А.А. Безбогов, А.В. Яковлев, В.Н. Шамкин

МЕТОДЫ И СРЕДСТВА ЗАЩИТЫ КОМПЬЮТЕРНОЙ ИНФОРМАЦИИ

ИЗДАТЕЛЬСТВО ТГТУ

Министерство образования и науки Российской Федерации

ГОУ ВПО "Тамбовский государственный технический университет"

А.А. Безбогов, А.В. Яковлев, В.Н. Шамкин

МЕТОДЫ И СРЕДСТВА ЗАЩИТЫ КОМПЬЮТЕРНОЙ ИНФОРМАЦИИ

Утверждено Ученым советом университета в качестве учебного пособия

Тамбов
Издательство ТГТУ
2006

УДК 681.322.067
ББК з973.26-018.2я73
Б391

Рецензенты:

Заведующий кафедрой КРЭМС Заслуженный деятель науки и техники РФдоктор технических наук, профессор
Ю.Л. Муромцев

Директор РУНЦ по ИБ
Заслуженный работник высшей школы, профессор
Ю.Ф. Мартемьянов

Безбогов, А.А.
Б391 Методы и средства защиты компьютерной информации : учебное пособие / А.А. Безбогов, А.В. Яковлев, В.Н. Шамкин. – Тамбов : Изд-во Тамб. гос. техн. ун-та, 2006. – 196 с. – 100 экз. – ISBN 5-8265-0504-4.

Рассматриваются проблемы уязвимости информации в современных системах обработки данных, анализируются и классифицируются угрозы безопасности информации, конкретизируются задачи систем ее обеспечения, дается обзор методов, технических приемов и аппаратуры защиты информации. Основное внимание уделяется проблемам опознавания пользова-

теля, криптографическим методам защиты информации, методам защиты от компьютерных вирусов, защите от утечки информации по техническим каналам, организационно-правовому обеспечению безопасности информации. Излагаются некоторые методы и этапы построения комплексной системы защиты информации, а также перспективы создания изначально защищенных информационных технологий.

Пособие может быть полезно при курсовом и дипломном проектировании, аспирантам, а также кругу читателей, интересующихся современными проблемами защиты информации.

УДК 681.322.067
ББК з973.26-018.2я73

ISBN 5-8265-0504-4

Учебное издание

БЕЗБОГОВ Александр Александрович ЯКОВЛЕВ Алексей Вячеславович ШАМКИН Валерий Николаевич

МЕТОДЫ И СРЕДСТВА ЗАЩИТЫ КОМПЬЮТЕРНОЙ ИНФОРМАЦИИ

Учебное пособие

Редактор З.Г. Чернова

Компьютерное макетирование Е.В. Кораблевой

Подписано в печать 5.10.2006
Формат 60 × 84/16. Бумага офсетная. Гарнитура Times New Roman.
11,5 уч.-изд. л. Тираж 100 экз. Заказ № 412

Издательско-полиграфический центр ТГТУ
392000, Тамбов, Советская, 106, к. 14

СОДЕРЖАНИЕ

Введение

Информационная безопасность – сравнительно молодая, быстро развивающаяся область информационных технологий. Словосочетание информационная безопасность в разных контекстах может иметь различный смысл. Состояние защищенности национальных интересов в информационной сфере определяется совокупностью сбалансированных интересов личности, общества и государства.

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.

В данном учебном пособии основное внимание будет сосредоточено на хранении, обработке и передаче информации вне зависимости от того, на каком языке она закодирована, кто или что является ее источником и какое психологическое воздействие она оказывает на людей.

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

С методологической точки зрения правильный 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.

Информационная безопасность – многогранная область деятельности, в которой успех может принести только систематический, комплексный подход. Для решения данной проблемы рассматриваются меры законодательного, административного, процедурного и программно-технического уровня.

Информационная безопасность не сводится исключительно к защите от несанкционированного доступа к информации, это принципиально более широкое понятие. Термин «компьютерная безопасность» (как эквивалент или заменитель ИБ) слишком узок. Компьютеры – только одна из составляющих информационных систем, и хотя внимание будет сосредоточено, в первую очередь, 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.

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 выражение, а целью защиты информации становится уменьшение размеров ущерба до допустимых значений.

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

Информационная безопасность является одним из важнейших аспектов интегральной безопасности, на каком бы уровне он ни рассматривался – национальном, отраслевом, корпоративном или персональном.

Для иллюстрации этого ограничимся несколькими примерами.

По распоряжению президента США Клинтона (от 15 июля 1996 г.) 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 г. 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.

В феврале 2001 г. двое бывших сотрудников компании Commerce One, воспользовавшись паролем администратора, удалили с сервера файлы, составлявшие крупный проект для иностранного заказчика. В августе 2002 г. преступники предстали перед судом.

При анализе проблематики, связанной с информационной безопасностью, необходимо учитывать специфику данной области, развивающейся беспрецедентно высокими темпами. Здесь важны не столько отдельные решения (законы, учебные курсы, про-

граммно-технические изделия), находящиеся на современном уровне, сколько механизмы генерации новых решений, позволяющие жить в темпе технического прогресса.

Приведем еще несколько цифр. В марте 2001 г. был опубликован очередной, шестой по счету, годовой отчет «Компьютерная преступность и безопасность-2001: проблемы и тенденции». В отчете отмечается резкий рост числа обращений в правоохранительные органы по поводу компьютерных преступлений (32 % из числа опрошенных); 30 % респондентов сообщили, что их информационные системы были взломаны внешними злоумышленниками; атакам через Internet подвергались 57 % опрошенных; в 55 % случаях отмечались нарушения со стороны собственных сотрудников. Примечательно, что 33 % респондентов на вопрос «были ли взломаны ваши Web-серверы и системы электронной коммерции за последние 12 месяцев?» ответили «не знаю».

В аналогичном отчете, опубликованном в апреле 2002 г., цифры изменились, но тенденция осталась прежней: 90 % респондентов (преимущественно из крупных компаний и правительственных структур) сообщили, что за последние 12 месяцев в их организациях имели место нарушения информационной безопасности; 80 % констатировали финансовые потери от этих нарушений; 44 % (223 респондента) смогли и/или захотели оценить потери количественно, общая сумма составила более 455 млн. долларов. Наибольший ущерб нанесли кражи и подлоги (более 170 и 115 млн. долларов соответственно).

Увеличение числа атак – еще не самая большая неприятность. Хуже то, что постоянно обнаруживаются новые уязвимые места в программном обеспечении и, как следствие, появляются новые виды атак.

Так, в информационном письме Национального центра защиты инфраструктуры США (National Infrastructure Protection Center, NIPC) от 21 июля 1999 г. сообщается, что за период с 3 по 16 июля 1999 г. выявлено девять проблем с ПО, риск использования которых оценивается как средний или высокий. Среди «пострадавших» операционных платформ – почти все разновидности ОС UNIX, Windows, MacOS. Это связано с тем, что обнаруженные новые ошибки тут же начинают активно использовать злоумышленники.

В таких условиях системы информационной безопасности должны уметь противостоять разнообразным атакам как внешним, так и внутренним, атакам автоматизированным и скоординированным, мгновенным и вялотекущим. Целью злоумышленников может быть нарушение всех составляющих ИБ – доступности, целостности или конфиденциальности.

Предполагается, что данное учебное пособие поможет заложить необходимый теоретический базис по проблемам защиты ИС будущим специалистам в области информационных технологий.

I. Основные понятия и положения защиты информации в информационно-вычислительных системах

1. ПРЕДМЕТ И ОБЪЕКТ ЗАЩИТЫ

1.1. ПРЕДМЕТ ЗАЩИТЫ ИНФОРМАЦИИ

Вопросы информационной безопасности занимают особое место и в связи с возрастающей ролью в жизни общества требуют к себе все большего внимания. Успех практически любой деятельности в немалой степени зависит от умения распоряжаться такой ценностью, как информация.

В законе РФ «Об информации, информатизации и защите информации» определено:

• «*информационные ресурсы* являются объектами собственности граждан, организаций, общественных объединений, государства»;

• «*информация* – сведения о лицах, предметах, событиях, явлениях и процессах (независимо от формы их представления), отраженные на материальных носителях, используемые в целях получения знаний и практических решений».

Информация имеет ряд особенностей:

– не материальна;

– хранится и передается с помощью материальных носителей;

– любой материальный объект содержит информацию о самом себе либо о другом объекте.

Информации присущи следующие свойства:

Ценность информации определяется степенью ее полезности для владельца. Законом РФ «Об информации, информатизации и защите информации» гарантируется право собственника информации на ее использование и защиту от доступа к ней других лиц (организаций). Если доступ к информации ограничен, то такая информация называется *конфиденциальной*. Конфиденциальная информация может содержать государственную или коммерческую тайну.

Конфиденциальность информации – субъективно определяемая (приписываемая) характеристика (свойство) информации, указывающая на необходимость введения ограничений на круг субъектов, имеющих доступ к данной информации, и обеспечиваемая способностью системы (среды) сохранять указанную информацию в тайне от субъектов, не имеющих полномочий доступа к ней. Объективные предпосылки подобного ограничения доступности информации для одних субъектов заключены в необходимости защиты их законных интересов от других субъектов информационных отношений.

Государственную тайну могут содержать сведения, принадлежащие государству (государственному учреждению). В соответствии с законом «О государственной тайне» сведениям, представляющим ценность для государства, может быть присвоена одна из трех возможных степеней (гриф) секретности: «*особая важность*», «*совершенно секретно*» и «*секретно*». Для менее важной информации в государственных учреждениях существует гриф «*для служебного пользования*».

Коммерческую тайну могут содержать сведения, принадлежащие частному лицу, фирме, корпорации и т.д. Для обозначения ценности конфиденциальной коммерческой информации используют три категории: «*коммерческая тайна – строго конфиденциально* (*строгий учет*)», «*коммерческая тайна – конфиденциально*», «*коммерческая тайна*».

Достоверность информации определяется достаточной для владельца точностью отражать объекты и процессы окружающего мира в определенных временных и пространственных рамках. Информация, искаженно представляющая действительность, может нанести владельцу значительный материальный и моральный ущерб. Если информация искажена умышлено, тот ее называют *дезинформацией.*

Своевременность информации, т.е. соответствие ценности и достоверности определенному временному периоду.

Данное свойство определяется выражением

$$C(t) = C_0 e^{-2,3t/\tau},$$

где C_0 – ценность информации в момент ее возникновения; t – время от момента возникновения информации до момента определения ее стоимости; τ – время от момента возникновения информации до момента ее устаревания.

Предметом защиты является информация, хранящаяся, обрабатываемая и передаваемая в компьютерных (информационных) системах. Особенностями данного вида информации являются:

– двоичное представление информации внутри системы, независимо от физической сущности носителей исходной информации;
– высокая степень автоматизации обработки и передачи информации;
– концентрации большого количества информации в КС.

1.2. ОБЪЕКТ ЗАЩИТЫ ИНФОРМАЦИИ

Объектом защиты информации является компьютерная (информационная) система или автоматизированная система обработки информации (АСОИ).

Информационная система – это организационно-упорядоченная совокупность информационных ресурсов, технических средств, технологий и персонала, реализующих информационные процессы в традиционном или автоматизированном режиме для удовлетворения информационных потребностей пользователей.

Информационная безопасность АСОИ – состояние рассматриваемой автоматизированной системы, при котором она, с одной стороны, способна противостоять дестабилизирующему воздействию внешних и внутренних информационных угроз, а с другой, – ее наличие и функционирование не создает информационных угроз для элементов самой системы и внешней среды.

Информационная безопасность достигается проведением соответствующего уровня политики информационной безопасности.

Под *политикой информационной безопасности* понимают совокупность норм, правил и практических рекомендаций, регламентирующих работу средств защиты АСОИ от заданного множества угроз безопасности.

Система защиты информации – совокупность правовых норм, организационных мер и мероприятий, технических, программных и криптографических средств и методов, обеспечивающих защищенность информации в системе в соответствии с принятой политикой безопасности.

Контрольные вопросы

1. Охарактеризуйте информацию и ее свойства.
2. Что является предметом и объектом защиты информации?
3. Чем определяется ценность информации? Приведите классификацию конфиденциальной информаций.
4. Охарактеризуйте свойства достоверности и своевременности информации.
5. Дайте определения информационной безопасности АСОИ и политики информационной безопасности

2. Угрозы безопасности информации в информационно-вычислительных системах

2.1. ПОНЯТИЕ УГРОЗЫ БЕЗОПАСНОСТИ

С позиции обеспечения безопасности информации в ИВС целесообразно рассматривать в виде трех связных взаимовлияющих друг на друга компонент:

1) информация;
2) технические и программные средства;
3) обслуживающий персонал и пользователи.

Целью создания любой ИВС является удовлетворение потребностей пользователей в своевременном получении достоверной информации и сохранении ее конфиденциальности. При этом задача обеспечения информации должна решаться путем защиты от внешних и внутренних неразрешенных (несанкционированных) воздействий.

Под *угрозой* обычно понимают потенциально возможно событие, действие (воздействие), процесс или явление, которое может привести к нанесению ущерба чьим-либо интересам. В дальнейшем изложении угрозой информационной безопасности АС будем называть возможность реализации воздействия на информацию, обрабатываемую АС, приводящего к искажению, уничтожению, копированию, блокированию, доступа к информации, а также возможность воздействия на компоненты АС, приводящего к утрате, уничтожению или сбою функционирования носителя информации, средства взаимодействия с носителем или средства его управления.

Утечка информации рассматривается как бесконтрольный и неправомерный выход конфиденциальной информации за пределы организации или круга лиц, которым эта информация была доверена.

Существует три разновидности угроз.

1. *Угроза нарушения конфиденциальности* заключается в том, что информация становится известной тому, кто не располагает полномочиями доступа к ней. Она имеет место всякий раз, когда получен доступ к некоторой секретной информации, хранящейся в вычис-

лительной системе или передаваемой от одной системы к другой. Иногда, в связи с угрозой нарушения конфиденциальности, используется термин «утечка».

2. *Угроза нарушения целостности* включает в себя любое умышленное изменение информации, хранящейся в вычислительной системе или передаваемой из одной системы в другую. Когда злоумышленники преднамеренно изменяют информацию, говорится, что целостность информации нарушена. Целостность также будет нарушена, если к несанкционированному изменению приводит случайная ошибка программного или аппаратного обеспечения. Санкционированными изменениями являются те, которые сделаны уполномоченными лицами с обоснованной целью (например, санкционированным изменением является периодическая запланированная коррекция некоторой базы данных).

Целостность информации – существование информации в неискаженном виде (неизменном по отношению к некоторому фиксированному ее состоянию). Чаще субъектов интересует обеспечение более широкого свойства – достоверности информации, которое складывается из адекватности (полноты и точности) отображения состояния предметной области и непосредственно целостности информации, т.е. ее неискаженности.

3. *Угроза отказа служб* возникает всякий раз, когда в результате преднамеренных действий, предпринимаемых другим пользователем или злоумышленником, блокируется доступ к некоторому ресурсу вычислительной системы. Реально блокирование может быть постоянным – запрашиваемый ресурс никогда не будет получен, или оно может вызывать только задержку запрашиваемого ресурса, достаточно долгую для того чтобы он стал бесполезным. В этих случаях говорят, что ресурс исчерпан.

Доступность информации – свойство системы (среды, средств и технологии обработки), в которой циркулирует информация, характеризующееся способностью обеспечивать своевременный беспрепятственный доступ субъектов к интересующей их информации и готовность соответствующих автоматизированных служб к обслуживанию поступающих от субъектов запросов всегда, когда в обращении к ним возникает необходимость.

2.2. КЛАССИФИКАЦИЯ УГРОЗ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

Классификация всех возможных угроз информационной безопасности АС может быть проведена по ряду базовых признаков.

1. По природе возникновения.

Естественные угрозы – угрозы, вызванные воздействиями на АС и ее компоненты объективных физических процессов или стихийных природных явлений, независящих от человека.

Искусственные угрозы – угрозы информационной безопасности АС, вызванные деятельностью человека.

2. По степени преднамеренности проявления.

Угрозы случайного действия и/или угрозы, вызванные ошибками или халатностью персонала. Угрозы, не связанные с преднамеренными действиями злоумышленников и реализуемые в случайные моменты времени, называют случайными или непреднамеренными. Классификация по этому признаку приведена на рис. 2.1.

Реализация угроз этого класса приводит к наибольшим потерям информации (до 80 % ущерба). При этом может происходить уничтожение, нарушение целостности, доступности и конфиденциальности информации, например:

- проявление ошибок программно-аппаратных средств АС;
- некомпетентное использование, настройка или неправомерное отключение средств защиты персоналом службы безопасности;
- неумышленные действия, приводящие к частичному или полному отказу системы или разрушению аппаратных, программных, информационных ресурсов системы (неумышленная порча оборудования, удаление, искажение файлов с важной информацией или программ, в том числе системных и т.п.);
- неправомерное включение оборудования или изменение режимов работы устройств и программ;
- неумышленная порча носителей информации;
- пересылка данных по ошибочному адресу абонента (устройства);
- ввод ошибочных данных;
- неумышленное повреждение каналов связи.

Угрозы преднамеренного действия, например:

- традиционный или универсальный шпионаж и диверсии (подслушивание, визуальное наблюдение; хищение документов и машинных носителей, хищение программ и атрибутов системы защиты, подкуп и шантаж сотрудников, сбор и анализ отходов машинных носителей, поджоги, взрывы);

– несанкционированный доступ к информации (реализуется посредством отсутствия системы разграничения доступа (СРД), сбоями или отказами технических средств), ошибками в СРД, фальсификацией полномочий);

– побочные электромагнитные излучения и наводки (ПЭМИН);

– несанкционированная модификация структур (алгоритмической, программной, технической);

– информационные инфекции (вредительские программы).

3. По непосредственному источнику угроз.

Угрозы, непосредственным *источником которых является природная среда* (стихийные бедствия, магнитные бури, радиоактивное излучение и т.п.).

Угрозы, источником которых является человек, например:

– внедрение агентов в число персонала системы (в том числе, возможно, и в административную группу, отвечающую за безопасность);

– вербовка (путем подкупа, шантажа и т.п.) персонала или отдельных пользователей, имеющих определенные полномочия;

– угроза несанкционированного копирования секретных данных пользователем АС;

– разглашение, передача или утрата атрибутов разграничения доступа (паролей, ключей шифрования, идентификационных карточек, пропусков и т.п.).

