен через оснастку Computer Management (Управление компьютером). Выберите элемент Users (Пользователи) из Local Users and Groups (Локальные пользователи и группы). Затем выберите New User (Новый пользователь) из меню Action (Действие) (см. рис. 20.11). Каждый пользователь должен иметь уникальный пользовательский идентификатор и свою собственную учетную запись. Если двум пользователям требуется доступ одинакового уровня, следует создать две учетные записи и разместить их в одной группе. Ни при каких обстоятельствах нельзя присваивать нескольким пользователям один и тот же идентификатор.

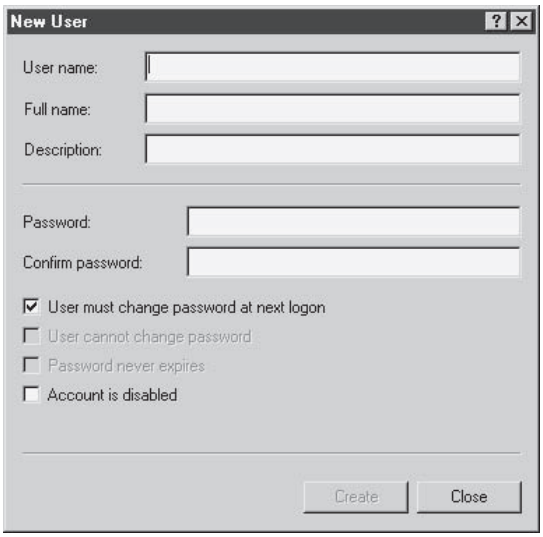


Рис. 20.11 - Окно New User (Новый пользователь)

Для каждого идентификатора нового пользователя следует назначить начальный пароль, а также отметить опцию User Must Change Password (Пользователь должен изменить пароль). Это потребует от пользователя смены

пароля при первом входе в систему. Ни в коем случае не отмечайте опцию Password Never Expires (Срок действия пароля не ограничен).

В организациях не должен использоваться один и тот же пароль для каждой новой учетной записи. Несмотря на то, что таким образом упрощается задача создания новых учетных записей, это обуславливает потенциальную уязвимость систем. Если новая учетная запись создается перед тем, как сотрудник будет принят на работу в организацию, эта запись будет доступна для использования неавторизованными лицами. Все, что им понадобится для доступа - это стандартный пароль новых пользователей. Рекомендуется использовать надежные и уникальные пароли новых пользователей.

Сразу после создания учетной записи ее следует добавить в соответствующие группы. Это можно сделать следующим образом: перейдите к каждой группе по отдельности, дважды щелкните на ней и нажмите кнопку Add (Добавить) (см. рис. 20.12). В качестве альтернативы щелкните правой кнопкой мыши на вновь созданном пользователе и выберите Properties (Свойства). Откройте вкладку Member Of (Член группы) и добавьте в список соответствующие группы (см. рис. 20.13). Стандартные пользовательские учетные записи не должны входить в состав группы Administrator (Администратор).

Рис. 20.12 - Добавление пользователей в группу с помощью списка групп

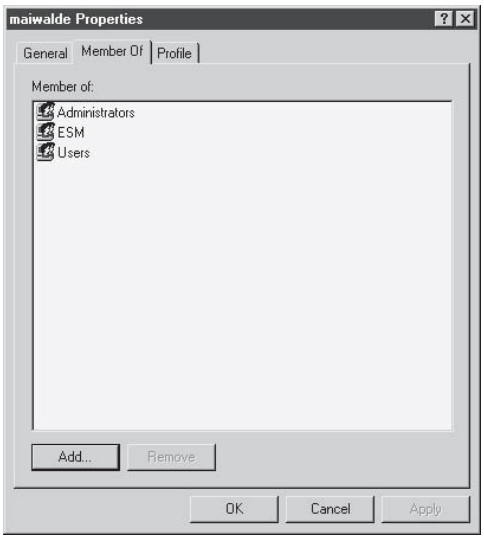


Рис. 20.13 - Добавление пользователей в группы в окне свойств

20.3.2 Настройка файловых разрешений

Для настройки разрешений файлов и общих местоположений следует использовать группы. Это позволит облегчить управление файловыми разрешениями (в отличие от предоставления отдельным пользователям полномочий на доступ к файлам и общим местоположениям). Убедитесь, что

членом группы Guests (Гости) является только учетная запись Guest (Гость), и что учетная запись Guest (Гость) отсутствует во всех остальных группах.

20.3.3 Удаление пользователей из системы

Как и при добавлении пользователей в систему, администраторам необходимо выполнять процедуры по управлению пользователями при удалении пользователей. Когда пользователь покидает организацию, его учетная запись должна немедленно отключаться с помощью утилиты Computer Management (Управление компьютером). Выберите нужного пользователя, щелкните на нем правой кнопкой мыши и выберите Properties (Свойства). Появившееся окно позволит отключить учетную запись. В то же время необходимо изменить пароль на произвольную случайную комбинацию символов. Это предотвратит использование учетной записи пользователем или кем бы то ни было еще.

Бывает так, что рассматриваемый пользователь имел файлы или полномочия, необходимые организации; его учетная запись должна оставаться отключенной в течение некоторого времени (как правило, 30 дней), чтобы начальник пользователя смог получить доступ к этим файлам и скопировать нужные материалы. Если пользователь использовал файловую систему EFS, то для доступа к файлам можно применять локальную учетную запись Administrator (Администратор). По прошествии 30 дней учетная запись должна быть удалена из системы вместе со всеми файлами и каталогами, принадлежащими учетной записи.

В некоторых организациях учетные записи не удаляются и находятся в отключенном состоянии, чтобы выяснить, будет ли кто-нибудь пытаться использовать старую учетную запись. Действия, производимые с учетной записью, обусловливаются процедурами управления пользователями, утвержденными в организации.

20.4 Аудит системы

Все системы Windows 2000 должны подвергаться аудиту. Политика аудита в системе настраивается в утилите Local Security Settings (Локальные параметры безопасности) - см. рис. 20.14. Выберите событие, аудит которого следует производить, и дважды щелкните на нем, чтобы отобразить окно конфигурации.

Политика аудита должна настраиваться в соответствии с политикой безопасности организации. Как правило, рекомендуется фиксировать следующие события:

1. аудит событий входа через учетные записи, успех или неудача;
2. аудит управления учетными записями, успех или неудача;
3. аудит событий входа, успех или неудача;
4. аудит доступа к объектам, неудача;

5. аудит изменения политики, успех или неудача;
6. аудит использования привилегий, неудача;
7. аудит системных событий, успех или неудача.

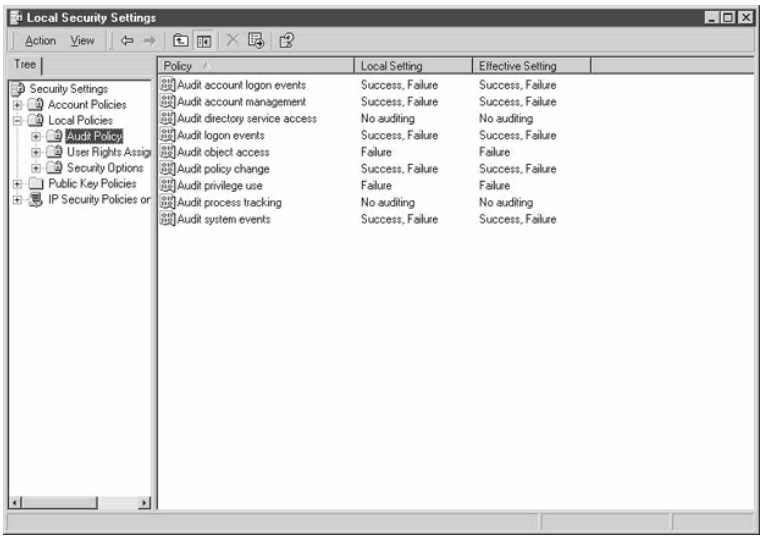


Рис. 20.14 - Настройка политики аудита в системе Windows 2000

При аудите доступа к объектам может генерироваться достаточно большое число записей журнала, даже если включена только опция записи неудачных событий. Тщательно отслеживайте новую систему и убедитесь, что по этой причине не происходит переполнение файлов журналов.

20.4.1 Журнал событий безопасности

Записи журнала аудита в системе Windows 2000 создаются в журнале событий безопасности, который расположен в папке \%systemroot%\system32\config. Разрешения журнала событий безопасности предоставляют доступ только администраторам. Администраторы должны регулярно проверять файлы журналов. Так как записи файлов журналов являются самым лучшим средством выявления неполадок в системе или несанкционированных действий пользователей, то, если администраторы не будут просматривать файлы журналов, смысл фиксирования информации сведется к нулю, в котором рассказывается о признаках подозрительной активности).

Если регулярно производится резервное копирование системы, файлы журнала также должны резервироваться. Если журналы событий нужно сохранять на более длительные периоды времени, рекомендуется периодически перемещать файлы журналов с системы. Файлы можно сохранять в виде текстовых файлов посредством команды Save As (Сохранить как) в меню Action (Действие) в программе Event Viewer (Просмотр событий).

20.4.2 Мониторинг признаков атак

Существует несколько признаков того, что в системе Windows 2000 что-то идет не так, как нужно, и что кто-то пытается выполнить запрещенные действия.

20.4.2.1 Попытки входа в систему

Если кто-либо пытается угадать пароли учетных записей (вручную или с привлечением автоматизированной программы), в журнал событий будут занесены записи, отображающие неудачные попытки входа в систему. Кроме того, если система настроена на блокировку учетных записей после определенного числа попыток входа, будет присутствовать набор заблокированных учетных записей. Сообщения о неудачных попытках входа в журнале событий безопасности содержат имя рабочей станции, с которой осуществлялась каждая попытка. С этой рабочей станции и следует начать выяснение причины неудачных попыток входа в систему. Метод выяснения зависит от источника попыток. Если источник внутренний, следует найти сотрудника, работающего за данной рабочей станцией, и поговорить с ним. Если источник внешний, следует заблокировать на межсетевом экране доступ с IP-адреса источника.

20.4.2.2 Ошибки доступа

Ошибки доступа могут означать, что доступ к секретным файлам пытается получить авторизованный пользователь. Единичные ошибки считаются в порядке вещей. Однако если обнаружится пользователь, совершивший неудачные попытки входа в большое число файлов или каталогов, то у вас появятся все основания для выяснения причин неудачных попыток.

20.4.2.3 Неудачные попытки входа

Информация в журнале событий безопасности содержит перечень неудачных попыток входа. Она не представляет собой доказательства того, что конкретный сотрудник пытался получить несанкционированный доступ к информации. Эти сообщения журнала могут генерироваться процессами, пытающимися осуществить доступ без ведома пользователя; также причиной возникновения этих записей является использование кем-либо учетной записи данного пользователя или его системы. Ни в коем случае не следует считать, что записи в журнале являются достаточным доказательством для того, чтобы обвинить сотрудника в совершении противоправных действий.

20.4.2.4 Отсутствие файлов журналов или пробелы в них

В работающей системе Windows 2000 с включенным аудитом файлы журналов никогда не бывают пусты. Многие злоумышленники очищают файлы журналов сразу после входа в систему в надежде скрыть факт своего присутствия. Если вы обнаружили пустой файл журнала, это говорит о том, что с системой что-то не в порядке, и следует немедленно начать выяснение причин отсутствия в журналах данных. Может оказаться, что другой администратор указал опцию очистки файлов журналов, так как они имели очень большой размер. Однако может выясниться, что в систему кто-то проник несанкционированно.

Не так давно начали выходить в свет утилиты, помогающие злоумышленникам изменять отдельные записи в файлах журналов. В результате этого действия в файле журнала может оказаться пробел. Чтобы обнаружить пробел, просмотрите содержимое файла и выясните, присутствуют ли в нем пропуски, большие, чем обычные. Если обнаружатся значительные пробелы в содержимом файла, следует выяснить причину их появления. Имейте в виду, что система не создает записи в журнале, когда она отключена. В данном случае в содержимом файла перед и после каждого пробела будут присутствовать записи отключения и запуска системы.

20.4.3 Неизвестные процессы

В системах Windows 2000 выполняется множество процессов. Некоторые из них обнаружить легко, другие - сложнее. Если посмотреть в окно программы Task Manager (Диспетчер задач) - см. рис. 20.15, то можно увидеть процессы, выполняющиеся в данный момент в системе, а также процент использования процессора и объем используемой процессами памяти.

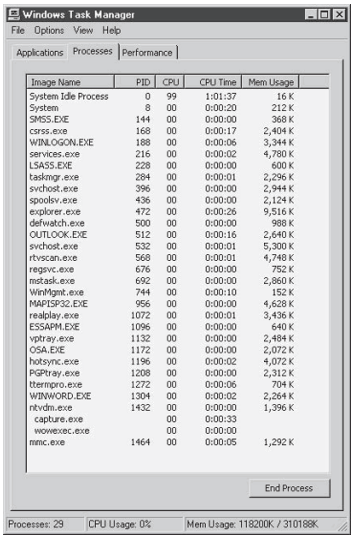


Рис. 20.15 - Диспетчер задач Windows 2000

Системные администраторы должны периодически открывать Диспетчер задач и выяснять, не выполняются ли в системе какие-либо неизвестные процессы. Например, рекомендуется всегда искать процессы CMD. Процесс CMD является сеансом командной строки или окном DOS. Если он работает, то на экране должно отображаться соответствующее окно. В некоторых случаях злоумышленники запускают процесс CMD для выполнения операций в системе. Это явный признак того, что в системе происходит что-то необычное.

21. ИСПОЛЬЗОВАНИЕ СЛУЖБЫ КАТАЛОГОВ И ГРУППОВЫХ ПОЛИТИК В WINDOWS 2000/XP И WINDOWS 2003 SERVER

21.1 Служба каталогов Active Directory

21.1.1 Использование Active Directory

Центральным элементом системы безопасности Windows 2000/2003 является Active Directory (AD).

Active Directory (AD) *- это служба каталогов, являющаяся масштабируемой структурой домена разработанная и внедренная в последние версии операционной системы Windows.*

AD может состоять из одного или более доменов, причем каждый домен имеет свои политики безопасности и безопасные (т. е. доверенные) взаимоотношения с другими доменами.

Пространство имен домена соответствует домену DNS, а домен Root - это первый домен, создаваемый в AD. Все домены в AD совместно используют одну и ту же конфигурацию, схему и глобальный каталог.

Ключевыми компонентами AD и их функциями являются следующие элементы:

1. ***Global Catalog (GC, Глобальный каталог).*** *Серверы GC содержат частичные реплики всех доменов в AD, а также полную реплику схемы и именования конфигурации*, поэтому эти системы являются носителями секретной информации и должны соответствующим образом защищаться.
2. ***Схема.*** *Схема определяет, какие объекты и атрибуты могут храниться в AD.* Она поддерживает все классы объектов и атрибуты, содержащиеся в AD. Для каждого класса объектов схема определяет место в AD для создания класса объекта, а также список атрибутов, которые должен содержать класс. Это ключевой компонент AD, и очень важно обеспечить его безопасность
3. ***Домен.*** *Это группа компьютеров, объединенных для формирования административной ограниченной области пользователей, групп, компьютеров и организационных единиц.*
4. ***Организационная единица (Organizacion Unit - OU)*** *- это тип объектов каталога, с которыми можно связать групповые политики и таким образом определять в них ограничения безопасности.* Это наименьшие административные единицы в AD, формирующие границы защищаемой области. По умолчанию, так как домен является ограниченной областью администрирования, и OU существует только внутри домена, домен является наиболее внешней организационной единицей.

5. ***Групповые политики.*** *Объект домена, обеспечивающий возможность группирования параметров безопасности и конфигурации в шаблоны, которые могут применяться к отдельным системам, доменам или организационным единицам.*
6. ***Доверительные взаимоотношения.*** *Доверительные взаимоотношения позволяют использовать информацию из одного домена, такую как идентификаторы безопасности пользователей, в другом домене.* По умолчанию в AD имеется двустороннее транзитивное доверие. Домены с двусторонним доверием полностью доверяют друг другу. Транзитивное доверие означает, что если домен A доверяет домену B, а домен B доверяет домену C, то домен A доверяет домену C.

21.1.2 Безопасная установка и настройка Active Directory

При настройке AD наиболее важным моментом, связанным с безопасностью, является выбор опции Permissions Compatible with Pre-Windows 2000 Server (Разрешения совместимы с версиями Windows, предшествующими Windows 2000 Server). Эта опция делает группу Everyone (Все) членом встроенной группы Pre-Windows 2000 Compatible Permissions (Разрешения, совместимые с операционными системами, предшествующими Windows 2000). Это позволяет устанавливать анонимные соединения с AD (т. е. предоставляются анонимные полномочия на чтение всем важным пользовательским и групповым атрибутам домена). Если поддержка систем, предшествующих Windows 2000, не требуется, не следует включать эту опцию.

На данном этапе (если вы не упустили какие-либо разрешения) AD должна быть достаточно защищена. Единственное, что осталось сделать, - убедиться, что пользователи используют надежные пароли, и что системы защищены от сетей без доверия (таких как интернет).

21.1.3 Средства администрирования Active Directory

Ниже приведен перечень основных средств, используемых для управления AD, с кратким описанием каждой утилиты.

- Active Directory Domains and Trusts**.** Эта утилита используется для запуска программы Domain Manager (Диспетчер домена), управления доверительными взаимоотношениями, установки режима функционирования и определения альтернативных суффиксов UPN (User Principal Name).
- Active Directory Sites and Services**.** Эта утилита используется для администрирования топологии репликации, добавления и удаления сайтов, переноса компьютеров в сайт, добавления в сайт подсети, связывания сайта с подсетью и создания связи сайта.