Угрозы, непосредственным источником которых являются санкционированные программно-аппаратные средства, например:

– запуск технологических программ, способных при некомпетентном пользовании вызывать потерю работоспособности системы (зависания или зацикливания) или необратимые изменения в системе (форматирование или реструктуризацию носителей информации, удаление данных и т.п.);

– возникновение отказа в работе операционной системы.

Угрозы, непосредственным источником которых являются несанкционированные программно-аппаратные средства, например:

– нелегальное внедрение и использование неучтенных программ (игровых, обучающих, технологических и других, не являющихся необходимыми для выполнения нарушителем своих служебных обязанностей) с последующим необоснованным расходованием ресурсов (загрузка процессора, захват оперативной памяти и памяти на внешних носителях);

– заражение компьютера вирусами с деструктивными функциями.

4. По положению источника угроз.

Угрозы, источник которых расположен вне контролируемой зоны территории (помещения), на которой находится АС, например:

– перехват побочных электромагнитных, акустических и других излучений устройств и линий связи, а также наводок активных излучений на вспомогательные технические средства, непосредственно не участвующие в обработке информации (телефонные линии, сети питания, отопления и т.п.);

– перехват данных, передаваемых по каналам связи, и их анализ с целью выяснения протоколов обмена, правил вхождения в связь и авторизации пользователя и последующих попыток их имитации для проникновения в систему;

– дистанционная фото- и видеосъемка.

Угрозы, источник которых расположен в пределах контролируемой зоны территории (помещения), на которой находится АС, например:

– хищение производственных отходов (распечаток, записей, списанных носителей информации и т.п.);

– отключение или вывод из строя подсистем обеспечения функционирования вычислительных систем (электропитания, охлаждения и вентиляции, линий связи и т.д.);

– применение подслушивающих устройств.

Угрозы, источник которых имеет доступ к периферийным устройства АС (терминалам).

Угрозы, источник которых расположен в АС, например:

– проектирование архитектуры системы и технологии обработки данных, разработка прикладных программ, которые представляют опасность для работоспособности системы и безопасности информации;

– некорректное использование ресурсов АС.

5. По степени зависимости от активности АС.

Угрозы, которые могут проявляться независимо от активности АС, например:

– вскрытие шифров криптозащиты информации;

– хищение носителей информации (магнитных дисков, лент, микросхем памяти, запоминающих устройств и компьютерных систем).

Угрозы, которые могут проявляться только в процессе автоматизированной обработки данных (например, угрозы выполнения и распространения программных вирусов).

6. По степени воздействия на АС.

Пассивные угрозы, которые при реализации ничего не меняют в структуре и содержании АС, например: угроза копирования секретных данных.

Активные угрозы, которые при воздействии вносят изменения в структуру и содержание АС, например:

– внедрение аппаратных спецвложений, программных «закладок» и «вирусов» («троянских коней» и «жучков»), т.е. таких участков программ, которые не нужны для выполнения заявленных функций, но позволяют преодолеть систему защиты, скрытно и незаконно осуществить доступ к системным ресурсам с целью регистрации и передачи критической информации или дезорганизации функционирования системы;

– действия по дезорганизации функционирования системы (изменение режимов работы устройств или программ, забастовка, саботаж персонала, постановка мощных активных радиопомех на частотах работы устройств системы и т.п.);

– угроза умышленной модификации информации.

7. По этапам доступа пользователей или программ к ресурсам АС.

Угрозы, которые могут проявляться на этапе доступа к ресурсам АС (например, угрозы несанкционированного доступа в АС).

Угрозы, которые могут проявляться после разрешения доступа к ресурсам АС (например, угрозы несанкционированного или некорректного использования ресурсов АС).

8. По способу доступа к ресурсам АС.

Угрозы, направленные на использование прямого стандартного пути доступа к ресурсам АС. Например:

– незаконное получение паролей и других реквизитов разграничения доступа (агентурным путем, используя халатность пользователей, подбором, имитацией интерфейса системы и т.д.) с последующей маскировкой под зарегистрированного пользователя («маскарад»);

– несанкционированное использование терминалов пользователей, имеющих уникальные физические характеристики, такие как номер рабочей станции в сети, физический адрес, адрес в системе связи, аппаратный блок кодирования и т.п.

Угрозы, направленные на использование скрытого нестандартного пути доступа к ресурсам АС, например:

– вход в систему в обход средств защиты (загрузка посторонней операционной системы со сменных магнитных носителей и т.п.);

– угроза несанкционированного доступа к ресурсам АС путем использования недокументированных возможностей ОС.

9. По текущему месту расположения информации, хранимой и обрабатываемой в АС.

Угрозы доступа к информации на внешних запоминающих устройства (например, угроза несанкционированного копирования секретной информации с жесткого диска).

Угрозы доступа к информации в оперативной памяти, например:

– чтение остаточной информации из оперативной памяти;

– чтение информации из областей оперативной памяти, используемых операционной системой (в том числе подсистемой защиты) или другими пользователями, в асинхронном режиме, используя недостатки мультизадачных АС и систем программирования;

– угроза доступа к системной области оперативной памяти со сторон прикладных программ.

Угрозы доступа к информации, циркулирующей в линиях связи, например:

– незаконное подключение к линиям связи с целью работы «между строк» с использованием пауз в действиях законного пользователя от его имени с последующим вводом ложных сообщений или модификацией передаваемых сообщений;

– незаконное подключение к линиям связи с целью прямой подмены законного пользователя путем его физического отключения после входа в систему и успешной аутентификации с последующим вводом дезинформации и навязыванием ложных сообщений;

– перехват всего потока данных с целью дальнейшего анализа не в реальном масштабе времени.

Угрозы доступа к информации, отображаемой на терминале или печатаемой на принтере, например, угроза записи отображаемой информации на скрытую видеокамеру.

2.3. КЛАССИФИКАЦИЯ ЗЛОУМЫШЛЕННИКОВ

Возможности осуществления вредительских воздействий в большой степени зависят от статуса злоумышленника по отношению к ИВС. Злоумышленником может быть:

– *разработчик* ИВС (владеет наиболее полной информацией о программных и аппаратных средствах ИВС и имеет возможность внедрения «закладок» на этапах создания и модернизации систем, но не получает доступа на эксплуатируемые объекты ИВС);

– *сотрудник из числа обслуживающего персонала* (наиболее опасный класс – работники службы безопасности информации, далее идут системные и прикладные программисты, инженерно-технический персонал);

– *пользователь* (имеет общее представление о структуре ИВС и механизмах ее защиты, но может осуществлять сбор информации методами традиционного шпионажа и попытками НСДИ);

– *постороннее лицо* (может осуществлять дистанционные методы шпионажа и диверсионную деятельность).

2.4. ОСНОВНЫЕ МЕТОДЫ РЕАЛИЗАЦИИ УГРОЗ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

К основным направлениям реализации злоумышленником информационных угроз относятся:

– непосредственное обращение к объектам доступа;

– создание программных и технических средств, выполняющих обращение к объектам доступа в обход средств защиты;

– модификация средств защиты, позволяющая реализовать угрозы информационной безопасности;

– внедрение в технические средства АС программных или технических механизмов, нарушающих предполагаемую структуру и функции АС.

К числу основных методов реализации угроз информационной безопасности АС относятся:

– определение злоумышленником типа и параметров носителей информации;

– получение злоумышленником информации о программно-аппаратной среде, типе и параметрах средств вычислительной техники, типе и версии операционной системы, составе прикладного программного обеспечения;

– получение злоумышленником детальной информации о функциях, выполняемых АС;

– получение злоумышленником данных о системах защиты;

– определение способа представления информации;

– определение злоумышленником содержания данных, обрабатываемых в АС, на качественном уровне (мониторинг дешифрования сообщений);

– хищение (копирование) машинных носителей информации, имеющих конфиденциальные данные;

– хищение (копирование) носителей информации;

– использование специальных технических средств для перехвата побочных электромагнитных излучений и наводок (ПЭМИН) – конфиденциальные данные перехватываются злоумышленником путем изменения информативных сигналов из электромагнитного излучения и наводок по цепям питания средств вычислительной техники, входящей в АС;

– уничтожение средств ВТ и носителей информации;

– несанкционированный доступ пользователя к ресурсам АС путем преодоления систем защиты с использованием спецсредств, приемов, методов;

– несанкционированное превышение пользователем своих полномочий;

– несанкционированное копирование программного обеспечения;

– перехват данных, передаваемых по каналам связи;

– визуальное наблюдение – конфиденциальные данные считываются с экранов терминалов, распечаток в процессе их печати и т.п.;

– раскрытие представления информации (дешифрование данных);

– раскрытие содержания информации на семантическом уровне к смысловой составляющей информации, хранящейся в АС;

– уничтожение машинных носителей информации;

– внесение пользователем несанкционированных изменений программно-аппаратные компоненты АС и обрабатываемых данных;

– установка и использование нештатного аппаратного и/или программного обеспечения;

– заражение программными вирусами;

– внесение искажений в представление данных, уничтожение на уровне представления, искажение информации при передаче по линиям связи;

– внедрение дезинформации;

– выведение из строя машинных носителей информации без уничтожения информации (выведение из строя электронных блоков жестких дисков и т.п.);

– проявление ошибок проектирования и разработки аппаратных программных компонентов АС;

– обход (отключение) механизмов защиты – загрузка злоумышленником нештатной операционной системы с дискеты, использование режимов программно-аппаратных компонент АС и т.п.

– искажение соответствия синтаксических и семантических конструкций языка – установление новых значений слов, выражений и т.п.;

– запрет на использование информации – имеющаяся информация каким-либо причинам не может быть использована.

Основные методы реализации угроз информационной безопасности приведены в табл. 2.1.

2.1. Основные методы реализации угроз ИБ

Уровень доступа к информации в АС	Основные методы реализации угроз информационной безопасности			
	Угроза раскрытия параметров системы	Угроза нарушения конфиденциальности	Угроза нарушения целостности	Угроза отказа служб (отказа доступа к информации)
Носителей информации	Определение типа и параметров носителей информации	Хищение (копирование) носителей информации. Перехват ПЭМИН	Уничтожение машинных носителей информации	Выведение из строя машинных носителей информации
Средств взаимодействия с носителем	Получение информации о программно-аппаратной среде. Получение детальной информации о функциях, выполняемых АС. Получение данных о применяемых системах защиты	Несанкционированный доступ к ресурсам АС. Совершение пользователем несанкционированных действий. Несанкционированное копирование программного обеспечения. Перехват данных, передаваемых по каналам связи	Внесение пользователем несанкционированных изменений в программы и данные. Установка и использование нештатного программного обеспечения. Заражение программными вирусами	Проявление ошибок проектирования и разработки программно-аппаратных компонент АС. Обход механизмов защиты АС
Представления информации	Определение способа представления информации	Визуальное наблюдение. Раскрытие представления информации (дешифрование)	Внесение искажения в представление данных; уничтожение данных	Искажение соответствия синтаксических и семантических конструкций языка
Содержания информации	Определение содержания данных на качественном уровне	Раскрытие содержания информации	Внедрение дезинформации	Запрет на использование информации

2.5. ПРИЧИНЫ, ВИДЫ И КАНАЛЫ УТЕЧКИ ИНФОРМАЦИИ

Основными причинами утечки информации являются:

- несоблюдение персоналом норм, требований, правил эксплуатации АС;
- ошибки в проектировании АС и систем защиты АС;
- ведение противостоящей стороной технической и агентурной разведок.

Несоблюдение персоналом норм, требований, правил эксплуатации АС может быть как умышленным, так и непреднамеренным. От ведения противостоящей стороной агентурной разведки этот случай отличает то, что в данном случае лицом, совершающим несанкционированные действия, двигают личные побудительные мотивы. Причины утечки информации достаточно тесно связаны с видами утечки информации. В соответствии с ГОСТ Р 50922–96 рассматриваются три вида утечки информации:

- разглашение;
- несанкционированный доступ к информации;
- получение защищаемой информации разведками (как отечественными, так и иностранными).

Под *разглашением* информации понимается несанкционированное доведение защищаемой информации до потребителей, не имеющих права доступа к защищаемой информации.

Под *несанкционированным доступом* понимается получение защищаемой информации заинтересованным субъектом с нарушением установленных правовыми документами или собственником, владельцем информации прав или правил доступа к защищаемой информации. При этом заинтересованным субъектом, осуществляющим несанкционированный доступ к информации, может быть: государство, юридическое лицо, группа физических лиц, в том числе общественная организация, отдельное физическое лицо.

Получение защищаемой информации разведками может осуществиться с помощью технических средств (техническая разведка) или агентурными методами (агентурная разведка).

Канал утечки информации – совокупность источника информации, материального носителя или среды распространения несущего указанную информацию сигнала и средства выделения информации из сигнала или носителя. Одним из основных свойств канала является месторасположение средства выделения информации из сигнала или носителя, которое может располагаться в пределах контролируемой зоны, охватывающей или вне ее.

Применительно к АС выделяют несколько каналов утечки информации. Обобщенная схема каналов утечки приведена на рис. 2.2.

1. *Электромагнитный канал*. Причиной его возникновения является электромагнитное поле, связанное с протеканием электрического тока в аппаратных компонентах АС. Электромагнитное поле может индуцировать токи в близко расположенных проводных линиях (наводки). Электромагнитный канал, в свою очередь, делится на следующие каналы:

- радиоканал (высокочастотное излучение);
- низкочастотный;
- сетевой (наводки на сеть электропитания);
- заземления (наводки на провода заземления);
- линейный (наводки на линии связи между компьютерами).

2. *Акустический* (*виброакустический*) *канал* – связан с распространением звуковых волн в воздухе или упругих колебаний в других средах, возникающих при работе устройств отображения информации АС.

3. *Визуальный канал* – связан с возможностью визуального наблюдения злоумышленником за работой устройств отображения информации без проникновения в помещения, где расположены компоненты системы. В качестве средства выделения информации в данном случае могут рассматриваться фото-, видеокамеры и т.п.

4. *Информационный канал* – связан с доступом (непосредственным и телекоммуникационным) к элементам АС, к носителям информации, к самой вводимой и выводимой информации (и результатам), к программному обеспечению (в том числе к операционным системам), а также с подключением к линиям связи. Информационный канал может быть разделен на следующие каналы:

- коммутируемых линий связи,
- выделенных линий связи,
- локальной сети,
- машинных носителей информации,
- терминальных и периферийных устройств.

Контрольные вопросы

1. Что понимается под угрозой информации? Назовите разновидности угроз информации.
2. Приведите классификацию угроз информации.
3. Какие основные направления и методы реализации угроз Вам известны?
4. Поясните классификацию злоумышленников.
5. Охарактеризуйте причины и виды утечки информации.
6. Назовите и приведите примеры каналов утечки информации.

II. Методы и средства защиты информации в информационно-вычислительных системах

3. ПРАВОВЫЕ И ОРГАНИЗАЦИОННЫЕ МЕТОДЫ ЗАЩИТЫ ИНФОРМАЦИИ В ИНФОРМАЦИОННО-ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМАХ

Комплексная защита информации создается на объектах для блокирования (парирования) всех возможных или наиболее вероятных угроз безопасности информации. Для парирования той или иной угрозы используется определенная совокупность средств и методов защиты, некоторые из них защищают от нескольких угроз одновременно.

Среди методов защиты имеются и универсальные методы, являющиеся базовыми при построении любой системы защиты.

Правовые методы защиты информации служат основой легитимного построения и использования системы защиты любого назначения.

Организационные методы защиты информации используются для парирования нескольких угроз, кроме того, их использование в любой системе защиты обязательно.

3.1. ПРАВОВОЕ РЕГУЛИРОВАНИЕ В ОБЛАСТИ БЕЗОПАСНОСТИ ИНФОРМАЦИИ

Государство должно обеспечить в стране защиту информации как в масштабах всего государства, так и на уровне организаций и своих граждан. Для этого государство обязано:

- выработать государственную политику безопасности в области информационных технологий;

– законодательно определить правовой статус ИВС, информации, систем защиты информации, владельцев и пользователей информации и т.д.;

– создать иерархическую структуру государственных органов, вырабатывающих и проводящих в жизнь политику безопасности информационных технологий;

– создать систему стандартизации, лицензирования и сертификации в области защиты информации;

– обеспечить приоритетное развитие отечественных защищенных информационных технологий;

– повышать уровень образования граждан в области информационных технологий, воспитывать у них патриотизм и бдительность;

– установить ответственность граждан за нарушения законодательства в области информационных технологий.

3.1.1. ГОСУДАРСТВЕННАЯ ПОЛИТИКА РФ В ОБЛАСТИ БЕЗОПАСНОСТИ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

В государстве должна проводиться единая политика в области безопасности информационных технологий. В Российской Федерации вопросы информационной безопасности и защиты информации обеспечиваются соблюдением следующих указов Президента, федеральных законов, постановлений Правительства Российской Федерации, руководящих документов Гостехкомиссии России и других нормативных документов:

- Доктрина информационной безопасности Российской Федерации, утвержденная 9 сент. 2000 г.;
- Концепция национальной безопасности Российской Федерации, утвержденная 17 дек. 1997 г.;
- Указ Президента РФ от 6 марта 1997 г. № 188. Об утверждении перечня сведений конфиденциального характера;
- Закон Российской Федерации от 20 февр. 1995 г. № 24-ФЗ. Об информации, информатизации и защите информации;
- Закон Российской Федерации от 16 февр. 1995 г. № 15-ФЗ. О связи;
- Закон Российской Федерации от 23 сент. 1992 г. № 3523-1. О правовой охране программ для электронных вычислительных машин и баз данных;
- Закон Российской Федерации от 4 июля 1996 г. № 85-ФЗ. Об участии в международном информационном обмене;
- Постановление Правительства Российской Федерации от 16.09.98. О лицензировании отдельных видов деятельности;
- Закон Российской Федерации от 21 июля 1993 г. О государственной тайне;
- ГОСТ Р 51583. Защита информации. Порядок создания автоматизированных систем в защищенном исполнении.
- Руководящий документ. Положение по аттестации объектов информатизации по требованиям безопасности информации (Утверждено Председателем Гостехкомиссии России 25 нояб. 1994 г.);
- Руководящий документ. Автоматизированные системы. Защита от несанкционированного доступа к информации. Классификация автоматизированных систем и требования к защите информации (Гостехкомиссия России, 1997 г.);
- «Положение о сертификации средств защиты информации по требованиям безопасности информации» (Постановление Правительства РФ № 608, 1995 г.);
- Руководящий документ. Средства вычислительной техники. Защита от несанкционированного доступа к информации. Показатели защищенности от несанкционированного доступа к информации. (Гостехкомиссия России, 1998 г.);
- Руководящий документ. Концепция защиты средств вычислительной техники от несанкционированного доступа к информации. (Гостехкомиссия России, 1998 г.);
- Руководящий документ. Защита от несанкционированного доступа к информации. Термины и определения (Гостехкомиссия России, 1998 г.);
- Руководящий документ. Временное положение по организации разработки, изготовления и эксплуатации программных и технических средств защиты информации от НСД в АС и СВТ. (Гостехкомиссия России, 1998 г.);
- Руководящий документ. Средства вычислительной техники. Межсетевые экраны. Защита от несанкционированного доступа к информации. Показатели защищенности от несанкционированного доступа к информации. (Гостехкомиссия России, 1997 г.);
- Руководящий документ. Защита от несанкционированного доступа к информации. Ч. 1. Программное обеспечение средств защиты информации. Классификация по уровню контроля отсутствия недекларированных возможностей. (Гостехкомиссия России, 1999 г.);
- Руководящий документ. Специальные требования и рекомендации по технической защите конфиденциальной информации. (Гостехкомиссия России, 2001 г.).

«Доктрина …» – это документ, «представляющий собой совокупность официальных

взглядов на цели, задачи, принципы и основные направления обеспечения информационной безопасности» России, служит основой для «формирования государственной политики» в сфере информации.

Доктрина включает в себя перечень основных видов возможных угроз для информационной безопасности, которые в том числе связаны с телекоммуникационными системами. К числу таких угроз отнесены:

– закупка органами государственной власти импортных средств информатизации, телекоммуникации и связи при наличии отечественных аналогов, не уступающих по своим характеристикам зарубежным образцам;

– вытеснение с отечественного рынка российских производителей средств информатизации, телекоммуникации и связи;

– неспособность предприятий национальных отраслей электронной промышленности производить на базе новейших достижений микроэлектроники, передовых информационных технологий конкурентоспособную наукоемкую продукцию, позволяющую обеспечить достаточный уровень технологической независимости России от зарубежных стран, что приводит к вынужденному широкому использованию импортных программно-аппаратных средств при создании и развитии в России информационной инфраструктуры;

– увеличение оттока за рубеж специалистов и правообладателей интеллектуальной собственности;

– нарушение установленного регламента сбора, обработки и передачи информации, преднамеренные действия и ошибки персонала информационных и телекоммуникационных систем, отказ технических средств и сбои программного обеспечения в информационных и телекоммуникационных системах;

– использование не сертифицированных в соответствии с требованиями безопасности средств и систем информатизации и связи, а также средств защиты информации и контроля их эффективности;

– привлечение к работам по созданию, развитию и защите информационных и телекоммуникационных систем организаций и фирм, не имеющих государственных лицензий на осуществление этих видов деятельности.

В качестве некоторых общих методов обеспечения информационной безопасности Доктриной, в частности, предполагаются:

– обеспечение технологической независимости Российской Федерации в важнейших областях информатизации, телекоммуникации и связи, определяющих ее безопасность, и, в первую очередь, в области создания специализированной вычислительной техники для образцов вооружения и военной техники;

– уточнение статуса инвесторов при привлечении иностранных инвестиций для развития информационной инфраструктуры России;

– законодательное закрепление приоритета развития национальных сетей связи и отечественного производства космических спутников связи.

В числе первоочередных мер, направленных на обеспечение информационной безопасности, в Доктрине, в частности, названы:

– принятие и реализация федеральных программ, предусматривающих формирование общедоступных архивов информационных ресурсов федеральных органов государственной власти и органов государственной власти субъектов Российской Федерации;

– повышение правовой культуры и компьютерной грамотности граждан;

– развитие инфраструктуры единого информационного пространства России;

– создание безопасных информационных технологий для систем, используемых в процессе реализации жизненно важных функций общества и государства;

– пресечение компьютерной преступности;

– создание информационно-телекоммуникационной системы специального назначения в интересах федеральных органов государственной власти и органов государственной власти субъектов Российской Федерации;

– обеспечение технологической независимости страны в области создания и эксплуатации информационно-телекоммуникационных систем оборонного назначения.

В число основных организационно-технических мероприятий по защите информации в общегосударственных информационных и телекоммуникационных системах включены:

– лицензирование деятельности организаций в области защиты информации;

– аттестация объектов информатизации по выполнению требований обеспечения защиты информации при проведении работ, связанных с использованием сведений, составляющих государственную тайну;

– сертификация средств защиты информации и контроля эффективности их использования, а также защищенности информации от утечки по техническим каналам систем и средств информатизации и связи.

3.1.2. ЗАКОНОДАТЕЛЬНАЯ БАЗА В ОБЛАСТИ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

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. Различают на законодательном уровне две группы мер:

1) меры, направленные на создание и поддержание в обществе негативного (в том числе карательного) отношения к нарушениям и нарушителям информационной безопасности;

2) направляющие и координирующие меры, способствующие повышению образованности общества в области информационной безопасности, помогающие в разработке и распространении средств обеспечения информационной безопасности.

К первой группе следует отнести, в первую очередь, главу 28 «Преступления в сфере компьютерной информации» раздела IX новой редакции Уголовного кодекса. Эта глава достаточно полно охватывает основные угрозы информационным системам, однако обеспечение практической реализуемости соответствующих статей пока остается проблематичным.

Закон «Об информации, информатизации и защите информации» можно причислить к этой же группе. Правда, положения этого закона 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.

В Государственной Думе подготовлены законы «О праве на информацию», «О коммерческой тайне», «О персональных данных», которые охватывают все категории субъектов информационных отношений.

К группе направляющих и координирующих законов и нормативных актов относится целая группа документов, регламентирующих процессы лицензирования и сертификации в области информационной безопасности. Главная роль здесь отведена Государственной технической комиссии (Гостехкомиссии) при Президенте Российской Федерации.

В области информационной безопасности законы реально преломляются и работают через нормативные документы, подготовленные соответствующими ведомствами. В этой связи очень важны Руководящие документы Гостехкомиссии, определяющие требования к классам защищенности средств вычислительной техники и автоматизированных систем. Необходимо выделить утвержденный в июле 1997 г. Руководящий документ по межсетевым экранам, вводящий в официальную сферу один из самым современных классов защитных средств. Как уже указывалось, самое важное на законодательном уровне – создать механизм, позволяющий согласовать процесс разработки законов с реалиями и прогрессом информационных технологий. Конечно, 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. Пока только Гостехкомиссия России демонстрирует способность динамично развивать нормативную базу.

В современном мире глобальных сетей нормативно правовая база должна быть согласована с международной практикой. Хотелось бы обратить особое внимание на желательность приведения российских стандартов и сертификационных нормативов в соответствие с международным уровнем информационных технологий вообще и информационной безопасности в частности. Есть много причин, по которым это должно быть сделано. Одна из них – необходимость защищенного взаимодействия с зарубежными организациями и зарубежными филиалами российских организаций. Вторая – доминирование аппаратно–программных продуктов зарубежного производства.

На законодательном уровне должен получить реалистичное решение вопрос об отношении к таким изделиям. Здесь необходимо разделить два аспекта: независимость в области информационных технологий и информационную безопасность.

Использование зарубежных продуктов в некоторых критически важных системах (в первую очередь военных) в принципе может представлять угрозу национальной безопасности (в том числе информационной), поскольку нельзя исключить вероятности встраивания закладных элементов. В то же время, в подавляющем большинстве случаев потенциальные угрозы информационной безопасности носят исключительно внутренний характер. В таких условиях незаконность использования зарубежных разработок (ввиду сложностей с их сертификацией) при отсутствии отечественных аналогов затрудняет (или вообще делает невозможной) защиту информации без серьезных на то оснований. Проблема сертификации аппаратно программных продуктов зарубежного производства

действительно является сложной, однако, как показывает опыт европейских стран, она может быть успешно решена. 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 снижения национальной безопасности.

Современному российскому законодательству не хватает позитивной (не карательной) направленности. Информационная безопасность – это новая область деятельности, здесь важно научить, разъяснить, помочь, а не запретить и наказать. Общество должно осознать важность данной проблематики, понять основные пути решения соответствующих задач, должны быть скоординированы научные, учебные и производственные планы. Государство может сделать это оптимальным образом. Здесь не нужно больших материальных затрат, требуются интеллектуальные вложения.

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

- разработка новых законов с учетом интересов всех категорий субъектов информационных отношений;
- ориентация на созидательные, а не карательные законы;
- интеграция в мировое правовое пространство;
- учет современного состояния информационных технологий.

3.1.3. СТРУКТУРА ГОСУДАРСТВЕННЫХ ОРГАНОВ, ОБЕСПЕЧИВАЮЩИХ БЕЗОПАСНОСТЬ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

Выработку политики информационной безопасности, подготовку законодательных актов и нормативных документов, контроль над выполнением установленных норм обеспечения безопасности информации осуществляют государственные органы, структура которых приведена на рис. 3.1.

Возглавляет государственные органы обеспечения безопасности Президент РФ. Он руководит Советом безопасности и утверждает Указы, касающиеся обеспечения безопасности информации в государстве.

Общее руководство системой информационной безопасности, наряду с другими вопросами государственной безопасности страны, осуществляют Президент и Правительство Российской Федерации.

Органом исполнительной власти, непосредственно занимающимся вопросами государственной безопасности является Совет безопасности. В состав Совета безопасности входит Межведомственная комиссия по информационной безопасности, осуществляющая подготовку указов Президента и координирующая деятельность министерств и ведомств в области информационной безопасности.

Рабочим органом Межведомственной комиссии по информационной безопасности является Государственная техническая комиссия при Президенте РФ. Она осуществляет подготовку проектов законов, разрабатывает нормативные документы (Решения ГТК), организует сертификацию средств защиты информации (за исключением криптографических средств), лицензирование деятельности в области средств защиты и обучения специалистов по защите информации. ГТК руководит аттестацией КС, координирует и направляет деятельность государственных научно-исследовательских учреждений, работающих в области защиты информации, обеспечивает аккредитацию органов лицензирования и испытательных центров по сертификации. ГТК обеспечивает также работу Межведомственной комиссии по защите государственной тайны.

На Межведомственную комиссию по защите государственной тайны возложена задача руководства лицензированием предприятий, учреждений и организаций, связанных с использованием сведений, составляющих государственную тайну, с созданием средств защиты информации, а также оказанием услуг по защите гостайны. Кроме того, эта комиссия осуществляет координацию работы по организации сертификации средств защиты информации.

Подразделения правительственной связи и информации при Президенте Российской Федерации обеспечивает правительственную связь и информационные технологии государственного управления. Агентство осуществляет сертификацию всех средств, используемых для организации правительственной связи и информатизации государственного управления, а также лицензирует все предприятия, учреждения и организации, занимающиеся производством таких средств. Кроме того, в исключительном ведении ФАПСИ находятся вопросы сертификации и лицензирования в области криптографической защиты информации.

В министерствах и ведомствах создаются иерархические структуры обеспечения безопасности информации, которые, как правило, совпадают с организационной структу-

рой министерства (ведомства). Называться они могут по-разному, но функции выполняют сходные.

3.2. ОБЩАЯ ХАРАКТЕРИСТИКА ОРГАНИЗАЦИОННЫХ МЕТОДОВ ЗАЩИТЫ

Законы и нормативные акты исполняются только в том случае, если они подкрепляются организаторской деятельностью соответствующих структур, создаваемых в государстве, в ведомствах, учреждениях и организациях. При рассмотрении вопросов безопасности информации такая деятельность относится к организационным методам защиты информации.

Организационные методы защиты информации включают меры, мероприятия и действия, которые должны осуществлять должностные лица в процессе создания и эксплуатации ИВС для обеспечения заданного уровня безопасности информации.

Организационные методы защиты информации тесно связаны с правовым регулированием в области безопасности информации. В соответствии с законами и нормативными актами в министерствах, ведомствах, на предприятиях (независимо от форм собственности) для защиты информации создаются специальные службы безопасности. Эти службы подчиняются, руководству учреждения. Руководители служб организуют создание и функционирование систем защиты информации. На организационном уровне решаются следующие задачи обеспечения безопасности информации в ИВС:

- организация работ по разработке системы защиты инфор-мации;
- ограничение доступа на объект и к ресурсам КС;
- разграничение доступа к ресурсам КС;
- планирование мероприятий;
- разработка документации;
- воспитание и обучение обслуживающего персонала и пользователей;
- сертификация средств защиты информации;
- лицензирование деятельности по защите информации;
- аттестация объектов защиты;
- совершенствование системы защиты информации;
- оценка эффективности функционирования системы защиты информации;
- контроль выполнения установленных правил работы в КС.

Организационные методы являются стержнем комплексной системы защиты информации в КС. Только с помощью этих методов возможно объединение на правовой основе технических, программных и криптографических средств защиты информации в единую комплексную систему. Конкретные организационные методы защиты информации будут приводиться при рассмотрении парирования угроз безопасности информации. Наибольшее внимание организационным мероприятиям уделяется при изложении вопросов построения и организации функционирования комплексной системы защиты информации.

Контрольные вопросы

1. Перечислите задачи государства в области безопасности информации.
2. Охарактеризуйте основные законы РФ, регулирующие отношения в области информационных технологий.
3. Назовите государственные органы, обеспечивающие безопасность информационных технологий, и решаемые ими задачи.
4. Дайте общую характеристику организационным методам защиты информации в КС.

4. СТАНДАРТЫ И СПЕЦИФИКАЦИИ В ОБЛАСТИ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

4.1. ОБЩИЕ КРИТЕРИИ БЕЗОПАСНОСТИ

Вопросы обеспечения информационной безопасности (ИБ) исследуются в разных странах достаточно давно. Можно констатировать, что к настоящему времени сложилась общепринятая точка зрения на концептуальные основы ИБ. Суть ее заключается в том, что подход к обеспечению ИБ должен быть комплексным, сочетающим меры следующих уровней:

- законодательного (законы, нормативные акты, стандарты);
- административного (действия общего характера, предпринимаемые руководством организации);
- процедурного (меры безопасности, реализуемые персоналом);

– программно-технического (конкретные технические меры).

При обеспечении ИБ существует два аспекта: формальный – определение критериев, которым должны соответствовать защищенные информационные технологии, и практический – определение конкретного комплекса мер безопасности применительно к рассматриваемой информационной технологии.

Критерии, которым должны соответствовать защищенные информационные технологии, являются объектом стандартизации более пятнадцати лет. В настоящее время разработан проект международного стандарта «Общие критерии оценки безопасности информационных технологий».

Попытки стандартизации практических аспектов безопасности начались сравнительно недавно. Первой удачной попыткой в этой области стал *британский стандарт BS* 7799 «*Практические правила управления информационной безопасностью*», изданный в 1995 г., в котором обобщен опыт обеспечения режима ИБ в информационных системах (ИС) разного профиля. Впоследствии было опубликовано несколько аналогичных документов: стандарты различных организаций и ведомств, например *германский стандарт BSI*. Содержание этих документов, в основном, относится к этапу анализа рисков, на котором определяются угрозы безопасности и уязвимости информационных ресурсов, уточняются требования к режиму ИБ.

Идеи, содержащиеся в этих документах, заключаются в следующем. Практические правила обеспечения ИБ на всех этапах жизненного цикла информационной технологии должны носить комплексный характер и основываться на проверенных практикой приемах и методах. Например, в информационной технологии должны обязательно использоваться некоторые средства идентификации и аутентификации пользователей (сервисов), средства резервного копирования, антивирусный контроль и т.д. Режим ИБ в подобных системах обеспечивается:

- на процедурном уровне – путем разработки и выполнения разделов инструкций для персонала по ИБ, а также мерами физической защиты;
- на программно-техническом уровне – применением апробированных и сертифицированных решений, стандартного набора контрмер: резервное копирование, антивирусная защита, парольная защита, межсетевые экраны, криптографическая защита и т.д.

При обеспечении ИБ важно не упустить каких-либо существенных аспектов. Это будет гарантировать некоторый минимальный (базовый) уровень ИБ, обязательный для любой информационной технологии. Таким образом, для обеспечения базового уровня ИБ используется упрощенный подход к анализу рисков, при котором рассматривается стандартный набор наиболее распространенных угроз безопасности без оценки их вероятностей. Для нейтрализации угроз применяется типовой набор контрмер, а вопросы эффективности защиты не рассматриваются. Подобный подход приемлем, если ценность защищаемых ресурсов с точки зрения организации не является чрезмерно высокой. Методология анализа рисков для базового уровня безопасности, предлагаемая в документах и стандартах различных организаций, различается и будет рассмотрена в разделе «Базовый уровень информационной безопасности».

В ряде случаев базового уровня безопасности оказывается недо-статочно, например, АСУ технологическим процессом предприятия с непрерывным циклом производства или АСУ войсками, когда даже кратковременный выход из строя автоматизированной системы приводит к очень тяжелым последствиям. В этом и подобных случаях важно знать параметры, характеризующие уровень безопасности информационной системы (технологии): количественные оценки угроз безопасности, уязвимостей, ценности информационных ресурсов. В случае повышенных требований в области ИБ используется полный вариант анализа рисков. В отличие от базового варианта, в том или ином виде производится оценка ценности ресурсов, характеристик рисков и уязвимостей. Как правило, проводится анализ по критерию стоимость/эффективность нескольких вариантов защиты.

Общие критерии позволяют сравнивать результаты независимых оценок безопасности ИТ. Чтобы достигнуть большей сравнимости между результатами оценок, оценки должны быть выполнены в пределах структуры авторитетной схемы оценки, которая устанавливает стандарты и контролирует качество оценок. Такие схемы оценки в настоящее время существуют в нескольких странах и основаны на различных критериях оценки.

Общие критерии предназначены для задания требований и оценки безопасности ИТ и включают функциональные требования и требования гарантии оценки, сопровождаемые информационным материалом.

4.1.1. ПОДГОТОВКА И ЦЕЛЕВАЯ НАПРАВЛЕННОСТЬ ОБЩИХ КРИТЕРИЕВ

Общие критерии являются результатом усилий ряда организаций по разработке критериев оценки безопасности ИТ. В 80-х годах Критерии оценки безопасности компьютерных систем (TCSEC) были разработаны и развиты в США. В последующем десятиле-

тии различные страны продолжили развитие критериев оценки на основе концепций TCSEC, сделав их более гибкими и приспособленными к развитию ИТ.