- Active Directory Users and Computers**.** Эта утилита используется для управления объектами в домене. С ее помощью осуществляется добавление, перенос, удаление и изменение атрибутов таких объектов AD, как пользователи, группы, компьютеры и общие папки.
- ADSIEdit. Эта оснастка позволяет выполнять LDAP-операции по отношению к любым разделам каталога (домен, конфигурация или схема). ADSIEdit осуществляет доступ к AD через ADSI и позволяет добавлять, удалять и перемещать объекты внутри AD. Также с ее помощью можно просматривать, изменять и удалять атрибуты.

21.1.4 Управление пользователями и группами Active Directory

Необходимо обеспечить правильность настроек безопасности для всех учетных записей. Это можно сделать двумя способами: посредством политики учетной записи через групповую политику в домене с рассматриваемой учетной записью или посредством отдельных ограничений в свойствах пользовательской учетной записи для конкретного объекта User (Пользователь). Политики учетных записей применяются через оснастку Local Security Policy (Локальная политика безопасности**)** или через механизм **Group** Policy (Групповые политики) в домене, в котором находится учетная запись. Свойства учетной записи пользователя устанавливаются для пользователей в индивидуальном порядке. Так как эти параметры специфичны для каждого пользователя, у них нет ничего общего с групповой политикой или локальными параметрами безопасности; они являются атрибутами объекта User. С помощью оснастки Active Directory Users and Computers (Пользователи и компьютеры Active Directory) можно осуществлять администрирование пользователей домена, а посредством оснастки Local Users and Groups (Локальные пользователи и группы) - администрирование локальных пользователей.

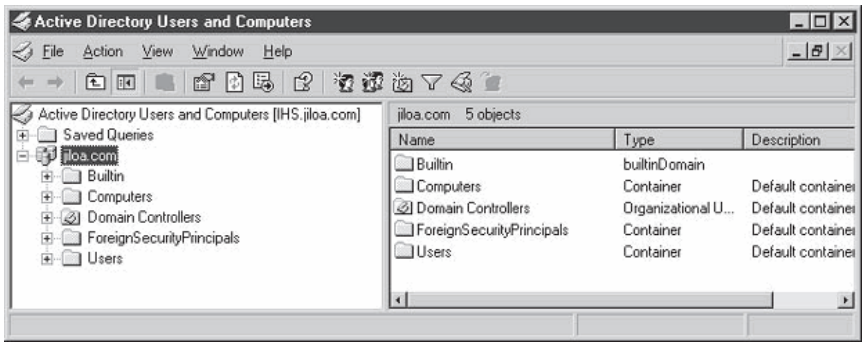


Рис. 21.1 - Утилита Active Directory Users and Computers (Пользователи и компьютеры Active Directory)

При создании учетных записей пользователей основной используемой утилитой администрирования является оснастка Active Directory Users and Computers (Пользователи и компьютеры Active Directory), предназначенная для

администрирования учетных записей в рамках домена Active Directory. Оснастка Active Directory Users and Computers (Пользователи и компьютеры Active Directory) (см. рис. 21.1) используется для управления пользователями, группами и другими элементами, такими как организационные единицы для доменов в лесу. По умолчанию оснастка запускается из меню Start/Programs/Administrative Tools (Пуск/Программы/Администрирование) на каждом контроллере домена. Эту оснастку также можно добавить в любую консоль MMC.

21.2 Групповая политика и безопасность

Групповые политики (Group Policies - GP) *- представляют собой основной метод обеспечения централизованного управления конфигурацией безопасности в Windows 2000 и Windows 2003.* Они могут применяться на уровне сайта, домена и OU, а также могут применяться к пользователям и компьютерам (Users and Computers) в Active Directory.

GP используются для выполнения следующих действий:
- *Блокировка рабочих столов пользователей.*
- *Применение параметров безопасности.*
- *Ограничение доступа к приложениям.*
- *Установка разрешения реестра и файловой системы.*
- *Настройка конфигурации беспроводной сети.*

Рекомендуется использовать утилиту Group Policies вместо Local System Policies, если это возможно.

21.2.1 Параметры конфигурации групповых политик

По умолчанию GP применяются в зависимости от расположения настраиваемого объекта. Пользовательские GP зависят от того, в каком сайте, домене и организационной единице находится объект «пользователь». То же самое относится и к компьютеру. GP применяются к компьютерам в зависимости от расположения объекта «компьютер» (сайт, домен и организационная единица, в которой находится компьютер). Это означает, что если GP применяется к объекту User (Пользователь), то используется конфигурация пользователя, а конфигурация компьютера групповой политики игнорируется. И наоборот, если GP применяется к объекту Computer (Компьютер), используется конфигурация компьютера, а конфигурация пользователя игнорируется.

Утилита Group Policies *разделена на две области:*
- User (Пользователь),
- Computer (Компьютер).

Область настройки пользователя User Configuration *содержит такие элементы, как параметры рабочего стола, параметры безопасности и сценарии входа и выхода их системы.* Эти элементы определены под деревом User Configuration и применяются при входе в систему или обновлении групповой политики.

Область настройки компьютера Computer Configuration *используется для настройки работающей системной среды (а не пользовательской оболочки), включая параметры служб, параметры безопасности и сценарии загрузки/отключения.* Эти элементы определены в дереве Computer Configuration и применяются при загрузке и обновлении Group Policy.

Соответственно областям групповая политика имеет два основных дерева данных конфигурации:

- Computer Configuration (Конфигурация компьютера),
- Users Configuration (Конфигурация пользователей).

Эти области отображаются в виде двух раздельных секций в окне Group Policy Object Editor (Редактор объекта групповой политики) (см. рис. 21.2).

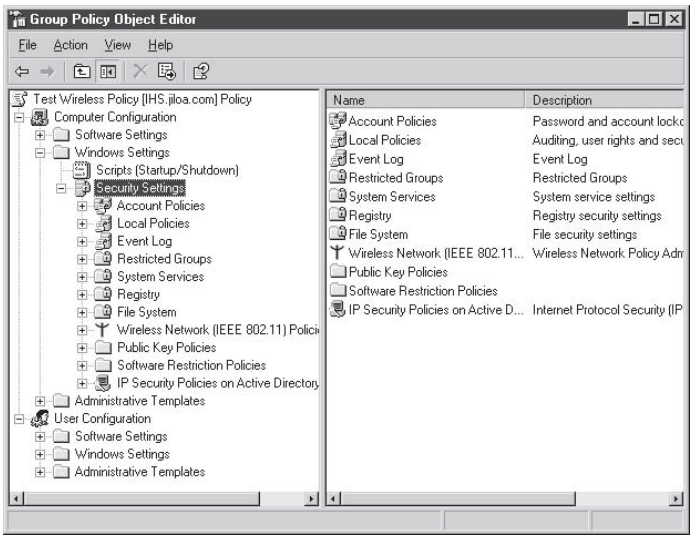


Рис. 21.2 - Редактор объекта групповой политики

Конфигурация компьютера содержит следующие настройки.

- Account Policies: Password Policy (Политики учетных записей: политика паролей). Позволяет настраивать историю, требования к возрасту, длине и сложности паролей.
- Account Policies: Account Lockout Policy (Политики учетных записей: политика блокировки учетных записей). Позволяет настраивать число попыток, длительность и сброс.
- Local Policies: Audit Policies (Локальные политики: политики аудита). Позволяет включать аудит в системах.

- Local Policies: User Rights Assignment (Локальные политики: присвоение прав пользователей). Позволяет присваивать пользовательские права пользователям и группам.
- Local Policies: Security Options (Локальные политики: параметры безопасности). Позволяет настраивать политики, связанные с безопасностью, включая подписи SMB, ограничения безопасности каналов, автоматический выход, уровень аутентификации LAN Manager, текстовое сообщение входа и примечание, а также множество других элементов (40 по умолчанию).
- Event Log: Settings for Event Logs (Журнал событий: параметры журналов событий). Позволяет настраивать объем журнала, ограничения доступа, параметры сохранения, а также необходимость отключения системы по заполнении журналов.
- Restricted Groups: Members of Restricted Group (Ограниченные группы: члены ограниченной группы). Предписывает членство в группе. Если пользователь или группа входят в список членов ограниченной группы, но не находятся в группе, происходит добавление в группу этого пользователя или группы. Если пользователь или группа является членом группы, но отсутствует в списке членов ограниченной группы, то этот пользователь или группа удаляется.
- Restricted Groups: Restricted Group Is Member Of (Ограниченные группы: ограниченная группа входит в…). Если ограниченная группа не входит в группу, которой она должна принадлежать, она добавляется в нее. В отличие от предписания членства в группе, описанного выше, если ограниченная группа принадлежит группе, которая здесь отсутствует, то эта ограниченная группа не удаляется.
- IP Security Policies (Политики безопасности IP). Позволяет настраивать списки и действия фильтров, правила политик, методы защиты и аутентификации, типы соединений и ключевые параметры и методы обмена.

Конфигурация пользователя содержит следующие настройки.

- Windows Settings: Internet Explorer Maintenance: Security (Настройки Windows: обслуживание Internet Explorer: безопасность). Позволяет настраивать особые зоны безопасности, оценку содержимого и параметры аутентификации.
- Windows Settings: Scripts (Настройки Windows: сценарии). Позволяет указывать сценарии входа и выхода из системы.
- Administrative Templates: Windows Components: Windows Explorer (Шаблоны администрирования: компоненты Windows: Проводник Windows). Позволяет настраивать пользовательские параметры для Проводника Windows. Среди этих параметров следует отметить:
  - удаление меню File (Файл),
  - опций Map Network Drive (Подключить сетевой диск),

  - Disconnect Network Drive (Отключить сетевой диск),
  - скрытие вкладки Hardware (Оборудование),
  - запрос аутентификационных данных для сетевых инсталляций,
  - и другое.
- Administrative Templates: Windows Components: Windows Installer (Шаблоны администрирования: компоненты Windows: программа установки Windows Installer). Позволяет запретить пользователям производить установку со съемных носителей, а также вносить другие изменения в конфигурацию.
- Administrative Templates: Start Menu and Taskbar (Шаблоны администрирования: меню Пуск и панель задач). Позволяет удалять папки пользователя из меню Start (Пуск), отключать и удалять ссылки на Windows Update, отключать опцию Log Off (Выход из системы) в меню Start (Пуск), отключать и удалять команду Shut Down (Завершение работы), удалять отдельные меню и др.
- Administrative Templates: Desktop (Шаблоны администрирования: Рабочий стол)**.** Используется для скрытия всех значков Рабочего стола, запрета на изменение пользователями пути к папке My Documents (Мои документы), необходимости сохранения параметров при выходе и др. Также позволяет настраивать элементы, связанные с Active Desktop, и взаимодействие пользователей с Active Directory.
- System: Group Policy (Система: групповая политика). Позволяет настраивать пользовательские параметры, такие как интервал обновления пользователей, выбор контроллера домена, автоматическое обновление файлов ADM и др.

Выше приведены наиболее важные компоненты оснастки Group Policies с указанием того, каким образом они связаны с безопасностью. Это лишь очень общее описание рассматриваемой области, а не полноценный обзор. Обязательно ознакомьтесь с более детальной информацией по данной теме перед тем, как вплотную заняться работой с оснасткой Group Policies.

21.2.2 Групповые политики по умолчанию

Имеются две групповые политики, установленные по умолчанию, создаваемые при создании домена:

1. Default Domain Policy (Политика домена по умолчанию) *применяется к контейнеру домена. Она может быть применена ко всем компьютерам в домене по умолчанию)*
2. Default Domain Controller Policy (Политика контроллера домена по умолчанию) *применяется к «специальному» контейнеру контроллера домена в домене и, кроме того, применима только к контроллерам домена).*

21.2.3 Дополнения групповой политики в Windows 2003

В Windows 2003 в групповую политику добавлены два отдельных элемента, связанных с безопасностью систем в AD. Этими элементами являются:

- Software Restriction Policies (Политики ограничения программного обеспечения)
- Wireless Network (IEEE 802.11) Policies (Политики беспроводных сетей [IEEE 802.11]).

Функции оснастки Group Policy такие же, как у оснастки Local Security Policy (Локальная политика безопасности), однако эту оснастку можно применить к домену или OU. Параметры, связанные с безопасностью, настраиваемые с помощью данной групповой политики, включают в себя следующие настройки:

- Тип беспроводной сети, к которой могут осуществлять доступ клиенты: Ad Hoc (Точка доступа), Infrastructure (Инфраструктура) или Any (Любая).
- Возможность запрета на использование беспроводными клиентами Windows локальных параметров Windows для настройки их параметров беспроводных сетевых соединений.
- Возможность разрешить пользователям подключаться только к предпочитаемым сетям.
- Возможность требовать аутентификацию 802.1X при каждом подключении к беспроводным сетям 802.11 (см. рис. 21.3).

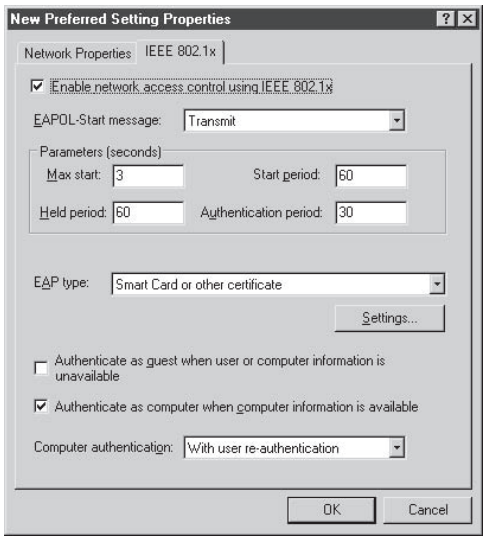


Рис. 21.3 - Свойства IEEE 802.1x

- Указание типа EAP: Smart Card or other certificate (Смарт-карта или другой сертификат) или Protected EAP (PEAP) (Защищенный EAP).

- Выбор метода аутентификации для использования в PEAP: Secured password (EAP-MSCHAP v2) (Защищенный пароль EAP-MSCHAP v2) или Smart Card or other certificate (Смарт-карта или другой сертификат).

21.2.4 Особенности применения настроек и политик безопасности

Ниже приведены шаги, автоматически выполняемые системой при оценке/применении Group Policy.

При загрузке системы:

1. Область Computer Configuration (Конфигурация компьютера) оснастки Local Security Policy (Локальная политика безопасности).
2. Область Computer Configuration (Конфигурация компьютера) оснастки Group Policies (Групповые политики), связанной с сайтом (в порядке предпочтения - от наименее до наиболее предпочтительного).
3. Область Computer Configuration (Конфигурация компьютера) оснастки Group Policies (Групповые политики), связанной с доменом.
4. Область Computer Configuration (Конфигурация компьютера) оснастки Group Policies (Групповые политики), связанной с OU, в порядке предпочтения - от самой внешней организационной единицы до самой внутренней, и внутри OU - с самого низкого уровня до самого высокого.

При входе пользователя:

1. Области User Configuration (Конфигурация пользователя) оснастки Local Security Policy (Локальная политика безопасности).
2. Области User Configuration (Конфигурация пользователя) оснастки Site Group Policies (Групповые политики сайта) в порядке предпочтения.
3. Области User Configuration (Конфигурация пользователя) оснастки Domain Group Policies (Групповые политики домена) в порядке предпочтения.
4. Области User Configuration (Конфигурация пользователя) оснастки OU Group Policies (Групповые политики организационного подразделения) в порядке предпочтения.

21.2.4.1 Замыкание на себя

По умолчанию GP применяются в зависимости от расположения настраиваемого объекта. Чтобы обойти эту возможность для пользователей, компания Microsoft реализовала замыкание на себя (loopback). Эта возможность используется для конфигурации пользователя групповых политик, а также конфигурации компьютера, в зависимости от расположения объекта «компьютер» (не пользователь) при входе пользователя в систему. Таким образом, каждый пользователь, осуществляющий вход в систему компьютера, получает конфигурацию пользователя (User Configuration) из

групповых политик этого компьютера. При включении опции можно также указать функцию Merge (Слияние) (объединение конфигурации из всех групповых политик) или Replace (Замещение) (только применение конфигураций пользователей в зависимости от расположения объекта «компьютер»).

21.2.4.2 Наследование

Во многом аналогично наследованию списков ACL, параметры GP передаются от самых дальних к самым ближним, причем ближние/низшие имеют большее старшинство. Порядок оценки таков:

1. Local Security Policy (Локальная политика безопасности),
2. Site Group Policies (Групповые политики сайта),
3. Domain Group Policies (Групповые политики домена),
4. OU Group Policies (Групповые политики организационного подразделения).

Существует возможность блокировки наследования политики, если не требуется наследовать параметры. Это позволит блокировать групповые политики, связанные с сайтами, доменами или организационными единицами высших уровней от применения их к текущему сайту, домену или организационному подразделению и к их дочерним объектам. Как администратору верхнего уровня вам может понадобиться включение принудительного использования некоторых политик верхнего уровня (например, минимальная длина пароля); для этого существует опция No Override (Игнорирование невозможно). Эту опцию можно включить для того, чтобы предотвратить обход (включая блокировку) политики любым дочерним объектом.

По большому счету, между сайтами и доменами в действительности нет никакого «наследования». Будет происходить оценка только тех групповых политик, связанных с конкретным сайтом или доменом, в котором находится пользователь или компьютер. Организационная единица является единственным контейнером, для которого действительно наблюдается наследование при проходе вниз по дереву элементов.