В Европе *Критерии оценки безопасности информационных техно*логий (ITSEC) версия 1.2 были изданы в 1991 г. Европейской комиссией в результате совместных усилий Франции, Германии, Голландии и Англии.

В Канаде *Критерии оценки компьютерных систем* (CTCPEC) версия 3.1 были изданы в 1993 г. как комбинация подходов TCSEC и ITSEC.

В США проект *Федеральных критериев для оценки безопасности информационных технологий* (FC) версия 1 был также издан в 1993 г. как второй шаг к объединению Американской и Европейской концепций для критериев оценки.

В 1990 г. Международная организация по стандартизации (ИСО) начала работу по разработке международных стандартов по критериям оценки безопасности ИТ для общего использования. Новые критерии должны были быть адаптированы к потребностям взаимного признания результатов оценки безопасности ИТ в глобальном масштабе. Эта задача была возложена на рабочую группу 3 (WG 3) из подкомиссии 27 (SC 27) ИСО.

В июне 1993 г. авторы ITSEC, TCSEC, FC, CTCPEC объединили свои усилия и начали разработку проекта *Общих критериев оценки безопасности информационных технологий* (CCEB). Цель проекта состояла в том, чтобы исключить концептуальные и технические различия, имеющиеся в исходных критериях, и представить результаты в ИСО как проект международного стандарта.

В январе 1996 г. была выпущена версия 1.0 Общих критериев (ОК), а в 1997 г. были выпущены дополнительные материалы к ней. Выпуск версии 2.0 осуществлен в декабре 1999 г.

Оценка безопасности ИТ – это методология исследования свойств безопасности изделия или системы информационных технологий, называемых в ОК объектами оценки.

При этом могут быть идентифицированы три группы пользователей с общим интересом к этим оценкам: потребители объекта оценки, разработчики объекта оценки и оценщики объекта оценки. Общие критерии разработаны таким образом, чтобы удовлетворить потребности всех трех групп пользователей.

Потребители могут использовать оценку для сравнения различные изделия или системы и решения о выполнении требования по безопасности. Общие критерии играют важную роль при задании потребителем функциональных требований к безопасности ИТ и определении возможности использования предопределенной структуры требований, названной «профилем защиты». Они помогают разработчикам при подготовке к оценке и оценке их изделий или систем на соответствие функциональным требованиям безопасности и требованиям гарантии оценки, содержащимся в «задании по безопасности». Также ОК содержат критерии, которые нужно использовать оценщикам при формировании заключений относительно соответствия объектов оценки требованиям безопасности.

4.1.2. ОРГАНИЗАЦИЯ ОБЩИХ КРИТЕРИЕВ

Общие критерии – это совокупность самостоятельных, но взаимосвязанных частей.

Представление и общая модель – определяет общую концепцию и принципы оценки безопасности ИТ, общую модель оценки, а также конструкции для формирования целей безопасности ИТ, для выбора и определения требований безопасности ИТ и для описания спецификаций высокого уровня для изделий и систем. Кроме того, в ней приведены категории пользователей с указанием на различные части ОК, где представлены их интересы к критериям оценки безопасности.

Требования к функциям безопасности – устанавливает набор функциональных компонентов как стандартный путь выражения функциональных требований к объектам оценки.

Требования гарантии безопасности – включает компоненты требований гарантии оценки, сгруппированные в семейства и классы, а также уровни гарантии оценки, которые определяют ранжирование по степени удовлетворения требований, определяет также критерии оценки для *профилей защиты* и *заданий по безопасности*.

Предопределенные профили защиты – содержат примеры профилей защиты, включающие функциональные требования безопасности и требования гарантии оценки, которые были идентифицированы в исходных критериях (ITSEC, CTCPEC, FC, TCSEC), а также требования, не представленные в исходных критериях.

4.1.3. ВОЗМОЖНОСТИ И ПРИМЕНИМОСТЬ

ОК поддерживают выбор и оценку безопасности объекта ИТ. ОК полезны при раз-

работке изделий или систем ИТ и при приобретении коммерческих изделий и систем с функциями безопасности. ОК дают основу для оценки объекта, чтобы установить уровень доверия к безопасности ИТ.

К таким объектам относятся, например, операционные системы, сети компьютеров, распределенные системы, прикладные программы.

Аспекты безопасности ИТ включают защиту информации от несанкционированного раскрытия, модификации или потери возможности использования при воздействии угроз, являющихся результатом преднамеренных или непреднамеренных действий человека. ОК могут быть также применимы и к другим аспектам безопасности ИТ.

ОК применимы при оценке безопасности ИТ, включая как аппаратные средства, так и программное обеспечение.

Некоторые аспекты безопасности ИТ находятся вне рамок ОК. К ним относятся следующие:

1) ОК не охватывают оценку административных мер безопасности. Административные меры безопасности в окружающей среде объекта оценки рассматриваются только если они могут противостоять угрозам;

2) в ОК не рассматривается оценка технических аспектов безопасности ИТ типа электромагнитного излучения;

3) ОК формулируют только критерии оценки и не содержат методик самой оценки, а также административных структур, которые должны их использовать;

4) вне рамок ОК процедуры для использования результатов оценки при приеме системы в эксплуатацию;

5) в ОК не входят критерии для оценки специфических качеств криптографических методов и алгоритмов защиты информации.

4.1.4. КОНЦЕПЦИИ ОБЩИХ КРИТЕРИЕВ

В соответствии с концепцией ОК требования безопасности объекта оценки подразделяются на две категории: функциональные требования и требования гарантированности.

В *функциональных требованиях ОК* описаны те функции объекта оценки, которые обеспечивают безопасность ИТ. Например, функциональные требования включают требования идентификации, установления подлинности (аутентификации) пользователей, протоколирования (аудита) и др.

Требования гарантированности отражают качества объекта оценки, дающие основание для уверенности в том, что требуемые меры безопасности объекта реализованы правильно и эффективны. Гарантированность получается на основе изучения назначения, структуры и функционирования объекта оценки.

В ОК функциональные требования и требования гарантированности представлены в одном и том же общем стиле и используют одну и ту же организацию и терминологию.

Термин *класс* используется для наиболее общей группировки требований безопасности. Все члены класса разделяют общее намерение при отличии в охвате целей безопасности.

Члены класса названы *семействами*. Семейство – группировка наборов требований безопасности, которые обеспечивают выполнение определенной части целей безопасности, но могут отличаться в акценте или жесткости.

Члены семейства названы *компонентами.* Компонент описывает определенный набор требований безопасности – наименьший набор требований безопасности для включения в структуры, определенные в ОК.

Компоненты построены из *элементов*. Элемент – самый нижний и неделимый уровень требований безопасности, на котором производится оценка их удовлетворения.

Организация требований безопасности в ОК по иерархии *класс–семейство–компонент–элемент* помогает потребителю правильно определить компоненты, как только будут идентифицированы угрозы безопасности объекта оценки.

Компоненты в семействе могут находиться в *иерархической связи*, когда необходимо наращивание требований для выполнения одной из целей безопасности, или нет, когда имеет место качественно новое требование.

Между компонентами могут существовать *зависимости*, которые возникают, когда компонент сам недостаточен для выполнения цели безопасности и необходимо наличие другого компонента. Зависимости могут существовать между функциональными компонентами, компонентами гарантированности и между теми и другими. Чтобы гарантировать законченность требований к объекту оценки, зависимости должны быть учтены при включении компонентов в *профиль защиты* и *задание по безопасности*. Компоненты могут быть преобразованы с помощью *разрешенных действий*, чтобы обеспечить выполнение определенной политики безопасности или противостоять определенной угрозе. Не все действия допустимы на всех компонентах. Каждый компонент идентифицирует и

определяет разрешенные действия или обстоятельства, при которых действие может применяться к компоненту, и результаты применения действия. К разрешенным действиям относятся: назначение, выбор и обработка.

Назначение – разрешает заполнить спецификацию идентифицированного параметра при использовании компонента. Параметр может быть признаком или правилом, которое конкретизирует требование к определенной величине или диапазону величин.

Выбор – это действие выбора одного или большего количества пунктов из списка, чтобы конкретизировать возможности элемента.

Обработка – позволяет включить дополнительные детали в элемент и предполагает интерпретацию требования, правила, константы или условия, основанную на целях безопасности. Обработка должна только ограничить набор возможных приемлемых функций или механизмов, чтобы осуществить требования, но не увеличивать их. Обработка не позволяет создавать новые требования или удалять существующие и не влияет на список зависимостей, связанных с компо-нентом.

ОК определяют также *набор структур*, которые объединяют компоненты требований безопасности.

Промежуточная комбинация компонентов названа *пакетом*. Пакет включает набор требований, которые обеспечивают выполнение поднабора целей безопасности. Пакет предназначен для многократного использования и определяет требования, которые являются необходимыми для достижения идентифицированных целей. Пакет может использоваться для формирования профилей защиты и заданий по безопасности.

Уровни гарантии оценки – предопределенные пакеты требований гарантии. Уровень гарантированности – это набор базовых требований гарантии для оценки. Каждый из уровней содержит полный набор требований гарантии и определяет масштаб гарантии в ОК.

Профиль защиты содержит набор функциональных требований и компонентов требований гарантированности, включенных в соответствующий уровень гарантии оценки. Профиль защиты предназначен для многократного использования и определяет совокупность требований безопасности к объекту оценки, которые являются необходимыми и эффективными для достижения поставленных целей.

Задание по безопасности содержит набор требований безопасности, которые могут быть представлены ссылкой на профиль защиты, непосредственно на требования ОК или сформулированы в явном виде. Задание по безопасности выражает требования безопасности для конкретного объекта оценки.

В задании по безопасности предусматривается возможность включения функциональных требований, не содержащихся в ОК. Однако, при включении новых компонентов в ЗБ необходимо учитывать следующее:

1. Такие требования должны быть четко и недвусмысленно сформулированы, чтобы их оценка и демонстрация соответствия были выполнимы. Уровень детализации и способ выражения соответствующих требований ОК должен использоваться как образец.
2. Результаты оценки, полученные с использованием функциональных компонентов, не входящих в ОК, и требований гарантированности, не входящих в ОК, должны быть также квалифицированы. Включение новых требований в Задание по Безопасности не только требует соответствия структуре и правилам ОК, но и не гарантирует универсальное принятие результатов оценки различными специалистами.

4.2. ДЕЙСТВУЮЩИЕ СТАНДАРТЫ И РЕКОМЕНДАЦИИ В ОБЛАСТИ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

К основополагающим документам в области информационной безопасности относятся:

- «Оранжевая книга» (TCSEC);
- «Радужная серия»;
- «Гармонизированные критерии Европейских стран» (ITSEC);
- «Концепция защиты от НСД» Гостехкомиссии при Президенте РФ;
- «Рекомендации X.800».

4.2.1. КРИТЕРИИ ОЦЕНКИ НАДЕЖНЫХ КОМПЬЮТЕРНЫХ СИСТЕМ («ОРАНЖЕВАЯ КНИГА» МИНИСТЕРСТВА ОБОРОНЫ США)

Данный труд, называемый чаще всего по цвету обложки «Оранжевой книгой», был впервые опубликован в августе 1983 г. Уже его название заслуживает комментария. Речь идет не о безопасных, а о надежных системах, причем слово «надежный» трактуется так же, как в сочетании «надежный человек» – человек, которому можно доверять.

«Оранжевая книга» поясняет понятие безопасной системы, которая «управляет, посредством соответствующих средств, доступом к информации, так что только должным образом авторизованные лица или процессы, действующие от их имени, получают право

читать, писать, создавать и удалять информацию». Очевидно, что абсолютно безопасных систем не существует, это абстракция. Любую систему можно «взломать», если располагать достаточно большими материальными и временными ресурсами. Есть смысл оценивать степень доверия, которое разумно оказать той или иной системе.

В «Оранжевой книге» *надежная система* определяется как «система, использующая достаточные аппаратные и программные средства, чтобы обеспечить одновременную обработку информации разной степени секретности группой пользователей без нарушения прав доступа».

Степень доверия (надежность систем) оценивается по двум основным критериям:

Политика безопасности – набор законов, правил и норм поведения, определяющих, как организация обрабатывает, защищает и распространяет информацию. Например, правила определяют, в каких случаях пользователь имеет право оперировать с определенными наборами данных и чем надежнее система, тем строже и многообразнее должна быть политика безопасности. В зависимости от сформулированной политики можно выбирать конкретные механизмы, обеспечивающие безопасность системы. Политика безопасности – это активный компонент защиты, включающий в себя анализ возможных угроз и выбор мер противодействия.

Гарантированность – это мера доверия, которая может быть оказана архитектуре и реализации системы. Гарантированность может проистекать как из тестирования, так и из проверки общего замысла и исполнения системы в целом и ее компонентов.

Надежная вычислительная база – это совокупность защитных механизмов компьютерной системы (включая аппаратное и программное обеспечение), отвечающих за проведение в жизнь политики безопасности.

Основные элементы *политики безопасности*:

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

Произвольное управление доступом – это метод ограничения доступа к объектам, основанный на учете личности субъекта или группы, в которую субъект входит. Произвольность управления состоит в том, что некоторое лицо может по своему усмотрению давать другим субъектам или отбирать у них права доступа к объекту.

С концептуальной точки зрения текущее состояние прав доступа при произвольном управлении описывается матрицей, в строках которой перечислены субъекты, а в столбцах – объекты. В клетках, расположенных на пересечении строк и столбцов, записываются способы доступа, допустимые для субъекта по отношению к объекту – например, чтение, запись, выполнение, возможность передачи прав другим субъектам и т.п.

В операционных системах компактное представление матрицы доступа основывается или на структурировании совокупности субъектов (владелец/группа/прочие в ОС UNIX), или на механизме списков управления доступом, то есть на представлении матрицы по столбцам, когда для каждого объекта перечисляются субъекты вместе с их правами доступа. За счет использования метасимволов можно компактно описывать группы субъектов, удерживая тем самым размеры списков управления доступом в разумных рамках.

Большинство операционных систем и систем управления базами данных реализуют именно произвольное управление доступом. Главное его достоинство – гибкость; главные недостатки – рассредоточенность управления и сложность централизованного контроля, а также оторванность прав доступа от данных, что позволяет копировать секретную информацию в общедоступные файлы.

Безопасность повторного использования объектов – важное на практике дополнение средств управления доступом, предохраняющее от случайного или преднамеренного извлечения секретной информации из «мусора». Безопасность повторного использования должна гарантироваться для областей оперативной памяти, для магнитных и других носителей.

Поскольку информация о субъектах также представляет собой объект, необходимо позаботиться о безопасности «повторного использования субъектов». Когда пользователь покидает организацию, следует не только лишить его возможности входа в систему, но и запретить доступ ко всем объектам. В противном случае, новый сотрудник может получить ранее использовавшийся идентификатор, а с ним и все права своего предшественника.

Современные интеллектуальные периферийные устройства усложняют обеспечение безопасности повторного использования объектов. Действительно, принтер может буферизовать несколько страниц документа, которые останутся в памяти даже после окончания печати.

Для реализации принудительного управления доступом с субъектами и объектами

ассоциируются *метки безопасности*. Метка субъекта описывает его благонадежность, метка объекта – степень закрытости содержащейся в нем информации.

Согласно «Оранжевой книге», метки безопасности состоят из двух частей – уровня секретности и списка категорий. Уровни секретности, поддерживаемые системой, образуют упорядоченное множество, которое может выглядеть, например, так: совершенно секретно; секретно; конфиденциально; несекретно.

Главная проблема, которую необходимо решать в связи с метками, это обеспечение их целостности:

- не должно быть непомеченных субъектов и объектов;
- при любых операциях с данными метки должны оставаться правильными.

В особенности это относится к экспорту и импорту данных. Например, печатный документ должен открываться заголовком, содержащим текстовое и/или графическое представление метки безопасности. Аналогично, при передаче файла по каналу связи должна передаваться и ассоциированная с ним метка, причем в таком виде, чтобы удаленная система могла ее протрактовать, несмотря на возможные различия в уровнях секретности и наборе категорий.

Одним из средств обеспечения целостности меток безопасности является разделение устройств на многоуровневые и одноуровневые. На многоуровневых устройствах может храниться информация разного уровня секретности. Одноуровневое устройство можно рассматривать как вырожденный случай многоуровневого, когда допустимый диапазон состоит из одного уровня. Зная уровень устройства, система может решить, допустимо ли записывать на него информацию с определенной меткой. Например, попытка напечатать совершенно секретную информацию на принтере общего пользования с уровнем «несекретно» потерпит неудачу.

Метки безопасности, ассоциируемые с субъектами, более подвижны, чем метки объектов. Субъект может в течение сеанса работы с си-стемой изменять свою метку, естественно, не выходя за предопределенные для него рамки. Иными словами, он может сознательно занижать свой уровень благонадежности, чтобы уменьшить вероятность непреднамеренной ошибки. Принцип минимизации привилегий – весьма разумное средство защиты.

Принудительное 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.

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. В терминах принудительного управления нельзя выразить предложение «разрешить доступ к объекту *X* еще и для пользователя *Y*». Конечно, можно изменить метку безопасности пользователя *Y*, но тогда он скорее всего, получит доступ ко многим дополнительным объектам, а не только к *X*.

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

1) идентификация и аутентификация;
2) предоставление надежного пути;
3) анализ регистрационной информации.

Рассмотрим эти категории подробнее.

Идентификация и аутентификация. Каждый пользователь, прежде чем получить право совершать какие-либо действия в системе, должен идентифицировать себя. Обычный способ идентификации – ввод имени пользователя при входе в систему. В свою очередь, система должна проверить подлинность личности пользователя, то есть что он является именно тем, за кого себя выдает. Стандартное средство проверки подлинности (аутентификации) – пароль, хотя в принципе могут использоваться также разного рода личные карточки, биометрические устройства (сканирование роговицы или отпечатков пальцев) или их комбинация.

Идентификация и аутентификация – первый и важнейший программно-технический рубеж информационной безопасности. Если не составляет проблемы получить доступ к системе под любым именем, то другие механизмы безопасности, теряют смысл. Очевид-

но и то, что без идентификации пользователей невозможно протоколирование их действий.

Надежный путь связывает пользователя непосредственно с надежной вычислительной базой, минуя другие, потенциально опасные компоненты системы. Цель предоставления надежного пути – дать пользователю возможность убедиться в подлинности обслуживающей его системы.