21.2.5 Средства управления групповой политикой

Следующие утилиты весьма полезны для управления групповыми политиками и просмотра результатов их работы.

Group Policy Management Console. Утилита Group Policy Management Console (Консоль управления групповой политикой) представляет собой оснастку MMC и набор сценариев, предоставляющих единый интерфейс управления групповой политикой на предприятии. Интерфейс показан на рисунке 21.4 с отображением части политики домена по умолчанию (Default Domain Policy) для домена *jiloa.com*.

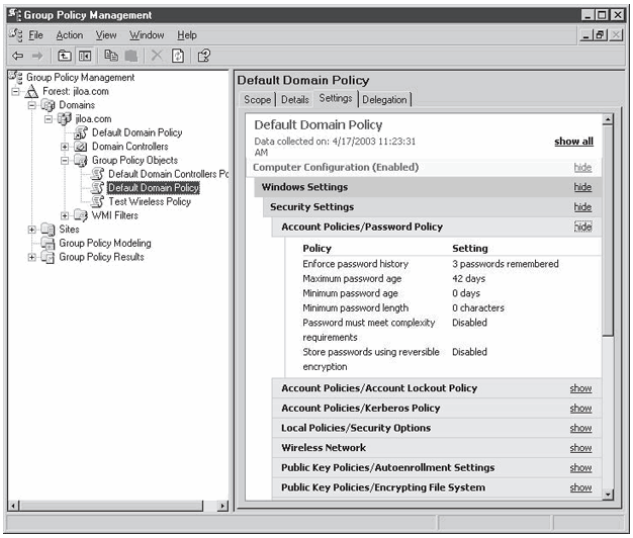


Рис. 21.4 - Консоль управления групповой политикой

Group Policy Results. Консоль управления групповой политикой предоставляет средство для определения результирующей политики для данного пользователя и/или системы. (Этот метод отличается от средства Resultant Set of Policy, обсуждаемого ниже) Чтобы сгенерировать запрос Group Policy Results (Результаты групповой политики) для пользователя/компьютера, нужно открыть лес, щелкнуть правой кнопкой мыши на пункте Group Policy Results (Результаты групповой политики**)** и затем выбрать Group Policy Results Wizard (Мастер результатов групповой политики). Выполните предписания мастера и введите соответствующую информацию в окнах ввода данных. На рисунке 21.5 показаны результаты запроса Group Policy Results для администратора в IHS в домене *jiloa.com*.

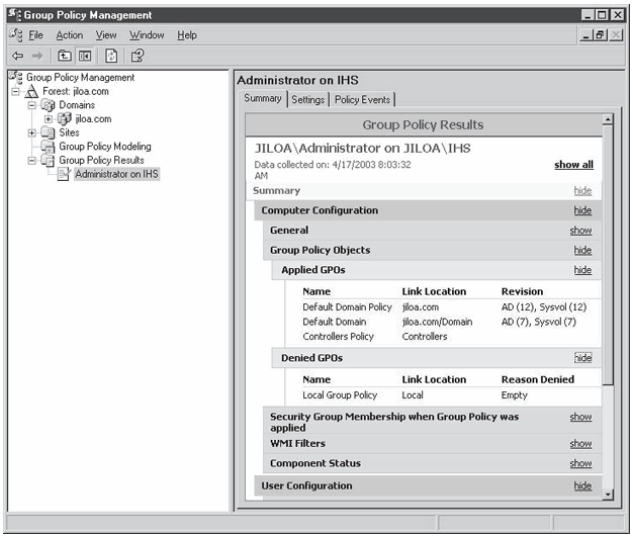


Рис. 21.5 - Результаты групповой политики для администратора в IHS

Resultant Set of Policy (RSoP). Утилита предназначена для облегчения процессов применения политик и устранения неполадок в них. Она предоставляет детальные сведения обо всех сконфигурированных параметрах политики и может помочь определить набор примененных

политик и порядок, в котором они применяются. Это очень полезно, когда несколько политик применяются на различных уровнях, таких как сайт, домен и организационное подразделение (единица).

Эта утилита используется для симуляции результатов применения параметров политики, которые вы собираетесь применить к компьютеру или пользователю, а также для определения параметров текущей политики для пользователя, находящегося в данный момент в системе компьютера. На рисунке 21.6 приведен пример RSoP для политики аудита системы IHS. RSoP находится в оснастке MMC и открывается в консоли управления Microsoft (MMC), оснастке Active Directory Users and Computers (Пользователи и компьютеры Active Directory) или оснастке Active Directory Sites and Services (Сайты и службы Active Directory).

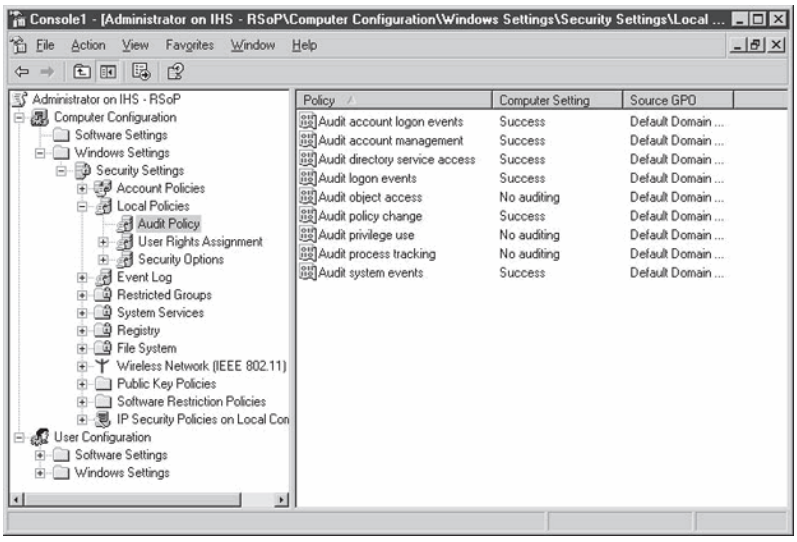


Рис. 21.6 - RSoP для политики аудита на IHS

22. ОСНОВЫ БЕЗОПАСНОСТИ ОПЕРАЦИОННЫХ СИСТЕМ СЕМЕЙСТВА UNIX

На протяжении большей части истории интернета системы Unix обеспечивали наивысший уровень функционирования служб в сети. В данной лекции приводятся некоторые базовые соображения безопасности, связанные с построением и защитой системы Unix. Ввиду большого числа доступных на рынке Unix-систем точные местоположения файлов и команды не являются абсолютно правильными для всех версий Unix.

22.1 Настройка системы

После построения системы Unix в ней, как правило, присутствует ряд уязвимостей. Большую их часть можно устранить посредством обновления системы или внесения изменений в конфигурационные файлы.

22.1.1 Файлы загрузки

Системы Unix настраиваются при загрузке с использованием соответствующих загрузочных файлов. В зависимости от версии Unix файлы загрузки могут располагаться в различных местах. В системе Solaris файлы загрузки находятся в каталоге /etc/rc2.d, в системе Linux - в каталоге /etc/rc.d/rc2.d. В различных версиях Unix файлы могут располагаться в различных местах, это расположение действительно для Red Hat.

В файлах загрузки запускается ряд служб. Некоторые из них (сеть, монтировка файловых систем и журнал запуска) необходимы для функционирования системы, и ничто не должно препятствовать их работе. Другие службы не являются столь критичными и запускаются в зависимости от того, каким образом используется система. Чтобы предотвратить запуск службы, просто измените имя файла. Если служба не понадобится в будущем, файл можно удалить.

Службы, обычно запускаемые при помощи файлов загрузки, включают в себя следующие сервисы:

- Inetd;
- NFS;
- NTP;
- Routed;
- RPC;
- Sendmail;
- Web servers.

22.1.2 Службы

Набор служб, выбранных для систем Unix, зависит от того, каким образом они будут использоваться. Некоторые из этих служб будут

запускаться с помощью файлов загрузки; ряд служб контролируется через сервис inetd и настраивается в файле /etc/inetd.conf.

Приведенный ниже текст представляет собой часть файла inetd.conf системы Solaris. Строки, начинающиеся с символа решетки <#> - являются комментариями.

```
#ident «@(#)inetd.conf 1.27 96/09/24 SMI»
/*SVr4.0 1.5 */
# Ftp and telnet are standard Internet services.
ftp stream tcp nowait root
/usr/sbin/in.ftpd in.ftpd
#telnet stream tcp nowait root /usr/sbin/in.telnetd
in.telnetd
#
# Shell, login, exec, comsat and talk are BSD protocols.
#shell stream tcp nowait root
/usr/sbin/in.rshd in.rshd
#login stream tcp nowait root /usr/sbin/in.rlogind
in.rlogind
#exec stream tcp nowait root
/usr/sbin/in.rexecd in.rexecd
#comsat dgram udp wait root
/usr/sbin/in.comsat in.comsat
#talk dgram udp wait root
/usr/sbin/in.talkd in.talkd
#
# Solstice system and network administration class agent server
#100232/10 tli rpc/udp wait root /usr/sbin/sadmind sadmind
```

Файл inetd.conf не только контролирует службы типа FTP и telnet, но и некоторые службы RPC. Файл inetd.conf необходимо очень внимательно проверять на предмет того, что в нем сконфигурированы только необходимые службы. После правильной настройки файла необходимо перезапустить службу inetd посредством следующей команды:

```
#kill -HUP <номер процесса inetd>
```

Команда -HUP вызывает повторное считывание службой inetd ее конфигурационного файла.

Многие службы, настраиваемые по умолчанию на системах Unix, необходимо отключить. Ниже приведен перечень этих служб:

Chargen	rexd	Systat
Discard	Routed	Tftp
Echo	Rquotad	Uucp
Finger	Rusersd	Walld
netstat	sprayd	

Кроме того, можно отключить службы Daytime, Time и SNMPD, если они не используются. Служба Time может использоваться некоторыми системами синхронизации, а служба SNMPD - для управления системой.

Как видно из приведенного выше фрагмента содержимого файла inetd.conf, службы telnet и FTP, как правило, настроены на рабочее состояние. Эти два протокола позволяют передавать идентификаторы пользователей и пароли через сеть в открытом виде. Возможно использование шифрующих версий этих протоколов для защиты паролей. При работе через telnet рекомендуется использовать Secure Shell (SSH). Некоторые версии SSH входят в программу Secure Copy (SCP) для передачи файлов.

22.1.3 Сетевая файловая система NFS

Внутри организации может потребоваться использование файловой системы Network File System (NFS). Если это не так, отключите NFS на любой системе, на которой не требуется ее использование. NFS предназначена для монтирования файловой системы с одной системы на другую. Если NFS настроена неправильно, то велика вероятность того, что кто-то получит доступ к секретным файлам. Чтобы правильно настроить NFS, следует соответствующим образом изменить файл /etc/dfs/dfstab.

Примечание. Неблагоразумно разрешать экспорт файловых систем во внешнюю среду из рассматриваемой организации.

22.1.4 Серверы и рабочие станции

В некоторых организациях операционная система Unix используется как на серверах, так и на рабочих станциях. При использовании на рабочей станции система обычно настраивается на функционирование системы X Window System. На системах Solaris в этом случае используется программа ToolTalk (RPC-программа, предназначенная для связи между приложениями).

Эти службы не нужны на серверах, а службы DNS и routed не требуются на рабочих станциях. Необходимо разработать руководство по настройке серверов и руководство для настройки рабочих станций, если система Unix используется описанным выше образом.

Примечание. Программа ToolTalk контролируется посредством inetd.conf на системах Solaris. Чтобы отключить эту программу, необходимо закомментировать следующую строку:

```
100083/1 tli rpc/tcp wait root
  /usr/dt/bin/rpc.ttdbserverd/usr/dt/bin/rpc.ttdbserverd.
```

22.1.5 Использование программ TCP Wrappers

Программы TCP Wrappers (доступны по адресу ftp://ftp.porcupine.org/pub/security) используются для обеспечения дополнительного уровня защиты в случае применения служб telnet или FTP. Как видно из названия, программы TCP Wrappers (wrap - оболочка) создают «оболочку» для служб telnet и FTP с целью обеспечения дополнительного контроля доступа и ведения журналов. Для использования программы TCP Wrappers необходимо настроить файл inetd.conf так, чтобы строки telnet и FTP выглядели следующим образом:

```
ftp stream tcp nowait root /usr/local/bin/tcpd /usr/sbin/in.ftpd
telnet stream tcp nowait root /usr/local/bin/tcpd /usr/sbin/in.telnetd
```

Эти строки вызывают запуск TCP Wrappers (tcpd) службой inetd, когда кто-либо пытается установить с системой сеанс связи через telnet или FTP.

Примечание. TCP Wrappers можно использовать и для других служб, таких как POP и IMAP. Нужно просто внести соответствующие изменения в строки конфигурации, представленные выше. TCP Wrappers можно настроить на блокировку или разрешение определенным узлам или сетям доступа к службам telnet и FTP. Файлы, используемые для этих действий по настройке, - это файлы /etc/hosts.allow и /etc/hosts.deny. Синтаксис для работы с этими файлами выглядит следующим образом:

```
<имя программы-оболочки>:  <ip-адрес>/<маска сети>
```

Следующие файлы представляют собой примеры файлов конфигурации TCP Wrapper.

```
hosts.allow:
#Allow telnets from my internal network (10.1.1.x)
in.telnet: 10.1.1.0/255.255.255.0
#Allow ftp from the world
in.ftpd: 0.0.0.0/0.0.0.0
hosts.deny:
#Deny telnets from anywhere else
in.telnetd: 0.0.0.0/0.0.0.0
```

Файл hosts.allow оценивается в первую очередь, после чего обрабатывается файл hosts.deny. Следовательно, можно сначала настроить все системы, которым разрешено работать с различными службами, после чего запретить все остальное в файле hosts.deny. Кроме того, следует внести изменение в настройку журнала, чтобы разрешить TCP Wrappers заносить данные в журнал системы.

22.1.6 Файлы конфигурации системы

Существует ряд изменений, которые можно внести в файлы конфигурации системы Unix, чтобы увеличить общий уровень безопасности системы. Это могут быть как предупреждающие сообщения, так и защита от переполнения буфера на некоторых системах. Любые изменения должны вноситься в конфигурацию в соответствии с политикой безопасности организации.

Внимание! Имейте в виду, что в различных версиях систем Unix файлы конфигурации располагаются в различных местах. Обратитесь к руководствам или инструкциям конкретной используемой версии Unix, чтобы удостовериться в корректности вносимых изменений в отношении рассматриваемой версии системы.

Приветственные сообщения могут использоваться для заявления о правах собственности перед входом пользователя в систему. Сообщение должно быть написано на языке, разрешенном для использования юридическим отделом организации.

Приветственное сообщение хранится в /etc/motd (сокр. от «message of the day» - сообщение дня). Однако это сообщение отображается не перед входом пользователя в систему, а после него. Большинство уведомлений, связанных с юридическими вопросами, необходимо отображать перед входом пользователя в систему.

Чтобы сообщение отображалось перед входом пользователя в систему, используйте следующий способ. В ОС Solaris предварительное уведомление хранится в каталоге /etc/default/telnetd. Можно создать сообщения входа для FTP посредством редактирования файла /etc/default/ftpd. Для создания сообщения добавьте в файл строку, аналогичную следующей:

```
BANNER=«\n\n<Enter Your Legal Message Here\n\n»
```

Параметр \n означает новую строку. Поэкспериментируйте с символами новой строки, чтобы сообщение приняло нужный вам вид.

В системах Linux для сообщений telnet используются два файла: /etc/issue и /etc/issue.net. Файл issue применяется для терминалов, подключенных напрямую, а issue.net используется в том случае, когда кто-либо устанавливает по сети соединение через telnet с рассматриваемой системой. К сожалению, только на изменении этих файлов создание сообщения не закончится, так как они создаются заново при каждой загрузке системы. Однако можно изменить сценарий загрузки, создающий эти файлы.

Файлы создаются в сценарии загрузки /etc/rc.d/rc.local. Чтобы предотвратить автоматическое создание /etc/issue и /etc/issue.net, закомментируйте следующие строки /etc/rc.d/rc.local:

```
# This will overwrite /etc/issue at every boot. So, make any changes you
# want to make to /etc/issue here or you will lose them when you reboot.
echo «« > /etc/issue
echo «$R» > /etc/issue
echo «Kernel $(uname -r) on $a $SMP$(uname -m)» >> /etc/issue
```

После этого можно изменить /etc/issue и /etc/issue.net, введя в них соответствующий текст с заявлением о правах.

22.2 Настройки паролей

Существует три этапа процедуры управления паролями в системе Unix.

- Настройка требований к паролям.
- Запрет на вход без пароля.
- Указание требований к содержимому паролей.

22.2.1 Настройка требований к паролю

В системах Unix требования к возрасту паролей и их длине устанавливаются посредством изменения файла конфигурации. В системе Solaris этим файлом является /etc/default/passwd. Файл содержит приведенные ниже строки, которые следует редактировать для соответствия политике безопасности организации.

```
#ident    «@(#)passwd.dfl    1.3    92/07/14 SMI»
MAXWEEKS=7
MINWEEKS=1
PASSLENGTH=8
```

Внимание! Будьте внимательны при указании значений максимального и минимального срока действия паролей, так как система воспринимает вводимые значения как количество недель, а не дней.

В каждой организации должны быть разработаны процедуры конфигурации, специфичные для конкретной используемой системы; при этом необходимо руководствоваться политикой безопасности. Эти процедуры должны определять, каким образом следует настраивать систему с использованием конкретной операционной системы, чтобы обеспечить соответствие ОС требованиям политики безопасности.