Относительно несложно реализовать надежный путь, если используется неинтеллектуальный терминал – достаточно иметь зарезервированную управляющую последовательность (при условии защищенности линии связи между терминалом и системой). Если же пользователь общается с интеллектуальным терминалом, персональным компьютером или рабочей станцией, задача обеспечения надежного пути становится чрезвычайно сложной, если вообще разрешимой.

Анализ регистрационной информации. Аудит имеет дело с действиями (событиями), затрагивающими безопасность системы. К их числу относятся:

- вход в систему и выход из нее;
- обращение к удаленной системе;
- операции с файлами (открыть, закрыть, переименовать, удалить);
- смена привилегий или иных атрибутов безопасности (режима доступа, уровня благонадежности пользователя и т.п.).

Протоколирование помогает следить за пользователями и реконструировать прошедшие события. Слежка важна, в первую очередь, как профилактическое средство. Реконструкция событий позволяет проанализировать случаи нарушений, понять причину, оценить размеры ущерба и принять меры по недопущению подобных нарушений.

При протоколировании события записывается следующая информация: дата и время события; уникальный идентификатор пользователя – инициатора действия; тип события; результат действия (успех или неудача); источник запроса (например, имя терминала); имена затронутых объектов (например, открываемых или удаляемых файлов); описание изменений, внесенных в базы данных защиты (например, новая метка безопасности объекта); метки безопасности субъектов и объектов события.

Гарантированность – это мера уверенности, с которой можно утверждать, что для проведения в жизнь сформулированной политики безопасности выбран подходящий набор средств, и что каждое из этих средств правильно исполняет отведенную ему роль.

В «Оранжевой книге» рассматривается два вида гарантированности – операционная и технологическая. Операционная гарантированность относится к архитектурным и реализационным аспектам системы, в то время как технологическая – к методам построения и сопровождения.

Операционная гарантированность включает в себя проверку следующих элементов: архитектура системы; целостность системы; анализ тайных каналов передачи информации; надежное администрирование; надежное восстановление после сбоев. Операционная гарантированность – это способ убедиться в том, что архитектура системы и ее реализация действительно проводят в жизнь избранную политику безопасности.

Технологическая гарантированность охватывает весь жизненный цикл системы, т.е. периоды проектирования, реализации, тестирования, продажи и сопровождения. Все перечисленные действия должны выполняться в соответствии с жесткими стандартами, чтобы обезопаситься от утечки информации и нелегальных «закладок».

Документация – необходимое условие гарантированной надежности системы и, одновременно, – инструмент проведения политики безопасности. Без документации люди не будут знать, какой политике следовать и что для этого нужно делать.

Согласно «Оранжевой книге», в комплект документации надежной системы должны входить следующие тома: руководство пользователя по средствам безопасности; руководство администратора по средствам безопасности; тестовая документация; описание архитектуры; описание политики безопасности данной организации.

Классы безопасности. «Критерии…» Министерства обороны США открыли путь к ранжированию информационных систем по степени надежности. В «Оранжевой книге» определяется четыре уровня безопасности (надежности) – *D*, *C*, *B* и *A*. Уровень *D* предназначен для систем, признанных неудовлетворительными. В настоящее время он содержит две подсистемы управления доступом для ПК. По мере перехода от уровня *C* к *A* к надежности систем предъявляются все более жесткие требования. Уровни *C* и *B* подразделяются на классы (*C*1, *C*2, *B*1, *B*2, *B*3) с постепенным возрастанием надежности. Таким образом, всего имеется шесть классов безопасности – *C*1, *C*2, *B*1, *B*2, *B*3, *A*1. Чтобы система в результате процедуры сертификации могла быть отнесена к некоторому классу, ее политика безопасности и гарантированность должны удовлетворять приводимым к данному классу требованиям. Требования к классам безопасности приведены в [23].

Требования к политике безопасности и к гарантированности распределены по классам безопасности. В «младших» классах политика довольно быстро ужесточается, по

существу достигая пика к классу *B*1. Напротив, меры гарантированности отнесены в основном в «старшие» классы, начиная с *B*2. Это подтверждает независимость двух основных групп критериев надежности и методологическую целесообразность их разделения по европейскому образцу.

Распределение требований по классам вызывает ряд конкретных возражений. Неоправданно далеко отодвинуты такие очевидные требования, как извещение о нарушении защиты, конфигурационное управление, безопасный запуск и восстановление после сбоев. Возможно, это оправдано в физически защищенной военной среде, но никак не в коммерческой, когда постоянное слежение за перемещениями сотрудников может быть очень дорогим удовольствием.

4.2.2. ГАРМОНИЗИРОВАННЫЕ КРИТЕРИИЕВРОПЕЙСКИХ СТРАН

Следуя по пути интеграции, европейские страны приняли согласованные критерии оценки безопасности информационных технологий (Information Technology Security Evaluation Criteria, ITSEC). Изложение этих Критериев основывается на версии 1.2, опубликованной в июне 1991 г. от имени соответствующих органов четырех стран – Франции, Германии, Нидерландов и Великобритании. Выгода от использования согласованных критериев очевидна для всех – и для производителей, и для потребителей, и для самих органов сертификации.

Принципиально важной чертой европейских критериев является отсутствие априорных требований к условиям, в которых должна работать информационная система. Требования к политике безопасности и к наличию защитных механизмов не являются составной частью критериев. Впрочем, чтобы облегчить формулировку цели оценки, критерии содержат в качестве приложения описание десяти примерных классов функциональности, типичных для правительственных и коммерческих систем.

Европейские критерии рассматривают следующие составляющие информационной безопасности:

- *конфиденциальность* – защита от несанкционированного получения информации;
- *целостность* – защита от несанкционированного изменения информации;
- *доступность* – защита от несанкционированного удержания информации и ресурсов.

В критериях проводится различие между системами и продуктами. *Система* – это конкретная аппаратно-программная конфигурация, построенная с вполне определенными целями и функционирующая в известном окружении. *Продукт* – это аппаратно-программный комплекс, который можно купить и по своему усмотрению встроить в ту или иную систему.

Каждая система и/или продукт предъявляет свои требования к обеспечению конфиденциальности, целостности и доступности. Чтобы удовлетворить эти требования, необходимо предоставить соответствующий набор функций (сервисов) безопасности, таких как идентификация и аутентификация, управление доступом или восстановление после сбоев.

С точки зрения ИБ основное отличие между системой и продуктом состоит в том, что система имеет конкретное окружение, которое можно определить и изучить сколь угодно детально, а продукт должен быть рассчитан на использование в различных условиях. Угрозы безопасности системы носят вполне конкретный и реальный характер. Относительно угроз продукту можно лишь строить предположения. Разработчик может специфицировать условия, пригодные для функционирования продукта; дело покупателя – обеспечить выполнение этих условий.

Из практических соображений важно обеспечить единство критериев оценки продуктов и систем – например, чтобы облегчить и удешевить оценку системы, составленной из ранее сертифицированных продуктов. В этой связи для систем и продуктов вводится единый термин – объект оценки. В соответствующих местах делаются оговорки, какие требования относятся исключительно к системам, а какие – только к продуктам.

Каждая система и/или продукт предъявляет свои требования к обеспечению конфиденциальности, целостности и доступности. Чтобы удовлетворить эти требования, необходимо предоставить соответствующий набор функций (сервисов) безопасности, таких как идентификация и аутентификация, управление доступом или восстановление после сбоев.

Сервисы безопасности реализуются посредством конкретных механизмов. Например, для реализации функции идентификации и аутентификации можно использовать такой механизм, как сервер аутентификации Kerberos.

Чтобы объект оценки можно было признать надежным, необходима определенная степень уверенности в наборе функций и механизмов безопасности, которую будем называть *гарантированностью*.

Гарантированность затрагивает два аспекта – *эффективность* и *корректность средств безопасности*. При проверке эффективности анализируется соответствие между целями, сформулированными для объекта оценки, и имеющимся набором функций безопасности, т.е. рассматриваются вопросы адекватности функциональности, взаимной согласованности функций, простоты их использования, а также возможные последствия эксплуатации известных слабых мест защиты. Кроме того, в понятие эффективности входит способность механизмов защиты противостоять прямым атакам (мощность механизма). Определяется три градации мощности – базовая, средняя и высокая.

Под корректностью понимается правильность реализации функций и механизмов безопасности. В критериях определяется семь возможных уровней гарантированности корректности – от *E*0 до *E*6 (в порядке возрастания от уровня *E*0 – отсутствие гарантированности (аналог уровня *D* «Оранжевой книги»).

Назовем основные элементы политики безопасности.

Функциональность. В европейских критериях средства, имеющие отношение к информационной безопасности, рассматриваются на трех уровнях детализации. Наиболее абстрактный – первый уровень – касается лишь целей безопасности. Второй уровень содержит спецификации функций безопасности. На третьем уровне содержится информация о механизмах безопасности, т.е. как реализуется декларированная функциональность.

Спецификации функций безопасности – важнейшая часть описания объекта оценки. Критерии рекомендуют выделить в этих спецификациях разделы со следующими заголовками: идентификация и аутентификация; управление доступом; подотчетность; аудит; повторное использование объектов; точность информации; надежность обслуживания; обмен данными.

Большинство из перечисленных тем рассматривались ранее при анализе «Оранжевой книги». Здесь остановимся лишь на специфичных для европейских критериев.

Под идентификацией и аутентификацией понимается не только проверка подлинности пользователей в узком смысле, но и функции для регистрации новых пользователей и удаления старых, а также функции для генерации, изменения и проверки аутентификационной информации, в том числе средства контроля целостности. Сюда же относятся функции для ограничения числа повторных попыток аутентификации.

Средства управления доступом также трактуются европейскими критериями достаточно широко. В этот раздел попадают, помимо прочих, функции, обеспечивающие временное ограничение доступа к совместно используемым объектам с целью поддержания целостности этих объектов – мера, типичная для систем управления базами данных. В этот же раздел попадают функции для управления распространением прав доступа и для контроля за получением информации путем логического вывода и агрегирования данных (типично для СУБД).

Под точностью в критериях понимается поддержание определенного соответствия между различными частями данных (точность связей) и обеспечение неизменности данных при передаче между процессами (точность коммуникаций). Точность выступает как один из аспектов целостности информации.

Функции надежности обслуживания должны гарантировать, что действия, критичные по времени, будут выполнены ровно тогда, когда нужно – не раньше и не позже, и что некритичные действия нельзя перевести в разряд критичных. Далее, должна быть гарантия, что авторизованные пользователи за разумное время получат запрашиваемые ресурсы. Сюда же относятся функции для обнаружения и нейтрализации ошибок, необходимые для минимизации простоев, а также функции планирования, позволяющие гарантировать время реакции на внешние события.

К области обмена данными относятся функции, обеспечивающие коммуникационную безопасность, т.е. безопасность данных, передаваемых по каналам связи. Здесь европейские критерии следуют в фарватере рекомендаций X.800, предлагая следующие подзаголовки: аутентификация; управление доступом; конфиденциальность данных; целостность данных; невозможность отказаться от совершенных действий.

Классы безопасности. Набор функций безопасности может специфицироваться с использованием ссылок на заранее определенные классы функциональности. В европейских критериях таких классов десять. Пять из них (*F-C*1, *F-C*2, *F-B*1, *F-B*2, *F-B*3) соответствуют классам безопасности «Оранжевой книги».

Класс *F-IN* предназначается для объектов оценки с высокими потребностями по обеспечению целостности данных и программ, что типично для систем управления базами данных. При описании класса *F-IN* вводится понятие роли, выдвигается требование по предоставлению доступа к определенным объектам только с помощью предопределенных процессов. Должны различаться следующие виды доступа: чтение, запись, добавление, удаление, переименование (для всех объектов), выполнение, удаление, переименование (для выполняемых объектов), создание и удаление объектов.

Класс *F-AV* характеризуется повышенными требованиями к доступности. Объект

оценки должен восстанавливаться после отказа отдельного аппаратного компонента таким образом, чтобы все критически важные функции оставались постоянно доступными. То же должно быть верно для вставки отремонтированного компонента, причем после этого объект оценки возвращается в состояние, устойчивое к одиночным отказам. Независимо от уровня загрузки должно гарантироваться время реакции на определенные события и отсутствие тупиков.

Класс *F-DI* характеризуется повышенными требованиями к целостности передаваемых данных. Перед началом общения стороны должны быть в состоянии проверить подлинность друг друга. При получении данных должна предоставляться возможность проверки подлинности источника. При обмене данными должны предоставляться средства контроля ошибок и их исправления.

Класс *F-DC* характеризуется повышенными требованиями к конфиденциальности передаваемой информации. Перед поступлением данных в каналы связи должно автоматически выполняться шифрование с использованием сертифицированных средств. На приемном конце также автоматически производится расшифровка. Ключи шифрования должны быть защищены от несанкционированного доступа.

Класс *F-DX* характеризуется повышенными требованиями и к целостности, и к конфиденциальности информации. Его можно рассматривать как объединение классов *F-DI* и *F-DC* с дополнительными возможностями шифрования, действующими из конца в конец, и с защитой от анализа трафика по определенным каналам.

Гарантированность эффективности. Для получения гарантий эффективности средств безопасности рассматриваются следующие вопросы:

– соответствие набора функций безопасности, т.е. их пригодность для противодействия угрозам, перечисленным в описании объекта оценки;

– взаимная согласованность различных функций и механизмов безопасности;

– способность механизмов безопасности противостоять прямым атакам;

– возможность практического использования слабостей в архитектуре объекта оценки, т.е. наличие способов отключения, обхода, повреждения и обмана функций безопасности;

– возможность небезопасного конфигурирования или использования объекта оценки при условии, что администраторы и/или пользователи имеют основание считать ситуацию безопасной;

– возможность практического использования слабостей в функционировании объекта оценки.

Важнейшей частью проверки эффективности является анализ слабых мест в защите объекта оценки. Цель анализа – найти все возможности отключения, обхода, повреждения, обмана средств защиты. Оценивается также способность всех критически важных защитных механизмов противостоять прямым атакам – мощность механизмов. Защищенность системы или продукта не может быть выше мощности самого слабого из критически важных механизмов, поэтому в критериях имеется в виду минимальная гарантированная мощность. Для нее определены три уровня – базовый, средний и высокий.

Согласно критериям, мощность можно считать *базовой*, если механизм способен противостоять отдельным случайным атакам.

Мощность считается *средней*, если механизм способен противостоять злоумышленникам с ограниченными ресурсами и возможностями.

Мощность можно считать *высокой*, если есть уверенность, что механизм может быть побежден только злоумышленником с высокой квалификацией, набор возможностей и ресурсов которого выходит за пределы практичности.

Эффективность защиты признается неудовлетворительной, если выявляются слабые места, допускающие практическое использование, и эти слабости не исправляются до окончания процесса оценки. В таком случае объекту присваивается уровень гарантированности *E*0.

Общая оценка системы складывается из минимальной мощности механизмов безопасности и уровня гарантированности корректности. Теоретически эти два аспекта независимы, хотя на практике нет смысла проверять правильность реализации «по высшему разряду», если механизмы безопасности не обладают даже средней мощностью.

4.2.3. РУКОВОДЯЩИЕ ДОКУМЕНТЫ ПО ЗАЩИТЕ ОТ НЕСАНКЦИОНИРОВАННОГО ДОСТУПА ГОСТЕХКОМИССИИ ПРИ ПРЕЗИДЕНТЕ РФ

С 1992 г. Гостехкомиссия при Президенте РФ опубликовала девять руководящих документов, посвященных проблеме защиты от несанкционированного доступа (НСД) к информации.

1. Концепция защиты СВТ и АС от НСД к информации (1992);
2. Средства вычислительной техники. Защита от НСД к информации. Показатели защищенности от НСД к информации (1992);
3. Защита от НСД к информации. Термины и определения (1992);
4. Временное положение по организации разработки, изготовления и эксплуатации программных и технических средств защиты информации от НСД в АС и СВТ (1992);
5. Положение по аттестации объектов информатизации по требованиям безопасности информации (1994);
6. Автоматизированные системы. Защита от несанкционированного доступа к информации. Классификация АС и требования к защите информации (1997);
7. Средства вычислительной техники. Межсетевые экраны. Защита от НСД к информации. Показатели защищенности от НСД к информации (1997);
8. Защита от несанкционированного доступа к информации. Ч. 1. Программное обеспечение средств защиты информации. Классификация по уровню контроля отсутствия недекларированных возможностей (1999);
9. Специальные требования и рекомендации по технической защите конфиденциальной информации (2001).

Рассмотрим важнейшие из них.

Концепция защиты СВТ и АС от НСД к информации

Концепция защиты СВТ и АС от НСД к информации является идейной основой набора руководящих документов. Она излагает систему взглядов, основных принципов, которые закладываются в основу проблемы защиты информации от несанкционированного доступа, являющейся частью общей проблемы безопасности информации.

В концепции различаются понятия средств вычислительной техники (СВТ) и автоматизированной системы (АС), аналогично тому, как в Европейских Критериях проводится деление на продукты и системы.

В концепции формулируются следующие основные принципы защиты от НСД к информации:

- защита СВТ обеспечивается комплексом программно-технических средств;
- защита АС обеспечивается комплексом программно-технических средств и поддерживающих их организационных мер;
- защита АС должна обеспечиваться на всех технологических этапах обработки информации и во всех режимах функционирования, в том числе при проведении ремонтных и регламентных работ;
- программно-технические средства защиты не должны существенно ухудшать основные функциональные характеристики АС (надежность, быстродействие, возможность изменения конфигурации АС);
- неотъемлемой частью работ по защите является оценка эффективности средств защиты, осуществляемая по методике, учитывающей всю совокупность технических характеристик оцениваемого объекта, включая технические решения и практическую реализацию средств защиты;
- защита АС должна предусматривать контроль эффективности средств защиты от НСД. этот контроль может быть либо периодическим, либо инициироваться по мере необходимости пользователем АС или контролирующими органами.

Концепция ориентируется на физически защищенную среду, проникновение в которую посторонних лиц считается невозможным, поэтому нарушитель определяется как субъект, имеющий доступ к работе с штатными средствами АС и СВТ как части АС.

Нарушители классифицируются по уровню возможностей, предоставляемых им штатными средствами АС и СВТ. При этом выделяется четыре уровня этих возможностей, а классификация является иерархической, т.е. каждый следующий уровень включает в себя функциональные возможности предыдущего.

Первый уровень определяет самый низкий уровень возможностей ведения диалога в АС – запуск задач (программ) из фиксированного набора, реализующих заранее предусмотренные функции по обработке информации.

Второй уровень определяется возможностью создания и запуска собственных программ с новыми функциями по обработке информации.

Третий уровень определяется возможностью управления функционированием АС, т.е. воздействием на базовое программное обеспечение системы и на состав и конфигурацию ее оборудования.

Четвертый уровень определяется всем объемом возможностей лиц, осуществляющих проектирование, реализацию и ремонт технических средств АС, вплоть до включе-

ния в состав СВТ собственных технических средств с новыми функциями по обработке информации.

В своем уровне нарушитель является специалистом высшей квалификации, знает все о АС и, в частности, о системе и средствах ее защиты.

Главным средством защиты от НСД в концепции рассматривается система разграничения доступа (СРД) субъектов к объектам доступа.

Основными функциями СРД являются:

– реализация правил разграничения доступа (ПРД) субъектов и их процессов к данным;

– реализация ПРД субъектов и их процессов к устройствам создания твердых копий;

– изоляция программ процесса, выполняемого в интересах субъекта, от других субъектов;

– управление потоками данных с целью предотвращения записи данных на носители несоответствующего грифа;

– реализация правил обмена данными между субъектами для АС и СВТ, построенных по сетевым принципам.

Кроме того, концепция предусматривает наличие обеспечивающих средств для СРД, которые выполняют следующие функции:

– идентификацию и опознание (аутентификацию) субъектов и поддержание привязки субъекта к процессу, выполняемому для субъекта;

– регистрацию действий субъекта и его процесса;

– предоставление возможностей исключения и включения новых субъектов и объектов доступа, а также изменение полномочий субъектов;

– реакцию на попытки НСД, например, сигнализацию, блокировку, восстановление после НСД;

– тестирование;

– очистку оперативной памяти и рабочих областей на магнитных носителях после завершения работы пользователя с защищаемыми данными;

– учет выходных печатных и графических форм и твердых копий в АС;

– контроль целостности программной и информационной части как СРД, так и обеспечивающих ее средств.

Функции системы разграничения доступа и обеспечивающих средств, предлагаемые в концепции, близки к аналогичным положениям «Оранжевой книги». Это вполне естественно, поскольку близки и исходные посылки – защита от несанкционированного доступа к информации в условиях физически безопасного окружения.

Технические средства защиты от НСД, согласно концепции, должны оцениваться по следующим основным параметрам:

– степень полноты охвата ПРД реализованной СРД и ее качество;
– состав и качество обеспечивающих средств для СРД;
– гарантии правильности функционирования СРД и ее средств.

Классификация СВТ по уровню защищенности от НСД

Предлагаемая Гостехкомиссией при Президенте РФ классификация средств вычислительной техники по уровню защищенности от НСД к информации, близка к «Оранжевой книге» и устанавливает семь классов защищенности СВТ от НСД к информации. Самый низкий класс – седьмой, самый высокий – первый.

Классы подразделяются на четыре группы, отличающиеся качественным уровнем защиты:

• первая группа содержит только один седьмой класс;

• вторая группа характеризуется дискреционной защитой и содержит шестой и пятый классы;

• третья группа характеризуется мандатной защитой и содержит четвертый, третий и второй классы;

• четвертая группа характеризуется верифицированной защитой и содержит только первый класс.

Седьмой класс присваивают СВТ, к которым предъявлялись требования по защите от НСД к информации, но при оценке защищенность СВТ оказалась ниже уровня требований шестого класса. Распределение показателей защищенности по классам СВТ приведено в табл. 4.1.

4.1. Распределение показателей защищенности по классам СВТ

Наименование показателя	Класс защищенности

	6	5	4	3	2	1
1. Дискреционный принцип контроля доступа	+	+	+	=	+	=
2. Мандатный принцип контроля доступа	–	–	+	=	=	=
3. Очистка памяти	–	+	+	+	=	=
4. Изоляция модулей	–	–	+	=	+	=
5. Маркировка документов	–	–	+	=	=	=
6. Защита ввода/вывода на отчуждаемый физический носитель информации	–	–	+	=	=	=
7. Сопоставление пользователя с устройством	–	–	+	=	=	=
8. Идентификация и аутентификация	+	=	+	=	=	=
9. Гарантия проектирования	–	+	+	+	+	+
10. Регистрация	–	+	+	+	=	=
11. Взаимодействие пользователя с КСЗ	–	–	–	+	=	=
12. Надежно восстановление	–	–	–	+	=	=
13. Целостность КСЗ	–	+	+	+	=	=
14. Контроль модификации	–	–	–	–	+	=
15. Контроль дистрибуции	–	–	–	–	+	=
16. Гарантии архитектуры	–	–	–	–	–	+
17. Тестирование	+	+	+	+	+	=
18. Руководство пользователя	+	=	=	=	=	=
19. Руководство по КСЗ	+	+	=	+	+	=
20. Текстовая документация	+	+	+	+	+	=
21. Конструкторская (проектная) документация	+	+	+	+	+	+

Обозначения: « – » – нет требования к данному классу; « + » – новые или дополнительные требования; « = » – требования совпадают с требованиями к СВТ; КСЗ – комплекс средств защиты.

Классификация АС по уровню защищенности от НСД

Классификация автоматизированных систем устроена иначе. Обратимся к соответствующему Руководящему документу.

Устанавливается девять классов защищенности АС от НСД к информации.

Каждый класс характеризуется определенной минимальной совокупностью требований по защите.

Классы подразделяются на три группы, отличающиеся особенностями обработки информации в АС.

В пределах каждой группы соблюдается иерархия требований по защите в зависимости от ценности (конфиденциальности) информации и, следовательно, иерархия классов защищенности АС.

Третья группа классифицирует АС, в которых работает один пользователь, допущенный ко всей информации АС, размещенной на носителях одного уровня конфиденциальности. Группа содержит два класса – 3Б и 3А.

Вторая группа классифицирует АС, в которых пользователи имеют одинаковые права доступа (полномочия) ко всей информации АС, обрабатываемой и (или) хранимой на носителях различного уровня конфиденциальности. Группа содержит два класса – 2Б и 2А.

Первая группа классифицирует многопользовательские АС, в которых одновременно обрабатывается и (или) хранится информация разных уровней конфиденциальности и не все пользователи имеют право доступа ко всей информации АС. Группа содержит пять классов – 1Д, 1Г, 1В, 1Б и 1А.

Требования к классам защищенности автоматизированных систем приведены в табл. 4.2.

«Оранжевая книга» Министерства обороны США и Руководящие документы Гостехкомиссии при Президенте РФ создавались в расчете на централизованные конфигурации, основу которых составляют большие машины. Распределенная организация современных информационных систем требует внесения существенных изменений и дополнений как в политику безопасности, так и в способы проведения ее в жизнь. Появились новые угрозы, для противодействия которым нужны новые функции и механизмы защиты. Основополагающим документом в области защиты распределенных систем стали рекомендации X.800.

4.2. 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 в/из системы (узла сети);			+	+	+	+	+	+	+
выдача печатных (графических) выходных документов;				+		+	+	+	+
запуска/завершения программ и процессов (заданий, задач);				+		+	+	+	+
доступа программ субъектов к защищаемым файлам, включая их создание и удаление, передачу по линиям и каналам связи;				+		+	+	+	+
доступа программ субъектов, доступа к терминалам, ЭВМ, узлам сети ЭВМ, 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				+		+	+	+	+

Продолжение табл. 4.2

Подсистемы и требования	Классы								
	3Б	3А	2Б	2А	1Д	1Г	1В	1Б	1А
2.4. Сигнализация попыток нарушения защиты							+	+	+
3. Криптографическая защита									
3.1. Шифрование конфиденциальной информации				+				+	+
3.2. Шифрование информации, принадлежащей различным субъектам доступа (группам доступа) 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 охраны и носителей ин-			+	+	+	+	+	+	+

формации									
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				+			+	+	+

Обозначение: « + » – есть требования к данному классу; СЗИ НСД – система защиты информации от несанкционированного доступа.

Межсетевые экраны. Показатели защищенности от НСД к информации

При анализе системы защиты внешнего периметра корпоративной сети в качестве основных критериев целесообразно использовать РД «СВТ. Межсетевые экраны. Защита от НСД к информации. Показатели защищенности от НСД к информации».

Данный документ определяет показатели защищенности межсетевых экранов (МЭ). Каждый показатель защищенности представляет собой набор требований безопасности, характеризующих определенную область функционирования МЭ.

Всего выделяется пять показателей защищенности:

1) управление доступом;
2) идентификация и аутентификация;
3) регистрация событий и оповещение;
4) контроль целостности;
5) восстановление работоспособности.

На основании показателей защищенности определяются следующие пять классов защищенности МЭ:

1) простейшие фильтрующие маршрутизаторы – 5 класс;
2) пакетные фильтры сетевого уровня – 4 класс;
3) простейшие МЭ прикладного уровня – 3 класс;
4) МЭ базового уровня – 2 класс;
5) продвинутые МЭ – 1 класс.

Защита от НСД к информации. Программное обеспечение средств защиты информации. Классификация по уровню контроля отсутствия недекларированных возможностей

Настоящий руководящий документ (РД) устанавливает классификацию программного обеспечения (ПО) (как отечественного, так и импортного производства) средств защиты информации (СЗИ), в том числе и встроенных в общесистемное и прикладное ПО, по уровню контроля отсутствия в нем недекларированных возможностей.

Действие документа не распространяется на программное обеспечение средств криптографической защиты информации.

Уровень контроля определяется выполнением заданного настоящим РД набора требований, предъявляемого:

– к составу и содержанию документации, представляемой заявителем для проведения испытаний ПО СЗИ;

– к содержанию испытаний.

4.2.4. ОСОБЕННОСТИ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ КОМПЬЮТЕРНЫХ СЕТЕЙ. РЕКОМЕНДАЦИИ X.800

Рекомендации X.800 – документ довольно обширный. Рассмотрим специфические сетевые функции (сервисы) безопасности, а также необходимые для их реализации защитные механизмы.

Чтобы почувствовать специфику распределенных систем, достаточно рассмотреть такое стандартное средство защиты, как подотчетность. Помимо других целей, записи в регистрационном журнале могут служить доказательством того, что определенный пользователь совершил то или иное действие (точнее, действие было совершено от его имени). В результате пользователь не может отказаться от содеянного и в некоторых случаях несет за это наказание. В распределенных системах действие порой совершается на нескольких компьютерах и, вообще говоря, не исключено, что их регистрационные журналы противоречат друг другу. Так бывает, когда злоумышленнику удается подделать сете-

вой адрес и имя другого пользователя. Значит, нужны иные средства обеспечения «неотказуемости» (невозможности отказаться) от совершенных действий.

Перечислим функции (сервисы) безопасности, характерные для распределенных систем:

Аутентификация – обеспечивает аутентификацию партнеров по общению и аутентификацию источника данных.

Аутентификация партнеров по общению используется при установлении соединения и, быть может, периодически во время сеанса. Она служит для предотвращения таких угроз, как маскарад и повтор предыдущего сеанса связи.

Аутентификация источника данных – это подтверждение подлинности источника отдельной порции данных. Функция не обеспечивает защиты против повторной передачи данных.

Управление доступом – обеспечивает защиту от несанкционированного использования ресурсов, доступных по сети.

Конфиденциальность данных – обеспечивает защиту от несанкционированного получения информации. Различают следующие виды конфиденциальности:

- конфиденциальность данных с установлением соединения;
- конфиденциальность данных без установления соединения;
- конфиденциальность отдельных полей данных (избирательная конфиденциальность);
- конфиденциальность трафика (защита информации, которую можно получить, анализируя трафик).

Целостность данных – подразделяется на подвиды в зависимости от того, какой тип общения используют партнеры – с установлением соединения или без такового, защищаются ли все данные или только отдельные поля, обеспечивается ли восстановление в случае нарушения целостности.

Неотказуемость – это функция, обеспечивающая невозможность отказаться от совершенных действий обеспечивает два вида услуг:

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

Механизмы безопасности. Для реализации функций безопасности могут использоваться следующие механизмы и их комбинации.

Шифрование – подразделяется на симметричное и асимметричное.

Различают также обратимое и необратимое шифрование. Последнее может использоваться для вычисления криптографических контрольных сумм (хэш-функций, дайджестов, имитовставок).

Электронная (*цифровая*) *подпись* – включает в себя две процедуры:

- выработку подписи;
- проверку подписанной порции данных.

Процедура выработки подписи использует информацию, известную только подписывающему порцию данных. Процедура проверки подписи является общедоступной, она не должна позволять найти секретный ключ подписывающего.

Механизмы управления доступом. При принятии решений по поводу предоставления запрашиваемого типа доступа могут использоваться следующие виды и источники информации:

- базы данных управления доступом (в такой базе, поддерживаемой централизованно или на оконечных системах, могут храниться списки управления доступом или структуры аналогичного назначения);
- пароли или иная аутентификационная информация;
- токены, билеты или иные удостоверения, предъявление которых свидетельствует о наличии прав доступа;
- метки безопасности, ассоциированные с субъектами и объектами доступа;
- время, маршрут и длительность запрашиваемого доступа.

Механизмы управления доступом могут располагаться на любой из общающихся сторон или в промежуточной точке. В промежуточных точках целесообразно проверять права доступа к коммуникационным ресурсам. Очевидно, требования механизма, расположенного на приемном конце, должны быть известны заранее, до начала общения.

Механизмы контроля целостности данных. Различают два аспекта целостности: целостность отдельного сообщения или поля информации и целостность потока сообщений или полей информации. Вообще говоря, контроль двух видов целостности осуществляет-

ся различными механизмами, хотя контролировать целостность потока, не проверяя отдельные сообщения, едва ли имеет смысл.

Процедура контроля целостности отдельного сообщения (поля) включает в себя два процесса – один на передающей стороне, другой на приемной. На передающей стороне к сообщению добавляется избыточная информация, которая является функцией от сообщения (разновидности контрольных сумм). На приемной стороне независимо генерируется контрольная сумма полученного сообщения с последующим сравнением результатов. Данный механизм сам по себе не защищает от дублирования сообщений.

Для проверки целостности потока сообщений (защита от кражи, переупорядочивания, дублирования и вставки сообщений) используются порядковые номера, временные штампы, криптографическое связывание (результат шифрования очередного сообщения зависит от предыдущего) или иные приемы.

Механизмы аутентификации. Аутентификация может достигаться за счет использования паролей, личных карточек или иных устройств аналогичного назначения, криптографических методов (когда демонстрируется знание секретного ключа), устройств измерения и анализа биометрических характеристик.

Аутентификация бывает односторонней (клиент доказывает свою подлинность серверу) и двусторонней (взаимной). Пример односторонней аутентификации – процедура входа пользователя в систему.

Для защиты от дублирования аутентификационной информации могут использоваться временные штампы и синхронизация часов в узлах сети.

Механизмы дополнения трафика, разумеется, эффективны только в сочетании со средствами обеспечения конфиденциальности, поскольку в противном случае злоумышленнику будет очевиден фиктивный характер дополнительных сообщений.

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 – служит для заверения таких коммуникационных характеристик, как целостность, время, личности отправителя и получателей. Заверение обеспечивается надежной третьей стороной, которая обладает достаточной информацией, чтобы ее заверениям можно было доверять. Обычно нотаризация опирается на механизм электронной подписи.

Администрирование средств безопасности – безопасности включает в себя распространение информации, необходимой для работы функций и механизмов безопасности, а также сбор и анализ информации об их функционировании. Примерами могут служить распространение криптографических ключей, установка значений параметров защиты, ведение регистрационного журнала и т.п.

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 проведения в жизнь избранной политики безопасности.

Усилия администратора средств безопасности должны распределяться по трем направлениям: администрирование системы в целом; администрирование функций безопасности; администрирование механизмов безопасности.

Среди действий, относящихся к системе в целом, отметим поддержание актуальности политики безопасности, взаимодействие с другими административными службами, реагирование на происходящие события, аудит и безопасное восстановление.

Администрирование функций безопасности включает в себя определение защищаемых объектов, выработку правил подбора механизмов безопасности (при наличии альтернатив), комбинирование механизмов для реализации функции безопасности, взаимодействие с другими администраторами для обеспечения согласованной работы.

Обязанности администратора механизмов безопасности определяются перечнем задействованных механизмов. Типичный список таков:

- управление ключами (генерация и распределение). Многие аспекты управления ключами (например, их доставка) выходят за пределы среды OSI;
- управление шифрованием (установка и синхронизация криптографических параметров). К управлению шифрованием можно отнести и администрирование механизмов электронной подписи и управление целостностью, если оно обеспечивается криптографическими средствами;

• 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 дополнением трафика (выработка и поддержание правил, задающих характеристики дополняющих сообщений – частоту отправки, размер и т.п.). Характеристики могут варьироваться по заданному закону в зависимости от даты и времени;

• управление маршрутизацией (выделение надежных путей);

• управление нотаризацией (распространение информации о нотариальных службах, администрирование этих служб).

Итак, администрирование средств безопасности в распределенной среде имеет много особенностей по сравнению с централизованными системами.

В табл. 4.3. указаны уровни эталонной семиуровневой модели OSI, на которых могут быть реализованы функции безопасности.

4.3. Распределение функций безопасности по уровням эталонной семиуровневой модели ISO

Функция 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							+

Обозначения: «+» – данный уровень может представить функцию безопасности.

Контрольные вопросы

1. Перечислите уровни информационной безопасности.
2. Охарактеризуйте целевую направленность Общих Критериев. Требования и концепции Общих Критериев.
3. Назовите действующие стандарты и рекомендации в области информационной безопасности.
4. Дайте общую характеристику стандартам и рекомендациям в области информационной безопасности.
5. Охарактеризуйте документы Гостехкомиссии РФ.

5. Административный уровень информационной безопасности в информационно-вычислительной системе

Информация в системе, поддержанная информационной технологией, является критическим ресурсом, который позволяет использующим его организациям выполнять свои функции. При этом система будет выполнять эти функции эффективно только при осуществлении надлежащего контроля за информацией, чтобы гарантировать, что она защищена от опасностей типа нежелательного или несанкционированного распростране-

ния, изменения или потери. Безопасность ИТ предназначена, чтобы предотвратить или уменьшить эти и подобные опасности.

Анализ возможных угроз и анализ рисков помогает выбору мер безопасности, которые должны быть осуществлены, чтобы уменьшить риск до приемлемого уровня. Эти меры безопасности можно обеспечить через соответствующие комбинации ИТ, реализующих функции системы, и/или через внешние меры.