В системах Linux требования к паролям находятся в файле /etc/login.defs. Следующие строки файла /etc/login.defs представляют собой настраиваемые параметры:

```
# Password aging controls:
#
#   PASS_MAX_DAYS Maximum number of days a password may be used.
#   PASS_MIN_DAYS Minimum number of days allowed between password changes.
#   PASS_MIN_LEN Minimum acceptable password length.
#   PASS_WARN_AGE Number of days warning given before a password expires.
#
PASS_MAX_DAYS 45
PASS_MIN_DAYS 1
PASS_MIN_LEN 8
PASS_WARN_AGE 7
```

Внимание! Имейте в виду, что в системах Linux минимальные и максимальные значения возраста паролей указываются в днях. Linux также позволяет предупреждать пользователей о том, что до окончания срока действия пароля осталось несколько дней.

22.2.2 Запрет на вход без пароля

Программы rlogin, rsh и rexec позволяют пользователям осуществлять вход в систему с определенных систем без указания пароля вручную. Этого делать не рекомендуется, так как злоумышленник, проникший в одну из систем, может таким образом получить доступ к остальным компьютерам. Помимо удаления служб rlogin, rsh и rexec из /etc/inetd.conf следует удостовериться в том, что файл /etc/host.equiv и любые файлы .rhost, имеющиеся в системе, найдены и удалены. Не забудьте также проверить домашние каталоги всех пользователей.

22.2.3 Указание требований к содержимому паролей

Запрет пользователям на выбор ненадежных паролей является одним из наилучших способов повышения уровня безопасности системы. К сожалению, до недавнего времени в системах Unix существовало несколько простых способов это сделать. Программы типа passwd+ и npasswd имеются для Linux, но не для Solaris. Обе эти программы позволяют указывать требования к надежности паролей и вынуждают пользователей выбирать пароли, соответствующие установленным правилам.

С выходом Solaris 2.6 и более поздних реализаций Linux появилось более совершенное средство отслеживания надежности паролей пользователей - это Pluggable Authentication Modules (PAM). Более подробная информация о PAM и о том, как создать фильтры паролей, находится по адресу http://www.sun.com/solaris/pam/; для системы Linux - по адресу ftp://ftp.kernel.org/pub/linux/libs/pam/index.html.

Примечание. Некоторые версии Unix, в особенности HPUX, поставляются с настройками по умолчанию надежности паролей для

обеспечения безопасности. В них указывается набор блокировок для учетных записей на случай слишком большого числа неудачных попыток входа в систему.

22.3 Контроль доступа к файлам

В системе Unix доступ к файлам контролируется посредством набора разрешений. Для владельца файла, группы, которой принадлежит файл, и для всех остальных лиц можно присваивать привилегии чтения, записи и выполнения. Файловые разрешения изменяются посредством команды chmod. Как правило, не рекомендуется разрешать пользователям создавать файлы, доступные для чтения или записи для любых лиц. Такие файлы могут считываться или записываться любым пользователем системы. Если злоумышленник получит доступ к идентификатору пользователя, он сможет считать или изменить любые из таких файлов.

Так как достаточно трудно убедить всех пользователей в необходимости изменять разрешения доступа к файлу при его создании, разумно создать механизм, используемый по умолчанию, предназначенный для настройки соответствующих разрешений при автоматическом создании файла. Это можно осуществить с помощью параметра unmask. В системах Solaris этот параметр располагается в файле /etc/default/login, в системах Linux - в /etc/profile. Команда выполняется следующим образом:

```
unmask 077
```

Цифры, указываемые после команды, определяют разрешения, которые не будут присвоены по умолчанию вновь создаваемому файлу. Первая цифра определяет разрешения относительно владельца файла, вторая цифра указывает разрешения для группы, а третья - для всех остальных пользователей. В случае, рассмотренном выше, все новые файлы присваивают разрешения чтения, записи и выполнения владельцу того или иного файла, а группе и всем остальным пользователям не предоставляется никаких разрешений.

Разрешения определяются числами следующим образом:
4 - Разрешение на чтение,
2 - Разрешение на запись,
1 - Разрешение на выполнение.

Следовательно, если требуется разрешить группе иметь по умолчанию разрешение на чтение, но запретить запись и выполнение, нужно указать команду unmask 037. Если требуется запретить группе запись, следует указать команду unmask 027.

22.4 Доступ через корневую учетную запись

Как правило, рекомендуется ограничивать прямой доступ с использованием корневой учетной записи. При таком подходе даже администраторам необходимо сначала выполнить вход систему с использованием их аутентификационных данных, и только после этого с помощью команды su получить доступ к корневой учетной записи. Это также обеспечивает создание записей в журнале, отображающих, какие идентификаторы пользователей использовались для получения доступа к корневой учетной записи. В качестве альтернативы вместо команды su в некоторых версиях Linux (например Ubuntu Linux) можно использовать команду sudo. Команда sudo обеспечивает дополнительные возможности по ведению журналов, заключающиеся в фиксировании команд, выполняемых пользователями, работающими в корневой учетной записи.

Существует возможность ограничить вход под корневой учетной записью таким образом, чтобы его можно было осуществлять только из консоли Solaris или Linux. В системе Solaris следует изменить файл /etc/default/login и убедиться в том, что следующая строка не закомментирована:

```
# If CONSOLE is set, root can only login on that device.
# Comment this line out to allow remote login by root.
#
CONSOLE=/dev/console
```

Посредством этого система разрешит прямой вход в корневую учетную запись только через консоль. В системе Linux можно реализовать аналогичную конфигурацию, редактируя файл /etc/securetty. Этот файл представляет собой список **TTY**, которые используются для входа в корневую учетную запись. Содержимым этого файла должно быть /dev/tty1. Если для управления системой используется последовательный канал связи, файл должен содержать /dev/ttyS0. Сетевые TTY - это, как правило, /dev/ttyp1 и выше.

Если требуется контролировать корневой доступ к системе, рекомендуется осуществлять контроль корневого доступа к FTP. Файл /etc/ftpusers и в системах Solaris, и в системах Linux представляет перечень учетных записей, которым не разрешено осуществлять доступ к системе через FTP. Убедитесь, что в данном списке присутствует корневая учетная запись.

22.5 Защита от переполнения буфера

Переполнение буфера - одна из наиболее серьезных опасностей, угрожающих системе. Solaris предоставляет способ предотвращения

выполнения команд вне стека при проявлении атак на переполнение буфера. Для этого необходимо добавить следующие строки в файл /etc/system:

```
set noexec_user_stack=1
set noexec_user_stack_log=1
```

Первая строка предотвращает выполнение команд вне стека, а вторая - заносит в журнал данные о произведенных попытках.

Внимание! Существует ряд программ, которым требуется выполнять команды вне стека. Если внести описанное изменение, то при работе этих программ возникнут сбои. Убедитесь, что данная команда протестирована, прежде чем применять ее на системах.

Существует несколько других проектов, предназначенных для повышения уровня защиты стека Linux. Один из них расположен по адресу http://www.openwall.com/linux/.

22.6 Управление пользователями

Как в случае с любой операционной системой, управление сообществом пользователей является очень важным процессом для поддержки общей безопасности системы. В организации должна присутствовать специальная процедура управления пользователями, предусматривающая в деталях все действия, которые необходимо выполнить, чтобы предоставить сотруднику доступ к системе. В процедуре должны быть определены шаги, которые следует предпринимать, когда сотрудник увольняется из компании.

Следующие разделы данной лекции содержат некоторые подробные рекомендации по управлению пользователями в системах Unix. Имейте в виду, что существует множество вариаций систем Unix. Средства, используемые для управления пользователями, различны для каждого поставщика и версии операционной системы.

22.6.1 Добавление пользователей в систему

В большей части версий Unix имеются утилиты для добавления пользователей в систему. Здесь ключевыми задачами являются следующие:

- Добавление имени пользователя в файл паролей.
- Присвоение соответствующего идентификатора пользователя.
- Присвоение соответствующего группового идентификатора.

Примечание. Большая часть систем содержит утилиты по добавлению пользователей для обеспечения автоматического выполнения этой задачи. В Linux для этого предназначена программа adduser. В системе Solaris эта утилита называется useradd.

22.6.1.1 Добавление имени пользователя в файл паролей

Файл /etc/passwd содержит перечень всех имен пользователей, принадлежащих пользователям системы. Каждый пользователь должен иметь уникальное имя, состоящее из восьми или менее символов. Для каждой записи в файле паролей должно быть определено реальное лицо, ответственное за учетную запись. Данную информацию можно добавить в поле GECOS (пятое поле в каждой строке).

22.6.1.2 Присвоение идентификационного номера пользователя

Каждому имени пользователя необходимо присвоить соответствующий идентификатор пользователя (UID). UID должен быть уникальным в рамках всей системы. Как правило, идентификатор пользователя должны быть больше 100. Он ни в коем случае не должен быть равен 0, так как это идентификатор корневой учетной записи.

Внимание! Система использует UID для идентификации владельцев файлов в системе и, таким образом, не рекомендуется даже повторное использование UID.

22.6.1.3 Присвоение группового идентификатора

Каждый пользователь должен иметь главную группу. Присвойте этот номер имени пользователя в файле /etc/passwd. Обычные пользователи не должны быть членами группы «wheel», так как она используется в административных целях.

22.6.2 Определение оболочки для входа в систему

В задачу определение соответствующей оболочки пользователя для входа в систему и работы (некоторые пользователи могут вовсе не иметь какой-либо оболочки) входит:

- Добавление имени пользователя в теневой файл shadow.
- Указание соответствующего начального пароля.
- Определение соответствующего псевдонима электронной почты.
- Создание домашнего каталога пользователя.

Интерактивным пользователям необходимо предоставить оболочку для входа в систему. Как правило, это оболочки ksh, csh или bash. Пользователям, которые не будут осуществлять вход в систему, нужно предоставить программу, не являющуюся оболочкой. Например, если имеются пользователи, которые только проверяют электронную почту через POP или IMAP, им можно разрешить изменять свои пароли в интерактивном режиме. В данном случае существует возможность определить оболочку, указав в

качестве нее /bin/passwd. При каждом подключении пользователей к системе через telnet им будет предоставляться возможность изменить пароль. По завершении этой операции пользователь будет выходить из системы.

22.6.2.1 Добавление имени пользователя в теневой файл

Пароли не должны храниться в файле /etc/passwd, так как этот файл доступен для чтения всем пользователям, и с его помощью злоумышленник может осуществить взлом пароля. Пароли должны храниться в файле /etc/shadow. Следовательно, имя пользователя должно быть добавлено и в файл /etc/shadow.

22.6.2.2 Присвоение начального пароля

После создания учетной записи следует установить начальный пароль. Большая часть утилит, используемая для добавления пользователей в системы, предлагает сделать это автоматически. В противном случае нужно войти в систему как пользователь и выполнить команду passwd. После этого появится предложение указать пароль для учетной записи. Начальные пароли должны быть сложными для угадывания, и рекомендуется не использовать один и тот же начальный пароль для всех учетных записей. Если используется один и тот же начальный пароль, атакующий может использовать новые учетные записи, прежде чем у легального пользователя появится возможность войти в систему и изменить пароль.

22.6.2.3 Определение электронной почты

При создании пользователя он автоматически получает адрес электронной почты <имя_пользователя>@host. Если пользователь хочет иметь другой адрес электронной почты, такой как имя.фамилия@host, то этот адрес можно присвоить посредством псевдонима электронной почты. Чтобы добавить псевдоним, измените файл /etc/aliases. Формат этого файла таков:

Alias: username

После создания псевдонима необходимо запустить программу newaliases, чтобы создать файл alias.db.

22.6.2.4 Создание домашнего каталога для пользователя

Каждый пользователь должен иметь свой собственный домашний каталог. Этот каталог определяется в файле /etc/passwd. После создания каталога в соответствующем месте в системе (как правило, это каталог /home или /export), владельцем каталога назначается пользователь командой chown следующим образом:

chown <username> <directory name>

22.6.2 Удаление пользователей из системы

Когда сотрудник увольняется из компании или переводится на другую работу, так что его учетная запись становится ненужной, необходимо выполнить соответствующую процедуру по управлению пользователями. В системе Unix все файлы пользователей принадлежат UID пользователя. Следовательно, если пользовательский UID повторно используется для новой учетной записи, эта новая учетная запись будет предусматривать владение всеми файлами старого пользователя.

Изначально, если пользователю больше не требуется учетная запись, ее следует заблокировать. Это можно сделать посредством замены пароля пользователя в файле /etc/shadow символами <*LK*>. По прошествии определенного числа дней (как правило, 30 дней), файлы пользователя могут быть удалены. Время, отведенное менеджеру пользователя на копирование или удаление файлов пользователя, требуемых организации, равно 30 дням.

22.6.3 Отключение неиспользуемых учетных записей

В Unix создается набор учетных записей, необходимых для различных целей (например, владение некоторыми определенными файлами), которые никогда не используются для входа в систему. Такими учетными записями являются sys, uucp, nuucp и listen. Для каждой учетной записи следует изменить их записи в файле /etc/shadow, чтобы предотвратить успешный вход в систему с их помощью.

```
root:XDbBEEYtgskmk:10960:0:99999:7:::
bin:*LK*:10960:0:99999:7:::
daemon:*LK*:10960:0:99999:7:::
adm:*LK*:10960:0:99999:7:::
lp:*LK*:10960:0:99999:7:::
sync:*LK*:10960:0:99999:7:::
shutdown:*LK*:10960:0:99999:7:::
halt:*LK*:10960:0:99999:7:::
mail:*LK*:10960:0:99999:7:::
news:*LK*:10960:0:99999:7:::
uucp:*LK*:10960:0:99999:7:::
operator:*LK*:10960:0:99999:7:::
games:*LK*:10960:0:99999:7:::
gopher:*LK*:10960:0:99999:7:::
ftp:*LK*:10960:0:99999:7:::
nobody:*LK*:10960:0:99999:7:::
```

Второе поле в каждой строке представляет собой поле пароля. В случае с обычными пользовательскими учетными записями здесь располагается зашифрованный пароль. Для учетных записей, вход посредством которых запрещен, второе поле должно содержать какие-либо данные с символом «*». Символ «*» не соответствует ни одному реальному паролю и, таким образом,

не может быть угадан или взломан. Посредством размещения в поле пароля соответствующих символов, таких как «LK», можно явным образом сообщать о том, что данная учетная запись заблокирована.

22.7 Управление системой

Управление системой Unix (относительно вопросов безопасности) заключается в ведении журнала и отслеживании системы на наличие признаков подозрительной активности. Системы Unix предоставляют достаточное количество информации о том, что происходит в системе, а также набор средств, которые могут использоваться для выявления подозрительной активности.

22.7.1 Аудит системы

В большинстве случаев ведение системных журналов является стандартной процедурой, выполняемой в большинстве версий Unix, и в них заносится достаточный объем данных, связанных с безопасностью системы. В некоторых ситуациях требуется проведение дополнительного аудита. В Solaris для этого предусмотрен модуль Basic Security Module (BSM). BSM не включен в Solaris по умолчанию. Необходимость в дополнительных возможностях здесь определяется пользователем.

Чтобы включить BSM, выполните сценарий /etc/security/bsmconv. При этом запустится фоновая программа аудита, но перезагрузка системы не потребуется. Файл /etc/security/audit_control используется для определения конфигурации аудита. Полная информация по этому файлу находится в инструкции к ОС (man audit_control), однако для начала рекомендуется использовать следующую конфигурацию:

```
#identify the location of the audit file directory
dir: <directory>
#identify the file system free space percentage when a warning should occur
minfree: 20
#flags for what to audit. This example audits login, administrative
#functions and failed file reads, writes, and attribute changes
flags: lo,ad,-fm
#This set of flags tells the system to also audit login and administrative
#events that cannot be attributed to a user
naflags: lo,ad
```

Как только файл будет настроен, начнут создаваться записи аудита. Для закрытия текущего файла записи аудита и открытия нового файла используется команда audit -n. Команда praudit <имя файла аудита> предназначена для просмотра содержимого файла аудита.

Внимание! BSM увеличивает общую нагрузку на систему и используется, только если уровень защиты системы того требует.

22.7.1.1 Файлы журналов

Большая часть систем Unix обеспечивает довольно широкие возможности по ведению журналов в программе syslog. Syslog - это фоновая программа, выполняющаяся и фиксирующая данные журнала согласно настройке. Syslog настраивается через файл /etc/syslog.conf. Следует заметить, файлы журналов должны просматриваться только корневым пользователем, и никто не должен иметь возможности их изменять.

Большая часть файлов syslog.conf направляет сообщения журналов в /var/log/messages или /var/adm/log/messages. Правильно написанный syslog.conf должен содержать следующую команду конфигурации:

```
auth.info /var/log/auth.log
```

С помощью этой команды Unix собирает информацию о попытках входа, попытках выполнения команды su, перезагрузке системы и других событиях, так или иначе связанных с безопасностью системы. Данная команда также позволяет программам TCP Wrappers заносить информацию в файл auth.log. Обязательно создайте файл /var/log/auth.log для фиксирования этой информации:

```
#touch     /var/log/auth.log
#chown     root /var/log/auth.log
#chmod     600 /var/log/auth.log
```

В Solaris при создании файла /var/adm/loginlog можно фиксировать неудачные попытки входа в систему. Создайте файл следующим образом:

```
#touch      /var/adm/loginlog
#chmod     600 /var/adm/loginlog
#chown     root /var/adm/loginlog
#chgrp     sys /var/adm/loginlog
```

Убедитесь, что /var предоставлено достаточное количество свободного пространства для ведения файлов журнала. Если /var расположен в том же разделе, что и /, корневая файловая система переполнится при сильном увеличении файлов журнала. Рекомендуется размещать каталог /var в другой файловой системе.

22.7.1.2 Скрытые файлы

Скрытые файлы представляют собой потенциальную проблему для систем Unix. Любой файл, начинающийся с точки (<.>), не отображается при выполнении стандартной команды ls. Однако при использовании команды

ls -a отобразятся все скрытые файлы. Хакеры научились использовать скрытые файлы для маскировки своих действий. Злоумышленник может просто скрыть свои файлы в скрытом каталоге. В других ситуациях хакеры могут скрывать файлы в каталогах, которые трудно обнаружить администратору. Например, если назвать каталог <...>, то он может остаться незамеченным. Добавление пробела после третьей точки (<...>) делает каталог труднодоступным, если не знать о наличии пробела. Чтобы отобразить все скрытые файлы и каталоги, имеющиеся в системе, выполните следующую команду:

```
#find / -name '.*' -ls
```

Использование -ls вместо -print позволяет вывести более подробный список расположения файла. Следует периодически выполнять эту команду и проверять любые новые скрытые файлы.

22.7.1.3 Файлы, которые могут изменять активного пользователя в процессе выполнения

Файлы, для которых разрешены полномочия Set UID (SUID) или Set Group ID (SGID), могут изменять идентификатор своего активного пользователя или группы в процессе выполнения. Некоторым файлам требуется такая возможность для выполнения своей работы, однако это должен быть ограниченный набор файлов, и ни один из них не должен находиться в домашних каталогах пользователей. Чтобы найти все файлы SUID и SGID, выполните следующие команды:

```
#find    /    -type f -perm -04000 -ls
#find    /    -type f -perm -02000 -ls
```

При построении системы необходимо выполнить данные команды и сохранить результаты их выполнения. Периодически следует выполнять эти команды и сопоставлять результаты с исходным списком. Любые обнаруженные изменения необходимо исследовать

22.7.1.4 Файлы, доступные для записи всем пользователям

Файлы, общедоступные для записи, являются еще одной потенциальной ошибкой в конфигурации системы Unix. Такие файлы позволяют злоумышленнику создать сценарий, который при выполнении будет использовать уязвимость. Если файлы SUID и SGID доступны для записи всем пользователям, у атакующего появляется возможность создать для самого себя самые обширные привилегии. Чтобы выявить все файлы, общедоступные для записи, выполните следующую команду:

```
#find   /   -perm -2 -type f -ls
```

Следует периодически выполнять эту команду, чтобы находить все общедоступные для записи файлы, имеющиеся в системе.

22.7.2 Мониторинг признаков подозрительной активности

Мы уже описали некоторые признаки, которые необходимо отслеживать в системе и которые могут означать проявление угрозы или проникновение в систему (скрытые файлы, файлы SUID и SGID и общедоступные для записи файлы). Существует несколько других способов проверки системы Unix на наличие подозрительной активности.

22.7.2.1 Смешанный режим

Интерфейс находится в смешанном режиме, когда в системе работает сниффер (сетевой анализатор пакетов). Сниффер переводит интерфейс в смешанный режим; при этом происходит фиксирование всей информации, проходящей через канал связи. Если при работе интерфейса в данном режиме выполнить команда ifconfig -a, то появится сообщение о том, что интерфейс находится в состоянии PROMISC (признак того, что работает анализатор пакетов). Если сниффер запущен не администратором системы, необходимо провести исследование причин этих обстоятельств.

Примечание. Solaris не выдает соответствующего отчета о том, что интерфейс находится в смешанном режиме. Причиной этому является ошибка в программном обеспечении ядра. Чтобы корректным образом проверить, находится ли интерфейс Solaris в смешанном режиме, необходимо использовать команду ifstatus, доступную по адресу ftp://ftp.cerias.purdue.edu/pub/tools/unix/sysutils/iftatus/.

22.7.2.2 Мониторинг активных сетевых соединений

Программа netstat

Программа netstat используется для выяснения того, какие сетевые соединения находятся в активном состоянии в системе Unix. Команду следует использовать следующим образом: netstat -an. Аргумент «n» сообщает netstat о том, что обработка IP-адресов не требуется.

```
#netstat -an
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address State
tcp        0      0 0.0.0.0:10000           0.0.0.0:*       LISTEN
tcp        0      0 0.0.0.0:25              0.0.0.0:*       LISTEN
tcp        0      0 0.0.0.0:515             0.0.0.0:*       LISTEN
tcp        0      0 0.0.0.0:98              0.0.0.0:*       LISTEN
tcp        0      0 0.0.0.0:113             0.0.0.0:*       LISTEN
tcp        0      0 0.0.0.0:79              0.0.0.0:*       LISTEN
tcp        0      0 0.0.0.0:513             0.0.0.0:*       LISTEN
tcp        0      0 0.0.0.0:514             0.0.0.0:*       LISTEN
tcp        0      0 0.0.0.0:23              0.0.0.0:*       LISTEN
tcp        0      0 0.0.0.0:21              0.0.0.0:*       LISTEN
tcp        0      0 0.0.0.0:111             0.0.0.0:*       LISTEN
udp        0      0 0.0.0.0:10000           0.0.0.0:*
udp        0      0 0.0.0.0:518             0.0.0.0:*
udp        0      0 0.0.0.0:517             0.0.0.0:*
udp        0      0 0.0.0.0:111             0.0.0.0:*
raw        0      0 0.0.0.0:1               0.0.0.0:*
raw        0      0 0.0.0.0:6               0.0.0.0:*
```

Как видно из результирующих данных, любая строка, содержащая слово «LISTEN», означает, что имеется программа, прослушивающая этот порт. Прослушиваться должны только сконфигурированные администратором порты. Если в системе присутствует прослушиваемый порт, который не конфигурировался администратором, систему необходимо проверить и выяснить, почему порт открыт.

Адреса, отображаемые в столбце локальных адресов, заканчиваются номером локального порта (число после столбца). Этот номер порта используется для определения того, является ли соединение входящим или исходящим. Например, если номер локального порта 23, то это входящее подключение к демону telnet. Если номер локального порта равен 1035, а номер внешнего порта - 23, то это исходящее соединение telnet.

Программа lsof

Одна из проблем, связанных с программой netstat, заключается в том, что данная команда не сообщает, какой процесс поддерживает открытое состояние порта. Поиск процесса, связанного с определенным портом, может стать очень трудной задачей. Однако существует программа под названием lsof (http://ftp.cerias.purdue.edu/pub/tools/unix/sysutil/Isof/), которая предоставляет такую информацию. Сразу после установки программы выполните команду lsof -i, как показано ниже:

```
#lsof -i
COMMAND  PID    USER   FD   TYPE   DEVICE  NODE NAME
portmap  311    root   4u   IPv4   301     UDP  *:sunrpc
portmap  311    root   5u   IPv4   302     TCP  *:sunrpc (LISTEN)
inetd    439    root   5u   IPv4   427     TCP  *:ftp (LISTEN)
```

```
inetd      439    root    6u    IPv4    428      TCP   *:telnet (LISTEN)
inetd      439    root    7u    IPv4    429      TCP   *:shell (LISTEN)
inetd      439    root    9u    IPv4    430      TCP   *:login (LISTEN)
inetd      439    root    10u   IPv4    431      UDP   *:talk
inetd      439    root    11u   IPv4    432      UDP   *:ntalk
inetd      439    root    12u   IPv4    433      TCP   *:finger (LISTEN)
inetd      439    root    13u   IPv4    434      TCP   *:auth (LISTEN)
inetd      439    root    14u   IPv4    435      TCP   *:linuxconf (LISTEN)
lpd        455    root    6u    IPv4    457      TCP   *:printer (LISTEN)
sendmail   494    root    4u    IPv4    495      TCP   *:smtp (LISTEN)
miniserv.  578    root    4u    IPv4    567      TCP   *:10000 (LISTEN)
miniserv.  578    root    5u    IPv4    568      UDP   *:10000
```

Как видно из результатов выполнения программы, lsof выводит перечень всех открытых портов с указанием того, какие процессы поддерживают открытое состояние портов. Убедитесь, что вам известно, какие функции выполняет каждый процесс, и почему открыт соответствующий ему порт.

Примечание. lsof заменяет номер порта в столбце справа именем порта, если оно присутствует в файле /etc/services.

22.7.2.3 Мониторинг активных процессов

Администратор также должен изучать результаты выполнения команды ps. Эта программа выводит все активные процессы, имеющиеся в системе, что необходимо при поиске снифферов, так как сниффер может не отображаться в lsof или в netstat. В большинстве систем выполнение команды ps -ef выводит перечень процессов в системе. В тех версиях Unix, где эта команда не работает, следует выполнить команду ps -aux. Результаты выполнения данной команды показаны ниже:

```
#ps -ef
UID        PID    PPID  C    STIME  TTY      TIME      CMD
root       1      0     0    13:09  ?        00:00:04  init
root       2      1     0    13:09  ?        00:00:00  [kflushd]
root       3      1     0    13:09  ?        00:00:00  [kupdate]
root       4      1     0    13:09  ?        00:00:00  [kpiod]
root       5      1     0    13:09  ?        00:00:00  [kswapd]
root       6      1     0    13:09  ?        00:00:00  [mdrecoveryd]
bin        3 11   1     0    13:09  ?        00:00:00  portmap
root       327    1     0    13:10  ?        00:00:00  /usr/sbin/apmd -p 10 -w 5 -W
root       380    1     0    13:10  ?        00:00:00  syslogd -m 0
root       391    1     0    13:10  ?        00:00:00  klogd
daemon     407    1     0    13:10  ?        00:00:00  /usr/sbin/atd
root       423    1     0    13:10  ?        00:00:00  crond
root       439    1     0    13:10  ?        00:00:00  inetd
root       455    1     0    13:10  ?        00:00:00  lpd
root       494    1     0    13:10  ?        00:00:00  sendmail: accepting connections
root       511    1     0    13:10  ?        00:00:00  gpm -t ps/2
xfs        528    1     0    13:10  ?        00:00:00  xfs -droppriv -daemon -port -1
root       570    1     0    13:10  tty1     00:00:00  login - root
root       571    1     0    13:10  tty2     00:00:00  /sbin/mingetty tty2
root       572    1     0    13:10  tty3     00:00:00  /sbin/mingetty tty3
root       573    1     0    13:10  tty4     00:00:00  /sbin/mingetty tty4
root       574    1     0    13:10  tty5     00:00:00  /sbin/mingetty tty5
root       575    1     0    13:10  tty6     00:00:00  /sbin/mingetty tty6
root       578    1     0    13:10  ?        00:00:00  perl /usr/libexec/webmin/miniser
root       579    570   0    13:10  tty1     00:00:00  -bash
root       621    579   0    13:17  tty1     00:00:00  ps -ef
```

Следует периодически проверять список процессов, работающих в системе. Если обнаруживается что-либо незнакомое, то необходимо выяснить, что это такое.

22.7.2.4 Измененные файлы

Когда злоумышленник успешно проникает в систему, он может попытаться изменить системные файлы для обеспечения продолжительного доступа к системе. Файлы, передаваемые в систему, обычно называются «rootkit», так как позволяют злоумышленнику осуществить доступ через корневую (root) учетную запись. В дополнение к таким программам, как снифферы, rootkit может содержать двоичные замещения для следующих файлов:

```
ftpd        passwd
inetd       ps
login       ssh
netstat     telnetd
```

Как правило, любой исполняемый файл, который может тем или иным образом помочь злоумышленнику поддерживать доступ, является кандидатом на замещение. Наилучший способ определить, был ли файл заменен - использовать криптографическую контрольную сумму. Лучше всего создавать контрольные суммы всех системных файлов при построении системы, после чего обновлять их при установке системных обновлений. Необходимо хранить контрольные суммы на безопасной системе, чтобы злоумышленник не мог изменить контрольные суммы при изменении файлов.

Если имеются подозрения нелегального проникновения в систему, пересчитайте контрольные суммы и сопоставьте их с исходными. Если они совпадают, то файлы изменены не были. Если же контрольные суммы различны, рассматриваемому файлу доверять не следует; его необходимо заменить оригиналом с установочного носителя.

Совет. По адресу http://www.chkrootkit.org/ можно найти утилиту, которая помогает в проверке наличия в системе rootkit-ов.

22.7.3 Общий алгоритм аудита системы Unix

Этот алгоритм покажет пути проверки систему Unix на ошибки в конфигурации или на наличие неизвестных процессов и учетных записей.

- Начните с системы Unix, к которой у вас имеется административный доступ (то есть у вас имеется пароль к корневой учетной записи этой системы) и на которой можно вносить изменения, не затрагивая рабочие приложения.
- Найдите файлы загрузки и определите, какие приложения запускаются при загрузке системы. Выявите приложения, которые являются необходимыми для системы, и отключите все остальные.
- Просмотрите файл inetd.conf и определите, какие службы включены. Определите службы, необходимые для системы, и отключите все остальные. Не забудьте выполнить команду kill -HUP для процесса inetd, чтобы перезапустить его с использованием новой конфигурации.
- Определите, используется ли в системе NFS. Внесите соответствующие изменения в файл dfstab.
- Если система использует telnet или FTP, загрузите TCP Wrappers и установите программу в системе. Настройте TCP Wrappers на разрешение доступа только к telnet и FTP, согласно требованиям системы.
- Найдите файл приветственного сообщения. Определите, используется ли корректное приветственное сообщение. Если это не так, разместите в системе корректное приветственное сообщение.

- Выясните, настроены ли в системе требуемые ограничения на пароли согласно политике безопасности организации. Если это не так, внесите соответствующие настройки.
- Определите, настроен ли в системе должным образом параметр umask по умолчанию. Если это не так, настройте umask соответствующим образом.
- Определите требования для входа через корневую учетную запись. Если администраторам требуется осуществлять вход сначала с использованием их собственного идентификатора (ID), настройте соответствующим образом конфигурацию системы.
- Проверьте систему на наличие неиспользуемых учетных записей. Все подобные учетные записи должны быть заблокированы.
- Установите в системе соответствующие обновления.
- Проверьте систему на некорректные пользовательские идентификаторы. В особенности следует искать учетные записи с UID, значение которого равно 0.
- Убедитесь в том, что в системе ведется журнал подозрительной активности, и что файл syslog.conf настроен соответствующим образом.
- Произведите в системе поиск скрытых файлов. Если будут найдены необычные скрытые файлы, исследуйте их, чтобы убедиться, что в систему никто не проник.
- Произведите поиск файлов SUID и SGID. Если будут обнаружены такие файлы, расположенные в каталогах пользователей, исследуйте их, чтобы убедиться, что в систему никто не проник.
- Произведите поиск файлов, общедоступных для записи. Если будут найдены такие файлы, либо устраните проблему посредством изменения разрешений (сначала выясните, для чего эти файлы используются), либо обратите на них внимание владельца.
- Проверьте сетевые интерфейсы на наличие любых неправильных настроек.
- Проверьте систему на предмет прослушиваемых (активных) портов. Если обнаружится какое-либо несоответствие, найдите процесс, использующий порт, и определите, должен ли данный процесс работать в системе.
- Проверьте таблицу процессов в системе и определите, выполняются ли какие-либо несоответствующие процессы.

22.8 Обновления системы

Для исправления ошибок и устранения уязвимостей для Unix выпускаются обновления и «патчи» аналогично тому, как это делается для операционных систем семейства Windows. Обновления должны устанавливаться регулярно, чтобы минимизировать число уязвимостей.

Различные поставщики систем Unix выпускают средства, помогающие в управлении обновлениями. Компания Sun предлагает программу Solaris Sunsolve Patch Manager, а Red Hat имеет онлайн-систему обновления в интернете (http://www.redhat.com/apps/support/errata/).

Примечание. При загрузке обновлений для систем Solaris имейте в виду, что Sun размещает многие обновления в кластере обновлений. Однако кластер обновлений может не содержать некоторых обновлений безопасности. Может понадобиться загрузить их в отдельном порядке и установить вручную.

23. БЕЗОПАСНОСТЬ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

23.1 Введение в защиту ПО

Безопасность программного обеспечения *(ПО) в широком смысле является свойством данного ПО функционировать без проявления различных негативных последствий для конкретной компьютерной системы.*

Под уровнем безопасности ПО *понимается вероятность того, что при заданных условиях в процессе его эксплуатации будет получен функционально пригодный результат.*

Причины, приводящие к функционально непригодному результату могут быть разными: сбои компьютерных систем, ошибки программистов и операторов, дефекты в ПО. При этом дефекты принято рассматривать двух типов: преднамеренные и непреднамеренные. Первые являются, как правило, результатом злоумышленных действий, вторые - ошибочных действий человека.

При исследовании проблем защиты ПО от преднамеренных дефектов неизбежна постановка следующих вопросов:

- кто потенциально может осуществить практическое внедрение программных дефектов деструктивного воздействия в исполняемый программный код;
- каковы возможные мотивы действий субъекта, осуществляющего разработку таких дефектов;
- как можно идентифицировать наличие программного дефекта;
- как можно отличить преднамеренный программный дефект от программной ошибки;
- каковы наиболее вероятные последствия активизации деструктивных программных средств при эксплуатации компьютерных систем (КС).

При ответе на первый вопрос следует отметить, что это: непосредственные разработчики алгоритмов и программ для компьютерных систем. Они хорошо знакомы с технологией разработки программных средств, имеют опыт разработки алгоритмов и программ для конкретных прикладных систем, знают тонкости существующей технологии отработки и испытаний программных компонентов и представляют особенности эксплуатации и целевого применения разрабатываемой КС.