Понятие «защищенности» принципиально не отличается от любых других свойств технической системы и является для системы априорно заданным. Особенностью понятия «защищенность» является его тесная связь с понятиями «злоумышленник» или «угрозы» (понятие, обезличивающее причину вывода системы из защищенного состояния злоумышленником).

При рассмотрении понятия «злоумышленник» практически всегда выделяется объект его воздействия – часть системы, на которую направлены те или иные его действия.

Обычно выделяют три компоненты, связанные с нарушением безопасности системы:

- *злоумышленник* – внешний по отношению к системе источник нарушения свойства безопасности;
- *объект атаки* – часть, принадлежащая системе, на которую направлены те или иные воздействия «злоумышленника»;
- *канал воздействия* – среда переноса злоумышленного воздействия.

5.1. ПОНЯТИЕ ПОЛИТИКИ БЕЗОПАСНОСТИ

Интегральной характеристикой защищаемой системы является политика безопасности – качественное (или количественно-качественное) выражение свойств защищенности в терминах, представляющих систему.

Наиболее часто рассматриваются политики безопасности, связанные с понятием «доступ». *Доступ* – категория субъективно-объективной политики, описывающая процесс выполнения операций субъектов над объектами.

Политика безопасности включает:

- множество операций субъектов над объектами;
- для каждой пары «субъект – объект» (S_i,O_i) множество разрешенных операций, из множества возможных операций.

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. Под эту программу выделяются ресурсы, назначаются ответственные, определяется порядок контроля выполнения программы и т.п.

5.1.1. АНАЛИЗ РИСКА

В настоящее время инвестиции в информационную безопасность могут рассматриваться как инвестиции для увеличения прибыли путем уменьшения административных затрат на ее поддержание или для защиты от потери прибыли путем предотвращения потенциальных затрат в случае негативных последствий. При этом стоимость средств обеспечения безопасности должна соответствовать риску и прибыли для той среды, в которой работает организация.

Риск – это ситуация, когда угроза использует уязвимое место для нанесения вреда вашей системе. Политика безопасности обеспечивает основу для внедрения средств обеспечения безопасности путем уменьшения числа уязвимых мест и как следствие уменьшает риск. Для того чтобы разработать эффективную и недорогую политику безопасности для защиты соединений с глобальными сетями, нужно выполнить тот или иной анализ риска для оценки требуемой жесткости политики, который определит необходимые затраты на средства обеспечения безопасности для выполнения требований политики. То, насколько жесткой будет политика, зависит от:

- уровня угроз, которым подвергается организация и видимость организации из внешнего мира;
- уязвимости организации к последствиям потенциальных инцидентов с безопасностью;
- государственных законов и требований вышестоящих организаций, которые могут явно определять необходимость проведения того или иного вида анализа риска или диктовать применение конкретных средств обеспечения безопасности для конкретных систем, приложений или видов информации.

Отметим, что здесь не учитывается ценность информации или последствия инцидентов с безопасностью. В прошлом такие оценки стоимости требовались как составная часть формального анализа риска в попытке осуществить оценку затрат на безопасность.

По мере того, как зависимость государственных и коммерческих организаций от глобальных сетей становилась большей, потери от инцидентов с безопасностью, которые практически невозможно оценить в деньгах, стали равными или большими, чем вычисляемые затраты. Время администраторов информационной безопасности может более эффективно потрачено на обеспечение гарантий внедрения «достаточно хорошей безопасности», чем на расчет стоимости чего-либо худшего, чем полная безопасность.

Для организаций, деятельность которых регулируется законами, или которые обрабатывают информацию, от которой зависит жизнь людей, могут оказаться более приемлемыми формальные методы оценки риска.

5.1.2. УГРОЗЫ. ВИДИМОСТЬ

Угроза – это любое событие, которое потенциально может нанести вред организации путем раскрытия, модификации или разрушения информации, или отказа в обслуживании критическими сервисами. Угрозы могут быть неумышленными, такими как те, что вызываются ошибками человека, сбоями оборудования или программ, или стихийными бедствиями. Умышленные угрозы могут быть разделены на ряд групп – от логичных (получение бесплатных материальных благ) до иррациональных (разрушение информации). Типичными угрозами в глобальных сетях являются:

- Сбой в работе одной из компонент сети – сбой из-за ошибок при проектировании или ошибок оборудования или программ может привести к отказу в обслуживании или компрометации безопасности из-за неправильного функционирования одной из компонент сети. Выход из строя брандмауэра или ложные отказы в авторизации серверами аутентификации являются примерами сбоев, которые оказывают влияние на безопасность.
- Сканирование информации – неавторизованный просмотр критической информации злоумышленниками или авторизованными пользователями может происходить, используя различные механизмы – электронное письмо с неверным адресатом, распечатка принтера, неправильно сконфигурированные списки управления доступом, совместное использование несколькими людьми одного идентификатора и т.д.
- Использование информации не по назначению – использование информации для целей, отличных от авторизованных, может привести к отказу в обслуживании, излишним затратам, потере репутации. Виновниками этого могут быть как внутренние, так и внешние пользователи.
- Неавторизованное удаление, модификация или раскрытие информации – специальное искажение информационных ценностей, которое может привести к потере целостности или конфиденциальности информации.
- Проникновение – атака неавторизованных людей или систем, которая может привести к отказу в обслуживании или значительным затратам на восстановление после инцидента.
- Маскарад – попытки замаскироваться под авторизованного пользователя для кражи сервисов или информации, или для инициации финансовых транзакций, которые приведут к финансовым потерям или проблемам для организации.

Наличие угрозы необязательно означает, что она нанесет вред. Чтобы стать риском, угроза должна использовать уязвимое место в средствах обеспечения безопасности системы и система должна быть видима из внешнего мира.

Видимость системы – это мера как интереса злоумышленников к этой системе, так и количества информации, доступной для общего пользования на этой системе.

Так как многие угрозы, основанные на глобальных сетях, являются вероятностными по своей природе, уровень видимости организации напрямую определяет вероятность того, что враждебные агенты будут пытаться нанести вред с помощью той или иной угрозы. В Интернете любопытные студенты, подростки-вандалы, криминальные элементы, промышленные шпионы могут являться носителями угрозы. По мере того как использование глобальных сетей для электронной коммерции и критических задач увеличивается, число атак криминальных элементов и шпионов будет увеличиваться.

5.1.3. УЯЗВИМОСТЬ. ПОСЛЕДСТВИЯ

Организации по-разному уязвимы к риску. Политики безопасности должны отражать уязвимость конкретной организации к различным типам инцидентов с безопасностью и делать приоритетными инвестиции в области наибольшей уязвимости.

Имеется два фактора, определяющих уязвимость организации. Первый фактор – последствия инцидента с безопасностью. Почти все организации уязвимы к финансовым потерям – устранение последствий инцидентов с безопасностью может потребовать значительных вложений, даже если пострадали некритические сервисы.

Одним из важных шагов при определении возможных последствий является ведение реестра информационных ценностей. Хотя это и кажется простым, поддержание точного списка систем, сетей, компьютеров и баз данных, использующихся в организации, является сложной задачей. Организации должны объединить этот список с результатами ра-

бот по классификации данных, в ходе которых информация классифицируется по степени важности для выполнения организацией своих задач.

Более серьезные последствия возникают, когда нарушается внутренняя работа организации, что приводит к убыткам из-за упущенных возможностей, потерь рабочего времени и работ по восстановлению работы. Самые серьезные последствия – это невозможность системы выполнять свои внешние функции. Последствия инцидента с безопасностью напрямую вызывают нарушения работы служб.

Второй фактор – это учет политических или организационных последствий.

Эти факторы надо учитывать при определении уязвимости организации к инцидентам с безопасностью (табл. 5.1.).

5.1. Матрица профиля риска

Угрозы	Рейтинг	Видимость	Рейтинг	Число очков
Ни одна из угроз не считается реальной	1	Очень маленькая	1	
Возможность возникновения угроз тяжело оценить	3	Средняя, периодические публикации об организации	3	
Угрозы реальны, имел место ряд случаев их возникновения	5	Большая, постоянные публикации об организации	5	
Последствия	Рейтинг	Уязвимость	Рейтинг	Число очков
Финансовых потерь не будет, возможные последствия учтены в бюджете или предприняты меры по переносу риска	1	Инциденты считаются приемлемыми; руководство организации с пониманием относится к этому	1	

Продолжение табл. 5.1

Угрозы	Рейтинг	Видимость	Рейтинг	Число очков
Будут затронуты внутренние функции организации, превышен бюджет, потеряны возможности	3	Инцидент повлияет на позицию среднего звена управления, изменится к безопасности	3	
Будут затронуты внешние функции организации, нанесен большой финансовый ущерб	5	Руководители организации станут жестче относиться к безопасности	5	
Общее число баллов:				

Рейтинг: Значение для угроз умножается на значение для видимости, а значение для последствий умножается на значение для уязвимости. Затем эти два числа складываются:

2 – 10: низкий риск; 11 – 29: средний риск; 30 – 50: высокий риск

5.1.4. УЧЕТ ИНФОРМАЦИОННЫХ ЦЕННОСТЕЙ

Чтобы гарантировать защиту всех информационных ценностей, и то, что текущая вычислительная среда организации может быть быстро восстановлена после инцидента с безопасностью, каждый сетевой администратор должен вести учет информационных систем в его зоне ответственности. Список должен включать в себя все существующую аппаратную часть вычислительной среды, программы, электронные документы, базы данных и каналы связи.

Для каждой информационной ценности должна быть описана следующая информация:

- тип: оборудование, программа, данные;
- использование в системе общего назначения или критическом приложении;
- ответственный за данную информационную ценность;
- ее физическое или логическое местоположение;
- учетный номер, где это возможно.

Для того чтобы разработать эффективную политику безопасности, информация, хранимая или обрабатываемая в организации, должна быть классифицирована в соответ-

ствии с ее критичностью к потере конфиденциальности, которая была рассмотрена ранее. На основе этой классификации потом можно легко разработать политику для разрешения (или запрещения) доступа к глобальным сетям.

5.2. МОДЕЛИ ОСНОВНЫХ ТИПОВ ПОЛИТИК БЕЗОПАСНОСТИ

5.2.1. ТИПЫ ПОЛИТИК БЕЗОПАСНОСТИ

Существуют следующие типы политик безопасности: *дискреционная, мандатная и ролевая.*

Основой *дискреционной* (*дискретной*) *политики безопасности* является дискреционное управление доступом (*Discretionary Access Control – DAC*), которое определяется двумя свойствами:

1) все субъекты и объекты должны быть идентифицированы;

2) права доступа субъекта к объекту системы определяются на основании некоторого внешнего по отношению к системе правила.

К достоинствам дискреционной политики безопасности можно отнести относительно простую реализацию соответствующих механизмов защиты. Этим обусловлен тот факт, что большинство распространенных в настоящее время автоматизированных систем обеспечивают выполнение положений именно данной политики безопасности.

В качестве примера реализаций дискреционной политики безопасности в АС можно привести матрицу доступов, строки которой соответствуют субъектам системы, а столбцы – объектам; элементы матрицы характеризуют права доступа. К недостаткам относится статичность модели. Это означает, что данная политика безопасности не учитывает динамику изменений состояния АС, не накладывает ограничений на состояния системы.

Кроме этого, при использовании дискреционной политики безопасности возникает вопрос определения правил распространения прав доступа и анализа их влияния на безопасность АС. В общем случае при использовании данной политики безопасности перед монитором безопасности объектов (МБО), который при санкционировании доступа субъекта к объекту руководствуется некоторым набором правил, стоит алгоритмически неразрешимая задача: проверить − приведут ли его действия к нарушению безопасности или нет.

В то же время имеются модели АС, реализующих дискреционную политику безопасности (например, модель Take – Grant), которые предоставляют алгоритмы проверки безопасности.

Матрица доступов не является тем механизмом, который бы позволил реализовать ясную и четкую систему защиты информации в АС. Этим обуславливается поиск других более совершенных политик безопасности.

Основу *мандатной* (*полномочной*) *политики безопасности* составляет мандатное управление доступом (*Mandatory Access Control – MAC*), которое подразумевает, что:

- все субъекты и объекты системы должны быть однозначно идентифицированы;
- задан линейно упорядоченный набор меток секретности;
- каждому объекту системы присвоена метка секретности, определяющая ценность содержащейся в нем информации – его уровень секретности в АС;
- каждому субъекту системы присвоена метка секретности, определяющая уровень доверия к нему в АС – максимальное значение метки секретности объектов, к которым субъект имеет доступ; метка секретности субъекта называется его уровнем доступа.

Основная цель мандатной политики безопасности – предотвращение утечки информации от объектов с высоким уровнем доступа к объектам с низким уровнем доступа, т.е. противодействие возникновению в АС информационных каналов сверху вниз.

Чаще всего мандатную политику безопасности описывают в терминах, понятиях и определениях свойств модели Белла-Лападула, которая будет рассмотрена позже. В рамках данной модели доказывается важное утверждение, указывающее на принципиальное отличие систем, реализующих мандатную защиту, от систем с дискреционной защитой: *если начальное состояние системы безопасно, и все переходы системы из состояния в состояние не нарушают ограничений, сформулированных политикой безопасности, то любое состояние системы безопасно.*

Кроме того, по сравнению с АС, построенными на основе дискреционной политики безопасности, для систем, реализующих мандатную политику, характерна более высокая степень надежности. Это связано с тем, что МБО такой системы должен отслеживать не только правила доступа субъектов системы к объектам, но и состояния самой АС. Таким образом, каналы утечки в системах данного типа не заложены в нее непосредственно (что мы наблюдаем в положениях предыдущей политики безопасности), а могут появиться только при практической реализации системы.

При большом количестве пользователей традиционные подсистемы управления доступом становятся крайне сложными для администрирования. Число связей в них пропор-

ционально произведению количества пользователей на количество объектов. Необходимы решения в объектно-ориентированном стиле, способные эту сложность понизить.

В 2001 г. Национальный институт стандартов и технологий США предложил проект стандарта *ролевого управления доступом*.

Ролевое разграничение доступа (РРД) представляет собой развитие политики дискреционного разграничения доступа; при этом права доступа субъектов системы на объекты группируются с учетом специфики их применения, образуя роли. РРД является составляющей многих современных систем и применяется в системах защиты СУБД, сетевых ОС.

Задание ролей позволяет определить более четкие и понятные для пользователей системы правила разграничения доступа, соответствующих их должностным полномочиям и обязанностям.

Роль является совокупностью прав доступа на объекты системы. Вместе с тем РРД не является частным случаем дискреционного разграничения доступа, так как правила РРД определяют порядок предоставления прав доступа субъектам системы в зависимости от сессии его работы и от имеющихся или отсутствующих у него ролей в каждый момент времени, что является характерным для систем мандатного разграничения доступа. В то же время правила РРД являются более гибкими, чем правила мандатного разграничения доступа, построенные на основе жестко определенной решетки (шкалы) ценности информации.

Суть ролевого разграничения доступа состоит в том, что между пользователями и их привилегиями появляются промежуточные сущности – роли. Для каждого пользователя одновременно могут быть активными несколько ролей, каждая из которых дает ему определенные права (рис. 5.1).

Ролевой доступ нейтрален по отношению к конкретным видам прав и способам их проверки; его можно рассматривать как объектно-ориентированный каркас, облегчающий администрирование, поскольку он позволяет сделать подсистему разграничения доступа управляемой при сколь угодно большом числе пользователей, прежде всего за счет установления между ролями связей, аналогичных наследованию в объектно-ориентированных системах. Кроме того, ролей должно быть значительно меньше, чем пользователей. В результате число администрируемых связей становится пропорциональным сумме (а не произведению) количества пользователей и объектов, что по порядку величины уменьшить уже невозможно.

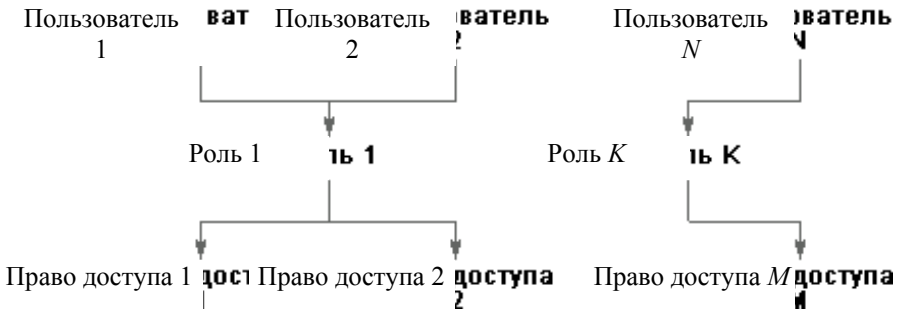


Рис. 5.1. Пользователи, объекты и роли

Ролевое управление доступом оперирует следующими основными понятиями:

– пользователь (человек, интеллектуальный автономный агент и т.п.);

– сеанс работы пользователя;

– роль (определяется в соответствии с организационной структурой);

– объект (сущность, доступ к которой разграничивается; например, файл ОС или таблица СУБД);

– операция (зависит от объекта; для файлов ОС – чтение, запись, выполнение и т.п.; для таблиц СУБД – вставка, удаление и т.п., для прикладных объектов операции могут быть более сложными);

– право доступа (разрешение выполнять определенные операции над определенными объектами).

Ролям приписываются пользователи и права доступа; можно считать, что они (роли) именуют отношения "многие ко многим" между пользователями и правами. Роли могут быть приписаны многие пользователи; один пользователь может быть приписан нескольким ролям. Во время сеанса работы пользователя активизируется подмножество ролей, которым он приписан, в результате чего он становится обладателем объединения прав, приписанных активным ролям. Одновременно пользователь может открыть несколько сеансов.

Между ролями (r) может быть определено отношение частичного порядка, называемое наследованием. Если роль $r2$ является наследницей $r1$, то все права $r1$ приписывают-

ся *r*2, а все пользователи *r*2 приписываются *r*1. Очевидно, что наследование ролей соответствует наследованию классов в объектно-ориентированном программировании, только правам доступа соответствуют методы классов, а пользователям – объекты (экземпляры) классов.

Отношение наследования является иерархическим, причем права доступа и пользователи распространяются по уровням иерархии навстречу друг другу. В общем случае наследование является множественным, т.е. у одной роли может быть несколько предшественниц (и, естественно, несколько наследниц, которых мы будем называть также преемницами).