Отметим, что алгоритмические и программные закладки могут быть реализованы в составе программного компонента вследствие следующих факторов:

- в результате инициативных злоумышленных действий непосредственных разработчиков алгоритмов и программ;

- в результате штатной деятельности специальных служб и организаций, а также отдельных злоумышленников;
- в результате применения инструментальных средств проектирования ПО, несущих вредоносное свойство автоматической генерации деструктивных программных средств.

Правомерно утверждать, что вредоносные программы, в отличие от широко применяемых электронных закладок, являются более изощренными объектами, обладающими большей скрытностью и эффективностью применения.

23.2 Угрозы безопасности ПО

Угрозы безопасности информации и программного обеспечения компьютерных систем возникают как в процессе их эксплуатации, так и при создании этих систем, что особенно характерно для процесса разработки ПО, баз данных и других информационных компонентов КС.

Наиболее уязвимы с точки зрения защищенности информационных ресурсов являются так называемые ***критические компьютерные системы****.*

Критические компьютерные системы *- сложные компьютеризированные организационно-технические и технические системы, блокировка или нарушение функционирования которых потенциально приводит к потере устойчивости организационных систем государственного управления и контроля, утрате обороноспособности государства, разрушению системы финансового обращения, дезорганизации систем энергетического и коммуникационно - транспортного обеспечения государства, глобальным экологическим и техногенным катастрофам.*

При решении проблемы повышения уровня защищенности информационных ресурсов КС необходимо исходить из того, что *наиболее вероятным информационным объектом воздействия будет выступать программное обеспечение, составляющее основу комплекса средств получения, семантической переработки, распределения и хранения данных, используемых при эксплуатации критических систем.*

Программные средства деструктивного воздействия по своей природе носят, вредоносный характер, а последствия их активизации и применения могут привести к значительному или даже непоправимому ущербу. *Такие вредоносные программы будем называть* ***разрушающими программными средствами (РПС)*****,** *а их обобщенная* ***классификация*** *может выглядеть следующим образом:*

- ***компьютерные вирусы*** - программы, способные размножаться, прикрепляться к другим программам, передаваться по линиям связи

и сетям передачи данных, проникать в электронные телефонные станции и системы управления и выводить их из строя;

- ***программные закладки*** *- программные компоненты, заранее внедряемые в компьютерные системы, которые по сигналу или в установленное время приводятся в действие, уничтожая или искажая информацию, или дезорганизуя работу программно-технических средств;*
- ***способы и средства, позволяющие внедрять компьютерные вирусы и программные закладки*** *в компьютерные системы и управлять ими на расстоянии*.

Под ***алгоритмической закладкой*** *будем понимать преднамеренное завуалированное искажение какой-либо части алгоритма решения задачи, либо построение его таким образом, что в результате конечной программной реализации этого алгоритма в составе программного компонента или комплекса программ, последние будут иметь ограничения на выполнение требуемых функций, заданных спецификацией, или вовсе их не выполнять при определенных условиях протекания вычислительного процесса*, задаваемого семантикой перерабатываемых программой данных. Кроме того, возможно появление у программного компонента функций, не предусмотренных прямо или косвенно спецификацией, и которые могут быть выполнены при строго определенных условиях протекания вычислительного процесса.

Под ***программной закладкой*** *будем понимать совокупность операторов и (или) операндов, преднамеренно в завуалированной форме включаемую в состав выполняемого кода программного компонента на любом этапе его разработки*. Программная закладка реализует определенный несанкционированный алгоритм с целью ограничения или блокирования выполнения программным компонентом требуемых функций при определенных условиях протекания вычислительного процесса, задаваемого семантикой перерабатываемых программным компонентом данных, либо с целью снабжения программного компонента не предусмотренными спецификацией функциями, которые могут быть выполнены при строго определенных условиях протекания вычислительного процесса.

Примеры уязвимостей ПО

Команда из четырех человек, работающих в разных университетах США, в 2008 году разработала **технологию автоматической генерации кода атаки на такую уязвимость в ПО, которая заранее неизвестна, а вычисляется путем сличения исходной и пропатченной версий программы**. Иначе говоря, инструкции для создания нового вредоносного

кода предоставляет, по сути, сама программная заплатка, выпущенная с целью латания очередной дыры.

Разработанная технология **APEG (Automatic Patch-based Exploit Generation)** позволяет за время от нескольких секунд до нескольких минут сгенерировать код атаки для большинства типов программных уязвимостей. Алгоритм APEG работает как доказательство корректности системы, проводимое в обратную сторону. Сначала выявляются различия в исполняемых кодах программы до и после применения заплатки, а затем по ее коду анализируется, для чего она предназначена. Патчи безопасности обычно содержат тест, который определенным образом ограничивает допустимые значения на входе системы, но существует процедура позволяющая пройти по коду и автоматически выявить набор входов, которые отлавливаются тестами нового патча. Когда это сделано, применяется специальный набор правил-эвристик для точной локализации места уязвимости, затем генерируется несколько вариантов кодов, потенциально способных эксплуатировать данную уязвимость, а тестовые испытания устанавливают, какой из кодов реально срабатывает.

По мнению разработчиков, это означает, что если корпорация Microsoft существенно не изменит способ распространения патчей среди клиентов, то последствия могут оказаться тяжелейшими. Ведь злоумышленники, заполучив в руки систему типа APEG, могут обнаружить уязвимость по свежевыпущенному патчу и провести атаку до того, как этот самый патч будет установлен на атакуемую машину.

Сотрудники Иллинойского университета (Урбана-Шампань) на конференции семинара по крупномасштабным и новым компьютерным угрозам (USENIX workshop on Large-Scale Exploits and Emergent Threats [LEET]) в 2008 году представили на удивление **эффективный подход к добавлению аппаратных закладок в компьютеры общего назначения**. Исследователи показали, что внесения в схему процессора совсем небольшого (одна-две тысячи) числа элементов достаточно для обеспечения широкого спектра дистанционных атак, которые невозможно выявить или предотвратить с помощью традиционных софтверных подходов к безопасности. Правда, для проведения подобных атак требуется фундаментально скомпрометировать компьютеры на этапе их создания или сборки. Это вполне по силам государственным спецслужбам.

Технически это выглядит так. Скрытые в процессоре вредоносные схемы обеспечивают атакующую сторону невидимым внутренним плацдармом для атак. Поскольку такие схемы занимают уровень, находящийся ниже стека программ, они способны обходить все традиционные техники защиты.

В работе представлена общая конструкция и конкретные формы реализации так называемых IMPs (Illinois Malicious Processors, «иллинойских вредоносных процессоров»). Показано, что даже с учетом жестких

ограничений по месту, его все равно достаточно для планирования разнообразных типов атак, а не одной узконаправленной. Такая гибкость схемы позволила разработчикам продемонстрировать две конкретные конструкции и реализовать их практически в конкретной системе FPGA-чипа, то есть процессора с перепрограммируемой логикой.

Вот примеры, подтверждающие общую концепцию.

Эскалация привилегий. Используя механизм доступа к памяти, реализован вредоносный сервис, поднимающий привилегии пользовательского процесса до высшего (root) уровня. При выполнении такой атаки программа эскалации привилегий использует аппаратную закладку в процессоре для отключения защиты привилегированных областей памяти. Он позволяет напрямую нарушать все предположения ОС относительно обеспечиваемой защиты памяти.

Входной бэкдор. Используя специально разработанный механизм теневого режима, разработчики реализовали вредоносный сервис, служащий постоянным «черным ходом» в систему. Чтобы начать атаку захвата, злоумышленник посылает сетевой пакет в систему жертвы, где ОС первым делом инспектирует этот пакет, проверяя контрольную сумму UDP. Сам акт проверки пакета (необходимый для принятия решения о том, следует ли его отвергнуть) запускает троянскую закладку в железе, а вредоносная программа интерпретирует содержимое пакета как новый код прошивки, который невидимо загружается в процессор. Операционная же система тем временем отбрасывает непрошеный пакет и продолжает работу, не заметив атаки.

Код прошивки, реализующий теневой режим, отслеживает login-приложение для входа в систему. И когда некто пытается войти с особым, заранее известным закладке паролем, та подменяет значение функции проверки пароля на «правильный» и тем самым гарантирует доступ в систему любому, кто знает хитрость. Чтобы скрыть следы атаки, сразу после успешной попытки логина прошивка сама себя выгружает и отключает теневой режим, возвращая системе все ресурсы процессора. Таким образом, послав сетевой UDP-пакет и тут же войдя в систему, злоумышленник может сократить время работы теневого режима до минимума. Если же система жертвы не имеет выхода в сеть, то для включения закладки-бэкдора можно использовать похожий механизм на основе внешнего накопителя. Например, в USB-модуле флэш-памяти для этого подходит самый первый блок, необходимый для идентификации типа файловой системы. Механизм теневого режима увеличивает количество логических гейтов схемы всего на 0,08%, давая при этом неограниченный доступ к компьютеру без опоры на какие-либо программные уязвимости.

Похищение паролей. С помощью того же механизма теневого режима можно реализовать сервис, ворующий пароли доступа у легитимных пользователей системы. Главная трудность здесь - отыскание паролей в гигантских массивах случайных данных. Но и эта задача вполне разрешима,

коль скоро в символьных строках кода, относящегося к записи и считыванию паролей, присутствует слово Password. В развитие этой же темы исследователи продемонстрировали и два существенно разных способа для скрытного слива похищенных паролей в сеть - как на уровне ОС, так и на уровне прямой модификации пакетов.

Подводя итог, иллинойские исследователи без ложной скромности отмечают, что им удалось заложить фундаментальные основы конструирования процессоров с аппаратными закладками, способными обеспечивать весьма сложные и продвинутые атаки для тех, кто владеет секретами конструкции. Сделано же это, по словам разработчиков, дабы продемонстрировать, что при нынешней организации поставок микросхем заказчикам имеются все предпосылки для злоупотреблений. То есть заинтересованные структуры, обладающие компетентными специалистами и надлежащими ресурсами, вполне способны разрабатывать и внедрять вредоносные микросхемы с аппаратными закладками.

Действия алгоритмических и программных закладок условно можно разделить на три класса:

- *изменение функционирования вычислительной системы (сети),*
- *несанкционированное считывание информации,*
- *несанкционированная модификация информации, вплоть до ее уничтожения.*

Следует отметить, что указанные классы воздействий могут пересекаться.

Класс воздействий **«изменение функционирования вычислительной системы»** направлен на:

- уменьшение скорости работы вычислительной системы (сети);
- частичное или полное блокирование работы системы (сети);
- имитация физических (аппаратурных) сбоев работы вычислительных средств и периферийных устройств;
- переадресация сообщений;
- обход программно-аппаратных средств криптографического преобразования информации;
- обеспечение доступа в систему с непредусмотренных периферийных устройств.

Несанкционированное считывание информации, осуществляемое в автоматизированных системах, направлено на:

- считывание паролей и их отождествление с конкретными пользователями;
- получение секретной информации;
- идентификацию информации, запрашиваемой пользователями;
- подмену паролей с целью доступа к информации;

- контроль активности абонентов сети для получения косвенной информации о взаимодействии пользователей и характере информации, которой обмениваются абоненты сети.

Несанкционированная модификация информации является наиболее опасной разновидностью воздействий программных закладок, поскольку приводит к наиболее опасным последствиям. В этом классе воздействий можно выделить следующие:

- разрушение данных и кодов исполняемых программ внесение тонких, трудно обнаруживаемых изменений в информационные массивы;
- внедрение программных закладок в другие программы и подпрограммы (вирусный механизм воздействий);
- искажение или уничтожение собственной информации сервера и тем самым нарушение работы сети;
- модификация пакетов сообщений.

С точки зрения времени внесения программных закладок в программы их можно разделить на две категории:

1. **априорные**, то есть закладки, внесенные при разработке ПО (или «врожденные»)
2. **апостериорные**, закладки, внесенные при испытаниях, эксплуатации или модернизации ПО (или «приобретенные»).

Хотя последняя разновидность закладок и относятся больше к проблеме обеспечения эксплуатационной, а не технологической безопасности ПО, однако методы тестирования программных комплексов, вероятностные методы расчета наличия программных дефектов и методы оценивания уровня безопасности ПО могут в значительной мере пересекаться и дополнять друг друга. Тем более что действие программной закладки после того как она была внесена в ПО либо на этапе разработки, либо на последующих этапах жизненного цикла ПО, практически не будет ничем не отличаться.

23.3 Разрушающие программные средства

Необходимым условием для отнесения программы к классу разрушающих программных средств *является наличие в ней процедуры нападения, которую можно определить как процедуру нарушения целостности вычислительной среды,* поскольку объектом нападения РПС всегда выступает элемент этой среды.

При этом необходимо учитывать два фактора:

1. любая прикладная программа, не относящаяся к числу РПС, потенциально может содержать в себе алгоритмические ошибки,

появление которых при ее функционировании приведет к непреднамеренному разрушению элементов вычислительной среды;
2. любая прикладная или сервисная программа, ориентированная на работу с конкретными входными данными может нанести непреднамеренный ущерб элементам операционной или вычислительной среды в случае, когда входные данные либо отсутствуют, либо не соответствуют заданным форматам их ввода в программу.

Для устранения указанной неопределенности по отношению к испытываемым программам следует исходить из предположения, что процедура нарушения целостности вычислительной среды введена в состав ПО умышленно.

*Кроме условия необходимости, целесообразно ввести **условия достаточности**, которые обеспечат возможность описания РПС различных классов:*

- *достаточным условием для отнесения РПС к классу компьютерных **вирусов** является наличие в его составе процедуры **саморепродукции**;*
- *достаточным условием для отнесения РПС к классу **средств несанкционированного доступа** являются наличие в его составе **процедуры преодоления защиты и отсутствия процедуры саморепродукции**;*
- *достаточным условием для отнесения РПС к классу **программных закладок** является **отсутствие** в его составе процедур саморепродукции и преодоления защиты.*

Предполагается наличие в РПС следующего набора возможных функциональных элементов:

- *процедуры захвата (получения) управления;*
- *процедуры самомодификации («мутации»);*
- *процедуры порождения (синтеза);*
- *процедуры маскировки (шифрования).*

Этих элементов достаточно для построения обобщенной концептуальной модели РПС, которая отражает возможную структуру (на семантическом уровне) основных классов РПС.

23.4 Модель угроз и принципы обеспечения безопасности ПО

Модель угроз технологической безопасности ПО должна представлять собой официально принятый нормативный документ, которым должен руководствоваться заказчики и разработчики программных комплексов.

Модель угроз должна включать:

полный реестр типов возможных программных закладок;

описание наиболее технологически уязвимых мест компьютерных систем (с точки зрения важности и наличия условий для скрытого внедрения программных закладок);
описание мест и технологические карты разработки программных средств, а также критических этапов, при которых наиболее вероятно скрытое внедрение программных закладок;
реконструкцию замысла структур, имеющих своей целью внедрение в ПО заданного типа (класса, вида) программных закладок диверсионного типа;
психологический портрет потенциального диверсанта в компьютерных системах .

На базе утвержденной модели угроз технологической безопасности должна разрабатываться прикладная модель угроз безопасности для каждого конкретного компонента защищаемого комплекса средств автоматизации КС. В основе этой разработки должна лежать схема угроз представленная на рис. 23.1.

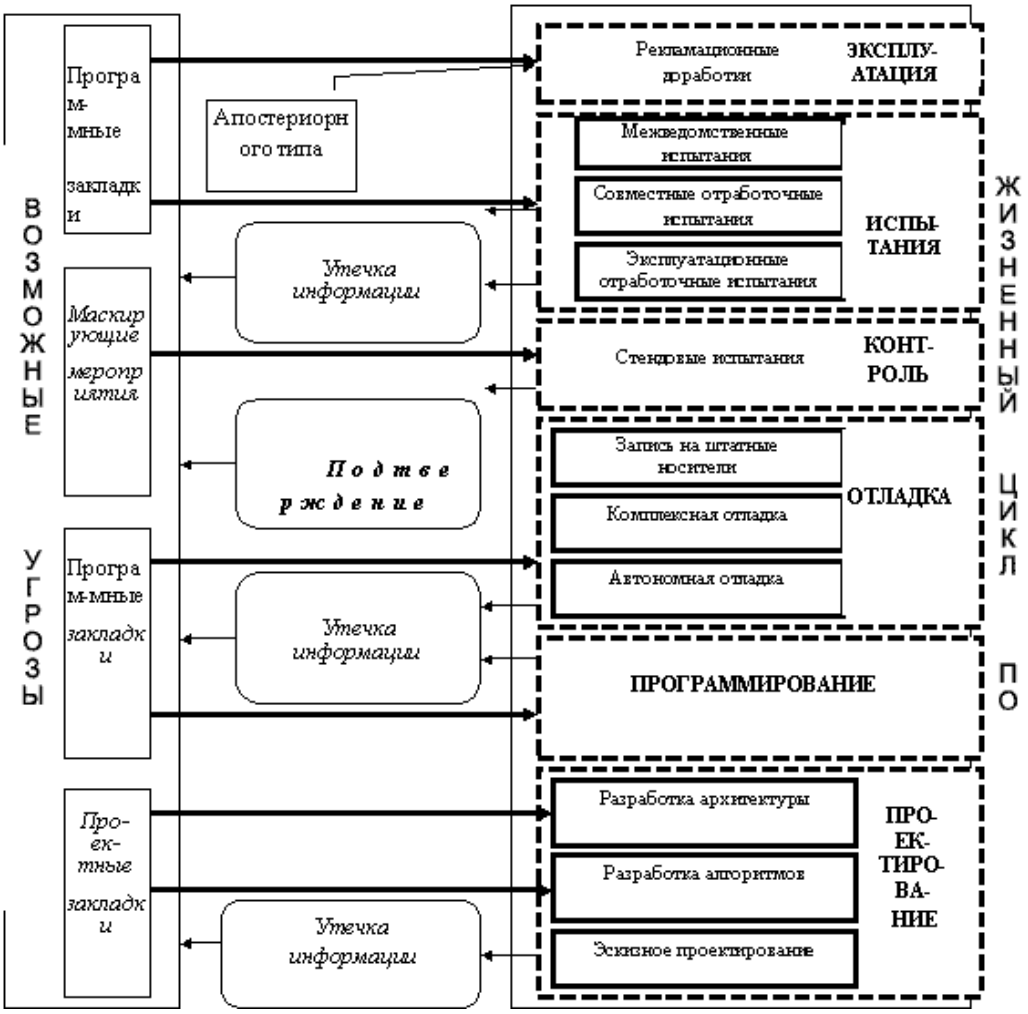


Рис. 23.1 - Прикладная модель угроз безопасности

Наполнение модели технологической безопасности ПО должно включать в себя следующие элементы:

- *матрицу чувствительности КС к «вариациям» ПО (то есть к появлению искажений),*
- *энтропийный портрет ПО (то есть описание «темных» запутанных участков ПО),*
- *реестр камуфлирующих условий для конкретного ПО,*
- *справочные данные о разработчиках*
- *реальный (либо реконструированный) замысел злоумышленников по поражению этого ПО.*

В таблице 23.1 приведен пример указанной типовой модели для сложных программных комплексов.

Таблица 23.1. Пример типовой модели технологической безопасности ПО для сложных программных комплексов

№	**Угрозы рассматриваемые в модели безопасности в соответсвии с жизненным циклом ПО**
1	ПРОЕКТИРОВАНИЕ
1.1	*Проектные решения* - Злоумышленный выбор нерациональных алгоритмов работы Облегчение внесения закладок и затруднение их обнаружения. - Внедрение злоумышленников в коллективы, разрабатывающие наиболее ответственные части ПО.
1.2	*Используемые информационные технологии* - Внедрение злоумышленников, в совершенстве знающих «слабые» места и особенности используемых технологий. Внедрение информационных технологий или их элементов, содержащих программные закладки. - Внедрение неоптимальных информационных технологий.
1.3	*Используемые аппаратно-технические средства* - Поставка вычислительных средств, содержащих программные, аппаратные или программно-аппаратные закладки. - Поставка вычислительных средств с низкими реальными характеристиками. - Поставка вычислительных средств, имеющих высокий уровень экологической опасности. - Задачи коллективов разработчиков и их персональный состав. - Внедрение злоумышленников в коллективы разработчиков программных и аппаратных средств. - Вербовка сотрудников путем подкупа, шантажа и т.п.

2	КОДИРОВАНИЕ
2.1	*Архитектура программной системы, взаимодействие ее с внешней средой и взаимодействие подпрограмм программной системы* - Доступ к «чужим» подпрограммам и данным. - Нерациональная организация вычислительного процесса. - Организация динамически формируемых команд или параллельных вычислительных процессов. - Организация переадресации команд, запись злоумышленной информации в используемые программной системой или другими программами ячейки памяти.
2.2	*Функции и назначение кодируемой части программной системы, взаимодействие этой части с другими подпрограммами* - Формирование программной закладки, воздействующей на другие части программной системы. - Организация замаскированного спускового механизма программной закладки. Формирование программной закладки, изменяющей структуру программной системы.
2.3	*Технология записи программного обеспечения и исходных данных* - Поставка программного обеспечения и технических средств со встроенными дефектами.
3	ОТЛАДКА И ИСПЫТАНИЯ
3.1	*Назначение, функционирование, архитектура программной системы* - Встраивание программной закладки как в отдельные подпрограммы, так и в управляющую программу программной системы. - Формирование программной закладки с динамически формируемыми командами. - Организация переадресации отдельных команд программной системы.
3.2	*Сведения о процессе испытаний (набор тестовых данных, используемые вычислительные средства, подразделения и лица, проводящие испытания, используемые модели* - Формирование набора тестовых данных, не позволяющих выявить программную закладку. - Поставка вычислительных средств, содержащих программные, аппаратные или программно-аппаратные закладки. - Формирование программной закладки, не обнаруживаемой с помощью используемой модели объекта в силу ее неадекватности описываемому объекту. - Вербовка сотрудников коллектива, проводящих испытания.

4	КОНТРОЛЬ
4.1	*Используемые процедуры и методы контроля* - Формирование спускового механизма программной закладки, не включающего ее при контроле на безопасность. - Маскировка программной закладки путем внесения в программную систему ложных «непреднамеренных» дефектов. - Формирование программной закладки в ветвях программной системы, не проверяемых при контроле. - Формирование «вирусных» программ, не позволяющих выявить их внедрение в программную систему путем контрольного суммирования. - Поставка программного обеспечения и вычислительной техники, содержащих программные, аппаратные и программно-аппаратные закладки.
5	ЭКСПЛУАТАЦИЯ
5.1	*Сведения о персональном составе контролирующего подразделения и испытываемых программных системах* - Внедрение злоумышленников в контролирующее подразделение. - Вербовка сотрудников контролирующего подразделения. - Сбор информации о испытываемой программной системе.
5.2	*Сведения об обнаруженных при контроле программных закладках* - Разработка новых программных закладок при доработке программной системы.
5.3	*Сведения об обнаруженных незлоумышленных дефектах и программных закладках*
5.4	*Сведения о доработках программной системы и подразделениях, их осуществляющих*
5.5	*Сведения о среде функционирования программной системы и ее изменениях*
5.6	*Сведения о функционировании программной системы, доступе к ее загрузочному модулю и исходным данным, алгоритмах проверки сохранности программной системы и данных*

23.5 Элементы модели угроз эксплуатационной безопасности ПО

Анализ угроз эксплуатационной безопасности ПО КС позволяет, разделить их на два типа:

1. случайные;
2. преднамеренные:
   - активные,
   - пассивные.

Активные угрозы направлены на изменение технологически обусловленных алгоритмов, программ функциональных преобразований или информации, над которой эти преобразования осуществляются.

Пассивные угрозы ориентированы на нарушение безопасности информационных технологий без реализации таких модификаций.

Вариант общей структуры набора потенциальных угроз безопасности информации и ПО на этапе эксплуатации КС приведен в табл. 23.2.

Рассмотрим основное содержание данной таблицы. Угрозы, носящие случайный характер и связанные с отказами, сбоями аппаратуры, ошибками операторов и т.п. предполагают нарушение заданного собственником информации алгоритма, программы ее обработки или искажение содержания этой информации. Субъективный фактор появления таких угроз обусловлен ограниченной надежностью работы человека и проявляется в виде ошибок (дефектов) в выполнении операций формализации алгоритма функциональных преобразований или описания алгоритма на некотором языке, понятном вычислительной системе.

Угрозы, носящие злоумышленный характер вызваны, как правило, преднамеренным желанием субъекта осуществить несанкционированные изменения с целью нарушения корректного выполнения преобразований, достоверности и целостности данных, которое проявляется в искажениях их содержания или структуры, а также с целью нарушения функционирования технических средств в процессе реализации функциональных преобразований и изменения конструктива вычислительных систем и систем телекоммуникаций.

На основании анализа уязвимых мест и после составления полного перечня угроз для данного конкретного объекта информационной защиты, например, в виде указанной таблицы, необходимо осуществить переход к неформализованному или формализованному описанию модели угроз эксплуатационной безопасности ПО КС. Такая модель, в свою очередь, должна соотноситься (или даже являться составной частью) обобщенной модели обеспечения безопасности информации и ПО объекта защиты.

После окончательного синтеза модели угроз разрабатываются практические рекомендации и методики по ее использованию для конкретного объекта информационной защиты, а также механизмы оценки адекватности модели и реальной информационной ситуации и оценки эффективности ее применения при эксплуатации КС.

Таблица 23.2. Вариант общей структуры набора потенциальных угроз безопасности информации и ПО на этапе эксплуатации КС

Угрозы нарушения безопасности ПО	Несанкционированные действия		
	Случайные	Преднамеренные	
		Пассивные	Активные
Прямые	невыявленные ошибки программного обеспечения КС; отказы и сбои технических средств КС; ошибки операторов; неисправность средств шифрования; скачки электропитания на технических средствах; старение носителей информации; разрушение информации под воздействием физических факторов(аварии и т.п.).	маскировка несанкционированных запросов под запросы ОС; обход программ разграничения доступа; чтение конфиденциальных данных из источников информации; подключение к каналам связи с целью получения информации («подслушивание» и/или «ретрансляция»); при анализе трафика использование терминалов и ЭВМ других операторов; намеренный вызов случайных факторов	включение в программы РПС, выполняющих функции нарушения целостности и конфиденциальности информации и ПО; ввод новых программ, выполняющих функции нарушения безопасности ПО; незаконное применение ключей разграничения доступа; обход программ разграничения доступа; вывод из строя подсистемы регистрации и учета; уничтожение ключей шифрования и паролей; подключение к каналам связи с целью модификации, уничтожения, задержки и переупорядочивания данных; вывод из строя элементов физических средств защиты информации КС; намеренный вызов случайных факторов.
Косвенные	нарушение пропускного режима и режима секретности; естественные потенциальные поля; помехи и т.п.	перехват ЭМИ от технических средств; хищение производственных отходов(распечаток);визуальный канал; подслушивающие устройства; дистанционное фотографирование и т.п	помехи; отключение электропитания; намеренный вызов случайных факторов.

23.6 Основные принципы обеспечения безопасности ПО на различных стадиях его жизненного цикла

В качестве объекта обеспечения технологической и эксплуатационной безопасности ПО рассматривается вся совокупность его компонентов в рамках конкретной КС. В качестве доминирующей должна использоваться стратегия сквозного тотального контроля технологического и эксплуатационного этапов жизненного цикла компонентов ПО.

23.6.1 Обеспечение безопасности при обосновании, планировании работ и проектном анализе ПО

Принципы обеспечения безопасности на этапах обоснования, планирования работ и проектном анализе ПО включают следующие принципы.

Комплексности обеспечения безопасности ПО, предполагающей рассмотрение проблемы безопасности информационно - вычислительных процессов с учетом всех структур КС, возможных каналов утечки информации и несанкционированного доступа к ней, времени и условий их возникновения, комплексного применения организационных и технических мероприятий.

Планируемости применения средств безопасности программ, предполагающей перенос акцента на совместное системное проектирование ПО и средств его безопасности, планирование их использования в предполагаемых условиях эксплуатации.

Обоснованности средств обеспечения безопасности ПО, заключающейся в глубоком научно-обоснованном подходе к принятию проектных решений по оценке степени безопасности, прогнозированию угроз безопасности, всесторонней априорной оценке показателей средств защиты.

Достаточности безопасности программ, отражающей необходимость поиска наиболее эффективных и надежных мер безопасности при одновременной минимизации их стоимости.

Гибкости управления защитой программ, требующей от системы контроля и управления обеспечением информационной безопасности ПО способности к диагностированию, опережающей нейтрализации, оперативному и эффективному устранению возникающих угроз в условиях резких изменений обстановки информационной борьбы.

Заблаговременности разработки средств обеспечения безопасности и контроля производства ПО, заключающейся в предупредительном характере мер обеспечения технологической безопасности работ в интересах недопущения снижения эффективности системы безопасности процесса создания ПО.

Документируемости технологии создания программ, подразумевающей разработку пакета нормативно-технических документов по контролю программных средств на наличие преднамеренных дефектов.

23.6.2 Обеспечение безопасности ПО в процессе его разработки

Принципы обеспечения безопасности ПО в процессе его разработки включают следующие принципы.

Регламентации технологических этапов разработки ПО, включающей упорядоченные фазы промежуточного контроля, спецификацию программных модулей и стандартизацию функций и формата представления данных.

Автоматизации средств контроля управляющих и вычислительных программ на наличие дефектов, создания типовой общей информационной базы алгоритмов, исходных текстов и программных средств, позволяющих выявлять преднамеренные программные дефекты.

Последовательной многоуровневой фильтрации программных модулей в процессе их создания с применением функционального дублирования разработок и поэтапного контроля.

Типизации алгоритмов, программ и средств информационной безопасности, обеспечивающей информационную, технологическую и программную совместимость, на основе максимальной их унификации по всем компонентам и интерфейсам.

Централизованного управления базами данных проектов ПО и администрирование технологии их разработки с жестким разграничением функций, ограничением доступа в соответствии со средствами диагностики, контроля и защиты.

Блокирования несанкционированного доступа соисполнителей и абонентов государственных сетей связи, подключенных к стендам для разработки программ.

Статистического учета и ведения системных журналов о всех процессах разработки ПО с целью контроля технологической безопасности.

Использования только сертифицированных и выбранных в качестве единых инструментальных средств разработки программ для новых технологий обработки информации и перспективных архитектур вычислительных систем.

23.6.3 Обеспечение безопасности ПО на этапах стендовых и приемо-сдаточных испытаний

Принципы обеспечения безопасности ПО на этапах стендовых и приемо-сдаточных испытаний включают принципы.

Тестирования ПО на основе разработки комплексов тестов, параметризуемых на конкретные классы программ с возможностью

функционального и статистического контроля в широком диапазоне изменения входных и выходных данных.

Проведения натурных испытаний программ при экстремальных нагрузках с имитацией воздействия активных дефектов.

Осуществления «фильтрации» программных комплексов с целью выявления возможных преднамеренных дефектов определенного назначения на базе создания моделей угроз и соответствующих сканирующих программных средств.

Разработки и экспериментальной отработки средств верификации программных изделий.

Проведения стендовых испытаний ПО для определения непреднамеренных программных ошибок проектирования и ошибок разработчика, приводящих к невыполнению целевых функций программ, а также выявление потенциально «узких» мест в программных средствах для разрушительного воздействия.

Отработки средств защиты от несанкционированного воздействия нарушителей на ПО.

Сертификации программных изделий АСУ по требованиям безопасности с выпуском сертификата соответствия этого изделия требованиям технического задания.

23.6.4 Обеспечение безопасности при эксплуатации ПО

Принципы обеспечения безопасности при эксплуатации ПО включают следующие принципы.

Сохранения и ограничения доступа к эталонам программных средств, недопущение внесения изменений в них.

Профилактического выборочного тестирования и полного сканирования программных средств на наличие преднамеренных дефектов.

Идентификации ПО на момент ввода его в эксплуатацию в соответствии с предполагаемыми угрозами безопасности ПО и его контроль.

Обеспечения модификации программных изделий во время их эксплуатации путем замены отдельных модулей без изменения общей структуры и связей с другими модулями.

Строгого учета и каталогизации всех сопровождаемых программных средств, а также собираемой, обрабатываемой и хранимой информации.

Статистического анализа информации обо всех процессах, рабочих операциях, отступлениях от режимов штатного функционирования ПО.

Гибкого применения дополнительных средств защиты ПО в случае выявления новых, непрогнозируемых угроз информационной безопасности.

23.7 Методы и средства анализа безопасности ПО

Широко известны различные средства программного обеспечения обнаружения элементов РПС - от простейших антивирусных программ-сканеров до сложных отладчиков и дизассемблеров - анализаторов и именно на базе этих средств и выработался набор методов, которыми осуществляется анализ безопасности ПО.

В целом полный процесс анализа ПО включает в себя три вида анализа:

1. *лексический верификационный анализ;*
2. *синтаксический верификационный анализ;*
3. *семантический анализ программ.*

Каждый из видов анализа представляет собой законченное исследование программ согласно своей специализации.

Лексический верификационный анализ предполагает поиск распознавания и классификацию различных лексем объекта исследования (программа), представленного в исполняемых кодах. При этом лексемами являются сигнатуры. Поиск лексем (сигнатур) реализуется с помощью специальных программ-сканеров. В данном случае осуществляется поиск сигнатур следующих классов:

- сигнатуры вирусов;
- сигнатуры элементов РПС;
- сигнатуры (лексемы) «подозрительных функций»;
- сигнатуры штатных процедур использования системных ресурсов и внешних устройств.

Синтаксический верификационный анализ предполагает поиск, распознавание и классификацию синтаксических структур РПС, а также построение структурно-алгоритмической модели самой программы.

Семантический анализ предполагает исследование программы изучения смысла составляющих ее функций (процедур) в аспекте операционной среды компьютерной системы. В отличие от предыдущих видов анализа, основанных на статическом исследовании, семантический анализ нацелен на изучение динамики программы - ее взаимодействия с окружающей средой.

Методы анализа безопасности программного обеспечения делят на:

1. *Контрольно-испытательные методы анализа,*
2. *Логико-аналитические методы контроля безопасности программ.*

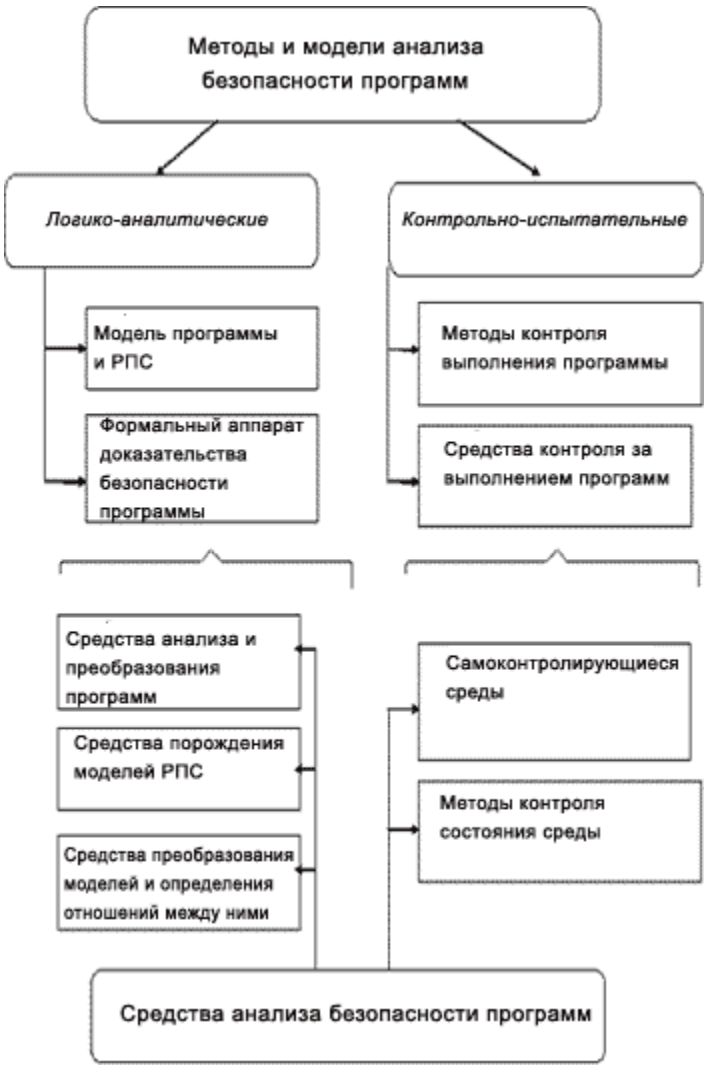


Рис. 23.2 - Методы и средства анализа безопасности ПО

Контрольно-испытательные методы *- это методы, в которых критерием безопасности программы служит факт регистрации в ходе тестирования программы нарушения требований по безопасности, предъявляемых в системе предполагаемого применения исследуемой программы*.

Контрольно-испытательные методы делятся на:

- те, в которых контролируется процесс выполнения программы
- те, в которых отслеживаются изменения в операционной среде, к которым приводит запуск программы. Эти методы наиболее распространены, так как они не требуют формального анализа, позволяют использовать имеющиеся технические и программные средства и быстро ведут к созданию готовых методик.

Рис.23.3 - Схема анализа безопасности ПО с помощью контрольно-испытательных методов

Рис. 23.4 - Схема анализа безопасности ПО с помощью логико-аналитических методов

*При проведении анализа безопасности с помощью **логико-аналитических методов** (см. рис.23.4) строится модель программы и формально доказывается эквивалентность модели исследуемой программы и модели РПС.*

Заключение

Проблема обеспечения информационной безопасности становится все более актуальной для российских компаний. Это связано и с обострением конкурентной борьбы на внутренних рынках, и с выходом компаний на международный уровень. Многие из них уже не могут обеспечить защиту коммерческой информации собственными силами и вынуждены пользоваться услугами профильных профессиональных IT-консультантов.

Обеспечение информационной безопасности является не только российской, но и мировой проблемой. Так в первые годы внедрения корпоративных локальных сетей головной болью компаний был несанкционированный доступ к коммерческой информации путем внешнего взлома (хакерской атаки). Сейчас с точки зрения информационной безопасности многие компании напоминают крепости, окруженные несколькими периметрами мощных стен – программными и аппаратными платформами ИБ. Однако практика показывает, что информация все равно утекает. При этом основной предпосылкой к утечке информации являются отсутствие единого системного подхода к обеспечению ИБ в компаниях.

В течение многих лет компании отчаянно боролись с вирусными эпидемиями, обносили периметр межсетевыми экранами и системами предотвращения вторжений, внедряли мощные инструменты против неавторизованного доступа. Однако компании упустили из вида главную опасность. Осутствие единой политики информационной безопастности, а также единой концепции построения профиля информационной защиты компании зачастую обесценивает многомиллионные затраты на программные и аппаратные комплексы ИБ. Еще пару лет назад IT-службы отвечали за защиту от внешних угроз, а с внутренними угрозами разбиралась служба безопасности. Сегодня она просто физически не может контролировать перемещение информации по электронным сетям и с помощью переносных носителей. Для этого нужны специально разработанные регламенты, ликбез сотрудников, специально подготовленные сотрудники безопасности и технические средства для выявления попыток несанкционированного доступа или перемещения информации. Все эти меры должны реализоваться специалистом ИБ в рамках единой концепции.

Бурное развитие IT технологий и направления ИБ приводит к росту спроса на профессиональных специалистов в данной области. Это актуализирует получение образования в области ИБ и широкой востребованости полученных профильных знаний на рынке труда.

Хочется надеятся, что данное учебное пособие поможет будущим профессионалам в сфере IT получить тот общий набор знаний и умений в области ИБ, чтобы оказаться востребованными и высокооплачиваемыми сотрудниками престижных компаний.

Список использованных источников

1. Галатенко В. А. Основы информационной безопасности**.** – М.: Интернет-университет информационных технологий - www.INTUIT.ru, 2008. – 208 с.

2. Астахов А. Анализ защищенности корпоративных автоматизированных систем // Jet Info [Эл. ресурс] – URL: www.jetinfo.ru\2002\7\1\article1.7.2002.html

3. Доля А. Внутренние угрозы ИТ-безопасности. // Byte-Россия [Эл. ресурс] – N 12, 2004. – URL: www.bytemag.ru/?ID=603365

4. Доля А. Внутренние ИТ-угрозы в России 2006 // КомпьютерПресс N 5, 2007.

5. Грудзаев С. Полезные мелочи - Aladdin Security Solution // LAN [Эл. ресурс] – URL: http://www.osp.ru/lan/2008/05/5068377/

6. Романец Ю.В., Тимофеев П.А., Шаньгин В.Ф. Защита информации в компьютерных системах и сетях / Под ред.В.Ф. Шаньгина.-2-е изд., перераб. и доп. - М.: Радио и связь, 2001. - 376 с.

7. Галатенко В. А. Стандарты информационной безопасности. – М.: Интернет-университет информационных технологий - www.INTUIT.ru, 2005. - 264 с.

8. Lonely R. Алгоритм шифрования данных с открытым ключом RSA. [Эл. ресурс] – URL: www.rusdoc.ru/material/raznoe/rsa.shtml

9. Лапонина О. Р. Криптографические основы безопасности: курс лекций для Интернет-университета информационных технологий.

10. Антивирусная защита компьютерных систем: курс лекций для Интернет-университета информационных технологий от лаборатории Касперского – М.: Интернет-университет информационных технологий - www.INTUIT.ru, 2007. – URL: www.intuit.ru/department/security/antiviruskasp/

11. Вирусы и средства борьбы с ними: курс лекций для Интернет-университета информационных технологий от лаборатории Касперского – М.: Интернет-университет информационных технологий - www.INTUIT.ru, 2007. – URL: www.intuit.ru/department/security/viruskasper/

12. Атака через Интернет / Медведовский И. Д., Семьянов П. В., Платонов В. В.; под ред. П. Д. Зегжды. - СПб.: изд. НПО «Мир и семья-95», 1997.

13. Мэйволд Э. Безопасность сетей: курс лекций для Интернет-университета информационных технологий – М.: Интернет-университет информационных технологий - www.INTUIT.ru, 2006. – URL: www.intuit.ru/department/security/netsec/

14. Кобб М. Джост М. Безопасность IIS: курс лекций для Интернет-университета информационных технологий – М.: Интернет-университет информационных технологий - www.INTUIT.ru, 2006. – URL: www.intuit.ru/department/internet/iissecurity/

15. Бейс Р. Введение в обнаружение атак и анализ защищенности // НИП «Информзащита» [Эл. ресурс] - URL: http://bugtraq.ru/library/books/icsa/

16. Семенов Ю. А. Процедуры, диагностики и безопасность в Интернет: курс лекций для Интернет-университета информационных технологий – М.: Интернет-университет информационных технологий - www.INTUIT.ru, 2007. – URL: www.intuit.ru/department/network/pdsi/

17. Пировских А. Взлом WPA // TNG.ru [Эл. ресурс] - URL: www.thg.ru/network/20050806/print.html

18. Таранов А., Слепов О. Безопасность систем электронной почты // Jet Info [Эл. ресурс] - № 6, 2003. – URL: www.citforum.ru/security/internet/email/article1.6.2003.html#AEN11

19. Иржавский А. Безопасность электронной почты // CIO - № 8, 2003. – URL: offline.cio-world.ru/2003/18/29383/index.html

20. Карпов В. Е., Коньков К. А. Основы операционных систем. – М.: Интернет-университет информационных технологий - www.INTUIT.ru, 2005. - 536 с.

21. Коньков К. А. Устройство и функционирование ОС Windows. – М.: Интернет-университет информационных технологий - ИНТУИТ.ру, БИНОМ. Лаборатория знаний, 2008. - 208 с.

22. Коньков К. А. Основы организации операционных систем Microsoft Windows: курс лекций для Интернет-университета информационных технологий – М.: Интернет-университет информационных технологий - www.INTUIT.ru, 2007. – URL: www.intuit.ru/department/os/osmswin/

23. Казарин О. В. Безопасность программного обеспечения компьютерных систем. – М.: МГУЛ, 2003. – 212 с. – URL: www.citforum.ru/security/articles/kazarin/#1-5

24. Малюк А. А. Информационная безопасность: концептуальные и методологические основы защиты информации. Уч. пособие для вузов. – М.: Горячая линия – Телеком, 2004. – 280 с.

25. Основы информационной безопасности. Учебное пособие для вузов / Е. Б. Белов, В. П. Лось, Р. В. Мещериков, А. А. Шелупанов. – М.: Горячая линия – Телеком, 2006. – 544 с.

26. Грушо А. А. Тимонина Е. Е. Теоретические основы защиты информации. - М.: Изд. агентства «Яхтсмен», 1996. – 72 с.

27. Хорошко В. А., Чекатков А. А. Методы и средства защиты информации. – М.: Юниор, 2003. – 504 с.

28. Ярочкин В. И. Информационная безопасность. Учебное пособие для студентов непрофильных вузов. — М.: Междунар. отношения, 2000. — 400 с.

29. Корнюшин П. Н. Костерин С. С. Информационная безопасность. – Владивосток: ДВГУ, 2003. – 155 с.

30. Халяпин Д. Б. Защита информации. Вас подслушивают? Защищайтесь! – М.: НОУ ШО «Баярд», 2004. – 432 с.

31. Щеглов А. Ю. Защита компьютерной информации от несанкционированного доступа. – СПб.: Наука и Техника, 2004. – 384 с.

32. Шнайер Б. Секреты и ложь. Безопасность данных в цифровом мире. – СПб.: Питер, 2003. – 368 с.

33. Шнайер Б. Прикладная криптография: Протоколы, алгоритмы, исходные тексты на языке Си. 2-е издание. – М.: Триумф, 2002. – 595 с.

34. Ященко В. В. и др. Ведение в криптографию // CIT Forum – URL: www.citforum.ru\security\cryptography\yaschenko\

35. Будко В. Н. Информационная безопасность и защита информации. Конспект лекций. – Воронеж: ВГУ, 2003. – 86 с.

36. Василенко О. Н. Теоретико-числовые алгоритмы в криптографии. — М.: МЦНМО, 2003. — 328 с.

37. Биккенин Р. Р. Стеганография — современный метод обеспечения безопасности информации // Информация и космос. - N 2, 2006. – С. 89-93.

38. Девянин П. Н. Модели безопасности компьютерных систем: учеб. пособие для студентов высш. учеб. заведений. – М.: Изд. центр «Академия», 2005. – 144 с.

39. Левин М. Библия хакера 2. Кн. 1. – М.: Майор, 2003. – 640 с.

40. Левин М. Библия хакера 2. Кн. 2. – М.: Майор, 2003. – 688 с.

41. Владимиров А. А. Wi-фу: боевые приемы взлома и защиты беспроводных сетей. / А. А. Владимиров, К. В. Гавриленко, А. А. Михайловский: пер. с англ. А. А. Силкина. – М.: НТ Пресс, 2005. – 463 с.

42. Белоусов С. А., Гуц А. К., Планков М. С. Троянские кони. Принципы работы и методы защиты: учебное пособие. – Омск: изд. Наследие. Диалог-Сибирь, 2003. – 84 с.

43. Аналитический бюллетень Secure List – URL: http://www.securelist.com/ru/

44. Электронный учебник по разработке информационной безопасности компьютеров // Help Antivirus – URL: http://help-antivirus.ru/developmentsafety/Menu.php

45. Хогланд Г., Мак-Гроу Г. Взлом программного обеспечения: анализ и использование кода.: Пер. с англ. – М.: Издательский дом «Вильямс», 2005. – 400 с.

46. Макаренко С. И. Анализ математического аппарата расчета качества обслуживания информационно-вычислительной сети на сетевом уровне эталонной модели взаимодействия открытых систем // VII Всероссийская конф. молодых ученых по математическому моделированию и информационным технологиям / ИВТ СО РАН, 2006. [Эл. ресурс] - URL: http://www.ict.nsc.ru/ws/YM2006/10566/article.htm

47. Макаренко С. И. Расчетные соотношения для определения числовых характеристик вероятностной оценки времени задержки трафика в проводном, спутниковом и радио каналах связи // Сборник докладов Всероссийской научно-технической школы-семинара «Передача, обработка и отображение информации при быстропротекающих процессах» / РАРАН, октябрь 2006, г. Сочи. - М.: РПА «АПР», 2006. – С. 147-149.

48. Макаренко С. И., Кихтенко А. В. Вывод расчетных соотношений для времени обслуживания и эффективной пропускной способности спутникового и радио каналов связи.: Ставропольское высшее военное инженерное училище (военный институт). – Ставрополь: 2006. – 24 с. - Библиогр.: с. 19. - Деп. в СИФ ЦВНИ Минобороны РФ 14.05.2007, № 15246. - СИФ ЦВНИ Минобороны РФ, инв. № В6554.

49. Макаренко С. И. Методика оценки времени задержки пакета в канале связи в условиях нестабильности входного трафика // Инфокоммуникационные технологии. 2007. Т. 5. № 3. С. 95-96.

50. Макаренко С. И., Кихтенко А. В. Методика оценки времени задержки пакета в спутниковой сети связи в условиях нестабильности входного трафика // Системы управления и информационные технологии. 2007. № 1.3 (27). С. 344-348.

51. Макаренко С. И., Сидорчук В. П., Краснокутский А. В. Методика оценки времени задержки пакета в сети воздушной радио связи в условиях нестабильности входного трафика // Физика волновых процессов и радиотехнические системы. 2007. Т. 10. № 6. С. 70-74.

52. Макаренко С. И., Кихтенко А. В. Показатели качества обслуживания информационно-вычислительной сети АСУ реального времени в условиях нестационарности потоков данных // Авиакосмические технологии и оборудование. Казань-2006: Мат. Международной научно-практической конференции. 15-16 августа 2006 года. - Казань: изд. КГТУ им. А. Н. Туполева, 2006. – С. 173–174.

53. Макарено С. И., Кихтенко А. В. Анализ методов оценки влияния нестабильности входных потоков данных на показатели качества обслуживания информационно вычислительной сети АСУ реального времени // Передача, обработка и отображение информации: Сб. по мат. докл. Всероссийской научно-технической школы-семинара «Проблемы совершенствования боевых авиационных комплексов, повышение эффективности их ремонта и эксплуатации» / г. Терскол, 2006. - Ставрополь: изд. СВВАИУ (ВИ). 2006. – С. 97-99.

54. Макаренко С. И. Показатели качества обслуживания информационно-вычислительной сети АСУ реального времени в условиях нестационарности потоков данных : Ставропольское высшее военное инженерное училище (военный институт). – Ставрополь: 2006. – 23 с. - Библиогр.: с. 19-20. - Деп. в СИФ ЦВНИ Минобороны РФ 14.05.2007, № 15247. - СИФ ЦВНИ Минобороны РФ, инв. № В6555.

55. Макаренко С. И. Задача адаптивного управления пропускной способностью каналов сети воздушной радиосвязи в условиях квазистационарности потоков данных. // Сб. докл. .Всероссийской научно-технической школы-семинара «Проблемы совершенствования боевых авиационных комплексов, повышение эффективности их эксплуатации и ремонта» - Ставрополь: изд. СВВАИУ, 2007. – с. 25-28.

56. Макаренко С. И. Адаптивное управление информационными и сетевыми ресурсами // Научное, экспертно-аналитическое и информационное обеспечение стратегического управления, разработки и реализации приоритетных национальных проектов и программ. Сб. науч. тр. ИНИОН РАН. Ред. кол.: Пивоваров Ю.С. (отв. ред.) и др. – М., 2007. – с. 534-538.

57. Макаренко С. И. Адаптивное управление скоростями логических соединений в канале радиосвязи множественного доступа // Информационно-управляющие системы. 2008. № 6. С. 54-58.

58. Макаренко С. И. Вычислительные системы, сети и телекоммуникации: учебное пособие. – Ставрополь: СФ МГГУ им. М. А. Шолохова, 2008. – 353 с.

59. Макаренко С. И. Операционные системы, среды и оболочки: учебное пособие. – Ставрополь: СФ МГГУ им. М. А. Шолохова, 2008. – 210 с.