Можно представить себе формирование иерархии ролей, начиная с минимума прав (и максимума пользователей), приписываемых роли «сотрудник», с постепенным уточнением состава пользователей и добавлением прав (роли «системный администратор», «бухгалтер» и т.п.), вплоть до роли «руководитель» (что, впрочем, не значит, что руководителю предоставляются неограниченные права; как и другим ролям, в соответствии с принципом минимизации привилегий, этой роли целесообразно разрешить только то, что необходимо для выполнения служебных обязанностей). Фрагмент подобной иерархии ролей показан на рис. 5.2.

Рис. 5.2. Фрагмент иерархии ролей

Для реализации еще одного упоминавшегося ранее важного принципа информационной безопасности вводится понятие разделения обязанностей, причем в двух видах: статическом и динамическом.

Статическое разделение обязанностей налагает ограничения на приписывание пользователей ролям. В простейшем случае членство в некоторой роли запрещает приписывание пользователя определенному множеству других ролей. В общем случае данное ограничение задается как пара «множество ролей – число» (где множество состоит, по крайней мере, из двух ролей, а число должно быть больше единицы), так что никакой пользователь не может быть приписан указанному (или большему) числу ролей из заданного множества. Например, может существовать пять бухгалтерских ролей, но политика безопасности допускает членство не более чем в двух таких ролях (здесь число равно трем).

При наличии наследования ролей ограничение приобретает несколько более сложный вид, но суть остается простой: при проверке членства в ролях нужно учитывать приписывание пользователей ролям-наследницам.

Динамическое разделение обязанностей отличается от статического только тем, что рассматриваются роли, одновременно активные (быть может, в разных сеансах) для данного пользователя (а не те, которым пользователь статически приписан). Например, один пользователь может играть роль и кассира, и контролера, но не одновременно; чтобы стать контролером, он должен сначала закрыть кассу. Тем самым реализуется так называемое временное ограничение доверия, являющееся аспектом минимизации привилегий.

Рассматриваемый проект стандарта содержит спецификации трех категорий функций, необходимых для администрирования РРД:

Административные функции (создание и сопровождение ролей и других атрибутов ролевого доступа): создать/удалить роль/пользователя, приписать пользователя/право роли или ликвидировать существующую ассоциацию, создать/удалить отношение наследования между существующими ролями, создать новую роль и сделать ее наследницей/предшественницей существующей роли, создать/удалить ограничения для статического/динамического разделения обязанностей.

Вспомогательные функции (обслуживание сеансов работы пользователей): открыть сеанс работы пользователя с активацией подразумеваемого набора ролей; активировать новую роль, деактивировать роль; проверить правомерность доступа.

Информационные функции (получение сведений о текущей конфигурации с учетом отношения наследования). Здесь проводится разделение на обязательные и необязательные функции. К числу первых принадлежат получение списка пользователей, приписанных роли, и списка ролей, которым приписан пользователь.

Для анализа и изучения свойств систем РРД используются математические модели, в основе которых лежит базовая модель.

5.2.2. МОДЕЛЬ МАТРИЦЫ ДОСТУПОВ ХАРРИСОНА–РУЗЗО–УЛЬМАНА

Модель Харрисона–Руззо–Ульмана (*HRU*) разработана и впервые предложена в 1971 г., а в 1976 г. появилось ее формальное описание. Она используется для анализа системы защиты, реализующей дискреционную политику безопасности, и ее основного элемента – матрицы доступов. При этом система защиты представляется конечным автоматом, функционирующим согласно определенным правилам перехода.

Обозначим: O – множество объектов системы; S – множество субъектов системы ($S \subseteq O$); R – множество прав доступа субъектов к объектам, например права на чтение (*read*), на запись (*write*), владения (*own*); M – матрица доступа, строки которой соответствуют субъектам, а столбцы – объектам; $M[s, o] \subseteq R$ – права доступа субъекта s к объекту о.

Отдельный автомат, построенный согласно положениям модели *HRU*, будем называть системой. Функционирование системы рассматривается только с точки зрения изменений в матрице доступа. Возможные изменения определяются шестью примитивными операторами:

«Внести» право $r \in R$ в $M[s, o]$ – добавление субъекту s права доступа r к объекту o. При этом в ячейку $M[s, o]$ матрицы доступов добавляется элемент r.

«Удалить» право $r \in R$ из $M[s, o]$ – удаление у субъекта s права доступа r к объекту o. При этом из ячейки $M[s, o]$ матрицы доступов удаляется элемент r.

«Создать» субъект s' – добавление в систему нового субъекта s'. При этом в матрицу доступов добавляются новые столбец и строка.

«Создать» объект o' – добавление в систему нового объекта o'. При этом в матрицу доступов добавляется новый столбец.

«Уничтожить» субъект s' – удаление из системы субъекта s'. При этом из матрицы доступов удаляются соответствующие столбец и строка.

«Уничтожить» объект o' – удаление из системы объекта o'. При этом из матрицы доступов удаляется соответствующий столбец.

В результате выполнения примитивного оператора a осуществляется переход системы из состояния $Q = (S, O, M)$ в новое состояние $Q' = (S', O', M')$ (табл. 5.2). Данный переход обозначим через $Q \vdash \alpha\, Q'$.

5.2. Таблица переходов из состояния в состояние модели *HRU*

Примитивный оператор модели *HRU*	Условия выполнения	Новое состояние системы
«Внести» право $r \in R$ в $M[s, o]$	$s \in S, o \in O$	$S' = S, O' = O, M'[s, o] = M[s, o] \cup \{r\}, (s', o') \neq (s, o) \Rightarrow M'[s', o'] = M'[s, o]$
«Удалить» право $r \in R$ из $M[s, o]$	$s \in S, o \in O$	$S' = S, O' = O, M'[s, o] = M[s, o] \setminus \{r\}, (s', o') \neq (s, o) \Rightarrow M'[s', o'] = M'[s, o]$
«Создать» субъект s'	$s' \notin S$	$S' = S \cup \{s'\}, O' = O \cup \{s'\}, (s, o) \in S' \times O' \Rightarrow M'[s, o] = M[s, o], o \in O' \Rightarrow M'[s', o] = \varnothing, s \in S' \Rightarrow M'[s, o'] = \varnothing$
«Создать» объект o'	$o' \notin O$	$S' = S, O' = O,$ $(s, o) \in S' \times O' \Rightarrow M'[s, o] = M[s, o], s \in S' \Rightarrow M'[s, o'] = \varnothing$
«Уничтожить» субъект s'	$s' \in S$	$S' = S/\{s'\}, O' = O/\{s'\}, (s, o) \in S' \times O' \Rightarrow M'[s, o] = M[s, o]$
«Уничтожить» объект o'	$o' \in O$ $o' \notin S$	$S' = S, O' = O/\{o'\}, (s, o) \in S' \times O' \Rightarrow M'[s, o] = M[s, o]$

Из примитивных операторов могут составляться команды. Каждая команда состоит из двух частей: условия, при котором выполняется команда, и последовательности примитивных операторов.

5.2.3. МОДЕЛЬ РАСПРОСТРАНЕНИЯ ПРАВ ДОСТУПА *TAKE–GRANT*

Модель распространения прав доступа *Take–Grant*, предложенная в 1976 г., используется для анализа систем дискреционного разграничения доступа, в первую очередь, для анализа путей распространения прав доступа в таких системах. В качестве основных элементов модели используются граф доступов и правила его преобразования. Цель модели

– дать ответ на вопрос о возможности получения прав доступа субъектом системы на объект в состоянии, описываемом графом до-ступов. В настоящее время модель *Take–Grant* получила продолжение как расширенная модель *Take–Grant*, в которой рассматриваются пути возникновения информационных потоков в системах с дискреционным разграничением доступа.

Переходя к формальному описанию модели *Take–Grant*, обозначим: O – множество объектов (например, файлов или сегментов памяти); $S \subseteq O$ – множество активных объектов-субъектов (например, пользователей или процессов); $R = \{r_1, r_2, ..., r_m\} \cup (t, g)$ – множество прав доступа, где t (*take*) – право брать права доступа; g (*grant*) – право давать права доступа; $G = (S, O, E)$ – конечный помеченный ориентированный граф без петель, представляющий текущие доступы в системе; множества S, O соответствуют вершинам графа, которые обозначим: ⊗ – объекты (элементы множества $O \backslash S$); • – субъекты (элементы множества S); элементы множества $E \subseteq O \times O \times R$ представляют дуги графа, помеченные непустыми подмножествами из множества прав доступа R.

Состояние системы описывается его графом доступов. Переход системы из состояния в состояние определяется операциями или правилами преобразования графа доступов. Преобразование графа G в граф G' в результате выполнения определенного правила обозначим через $G \vdash_{op} G'$.

В классической модели *Take–Grant* правило преобразования графа может быть одним из четырех, перечисленных ниже.

1. Правило «Брать» – *take* (α, x, y, z). Пусть $x \in S, y, z \in O$ – различные вершины графа G, $\beta \subseteq R$, $\alpha \subseteq \beta$. Правило определяет порядок получения нового графа доступов G' из графа G (рис. 5.3).

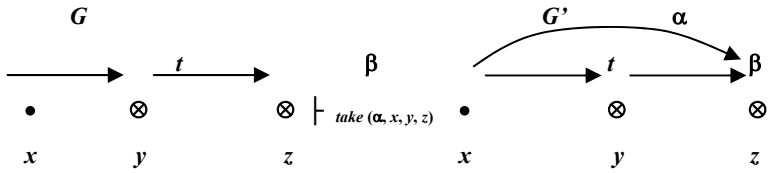


Рис. 5.3. Субъект x берет у объекта y права $\alpha \subseteq \beta$ на объект z

2. Правило «Давать» – *grant* (α, x, y, z). Пусть $x \in S, y, z \in O$ – различные вершины графа G, $\beta \subseteq R$, $\alpha \subseteq \beta$. Правило определяет порядок получения нового графа G' из графа G (рис. 5.4).

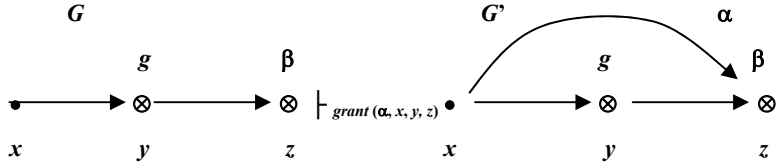


Рис. 5.4. Субъект x дает объекту y права $\alpha \subseteq \beta$ на объект z

3. Правило «Создать» – *create* (β, x, y). Пусть $x \in S$, $\beta \subseteq R$, $\beta \neq \varnothing$. Правило определяет порядок получения нового графа G' из графа G; $y \in O$ – новый объект или субъект (рис. 5.5).

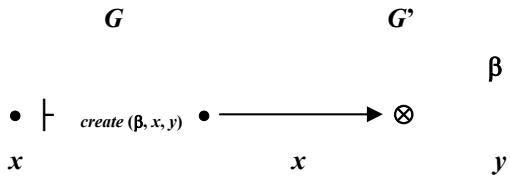


Рис. 5.5. Субъект x создает новый β – доступный объект y

4. Правило «Удалить» – *remove* (α, *x*, *y*). Пусть $x \in S$, $y \in O$ – различные вершины графа *G*, $\beta \subseteq R$, $\alpha \subseteq \beta$. Правило определяет порядок получения нового графа *G*' из графа *G* (рис. 5.6).

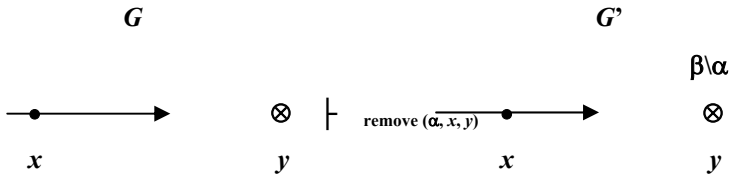


Рис. 5.6. Субъект *x* удаляет права доступа α на объект *y*

Перечисленные правила «Брать», «Давать», «Создать», «Удалить» сведены в табл. 5.3.

5.3. Правила модели *Take-Grant*

Правила модели *Take–Grant*	Условия	Результирующее состояние системы
«Брать» *take* (α, *x*, *y*, *z*)	$x \in S, (x, y, t) \in E, (x, y, \beta) \in E, x \neq z, \alpha \subseteq \beta$	$S' = S, O' = O, E' = E \cup \{(x, z, \alpha)\}$
«Давать» *grant* (α, *x*, *y*, *z*)	$x \in S, (x, y, g) \in E, (x, z, \beta) \in E, y \neq z, \alpha \subseteq \beta$	$S' = S, O' = O, E' = E \cup \{(y, z, \alpha)\}$
«Создать»*create* (β, *x*, *y*)	$x \in S, y \notin O$	$O' = O \cup \{y\}, S' = S \cup \{y\}$, если *y* – субъект; $E' = E \cup \{(x, y, \beta)\}$
«Удалить» *remove* (α, *x*, *y*)	$x \in S, y \in O, (x, z, \beta) \in E, \alpha \subseteq \beta$	$S' = S, O' = O\ E' = E \setminus \{(x, y, \alpha)\}$

В модели *Take–Grant* основное внимание уделяется определению условий, при которых в системе возможно распространение прав до-ступа определенным способом. Рассмотрим условия реализации: способа санкционированного получения прав доступа и способа похищения прав доступа.

5.2.4. МОДЕЛЬ СИСТЕМЫ БЕЗОПАСНОСТИ БЕЛЛА–ЛАПАДУЛА

Классическая модель Белла–Лападула (БЛ) построена для анализа систем защиты, реализующих мандатное (полномочное) разграничение доступа. Возможность ее использования в качестве формальной модели таких систем непосредственно отмечена в критерии *TCSEC* («Оранжевая книга»). Модель БЛ была предложена в 1975 г.

Пусть определены конечные множества: *S* – множество субъектов системы (например, пользователи системы и программы); *O* – множество объектов системы (например, все системные файлы); *R* = (*read, write, append, execute*) – множество видов доступа субъектов из *S* к объектам из *O*, где *read* – доступ на чтение, *write* – *на* запись, *append* – на запись в конец объекта, *execute* – на выполнение.

Обозначим:

$B = \{b \subseteq S \times O \times R\}$ – множество возможных множеств текущих доступов в системе;

$M = \|M_{so}\|$ – матрица разрешенных доступов, где $M_{so} \in R$ – разрешенный доступ субъекта *s* к объекту *o*;

L – множество уровней секретности, например $L = \{U, C, S, TS\}$, где $U < C < S < TS$;

$(f_s, f_o, f_c) \in F = L^s \times L^o \times L^c$ – тройка функций (f_s, f_o, f_c), определяющих:

$f_s : S \to L$ – уровень допуска субъекта;

$f_o : S \to L$ – уровень секретности объекта;

$f_c : S \to L$ – текущий уровень допуска субъекта, при этом $\forall s \in Sf_c(s) \leq f_s(s)$;

H – текущий уровень иерархии объектов;

$V = B \times M \times F \times H$ – множество состояний системы;

Q – множество запросов системе;

D – множество решений по запросам, например {*yes*, *no*, *error*};

$W \in Q \times D \times V \times V$ – множество действий системы, где четверка $(q, d, v_2, v_1) \in W$ означает, что система по запросу *q* с ответом *d* перешла из состояния v_1 в состояние v_2;

No – множество значений времени {*No* = 0, 1, 2, ...};

X – множество функций *x*: $No \to Q$, задающих все возможные последовательности запросов к системе;

Y – множество функций *y*: $No \to D$, задающих все возможные последовательности

ответов системы по запросам;

Z – множество функций z: $No \to V$, задающих все возможные последовательности состояний системы.

Безопасность системы определяется с помощью трех свойств:

ss – свойства простой безопасности (simple security);

*– свойства звезды;

ds – свойства дискретной безопасности (discretionary security).

Оперируя этими свойствами и их сочетаниями возможно построение защиты системы любой сложности.

5.2.5. МОДЕЛЬ *LOW–WATER–MARK*

Модель *Low–Water–Mark* (*LWM*) представляет близкий к модели БЛ подход к определению свойств системы безопасности, реализующей мандатную (полномочную) политику безопасности. В модели *LWM* предлагается порядок безопасного функционирования системы в случае, когда по запросу субъекта ему всегда необходимо предоставлять доступ на запись в объект.

Пусть определены конечные множества: S – множество субъектов системы; O – множество объектов системы; $R = \{read, write\}$ – множество видов доступа субъектов из S к объектам из O.

Обозначим: $B = \{b \subseteq S \times O \times R\}$ – множество возможных множеств текущих доступов в системе; L – множество уровней секретности; $(f_s, f_o) \in \in F = L^s \times L^o$ – двойка функций (f_s, f_o), определяющих: f_s: $S \to L$ – уровень допуска субъекта; уровень допуска субъекта и $f_o : S \to L$ – уровень секретности объекта; $V = B \times F$ – множество состояний системы; $W \subseteq OP \times \times V \times V$ – множество действий системы, где тройка $(op, (b, f), (b^*, f^*)) \in W$ означает, что система в результате выполнения операции $op \in OP$ перешла из состояния (b, f) в состояние (b^*, f^*).

Множество OP содержит операции *read*, *write*, *reset*, описанные в табл. 5.4.

5.4. Основные операции модели *LWM*

Операция	Условиявыполнения	Результат выполнения операции
read (*s, o*)	$f_s(s) \geq f_o(o)$	$f^* = f$; $b^* = b \cup \{(s, o, read)\}$
write (*s, o*)	$f_s(s) = f_o(o)$	$f^*_s = f_s$, $\forall\, o' \neq o$ $f^*_o(o') = f_o(o)$, $f^*_o(o) = f_s(s)$, *if* $(f^*_o(o') < f_o(o))$ *then* $o = \varnothing$, $b^* = b \cup \{(s, o, read)\}$
reset (*s, o*)	$f_s(s) > f_o(o)$	$f^*_s = f_s$, $\forall\, o' \neq o$ $f^*_o(o') = f_o(o')$, $f^*_o(o) = \max(L)$

В результате выполнения операции *write* уровень секретности объекта снижается до уровня доступа субъекта. Если это снижение реально происходит, то вся информация в объекте стирается. В результате выполнения операции *reset* уровень секретности объекта становится максимально возможным в системе.

Таким образом, рассмотренные модели *HRU*, *Take–Grant*, БЛ могут быть использованы при построении политик безопасности и анализе детерминированных систем защиты, т.е. систем, которые не включают элементов, имеющих вероятностную природу. При исследовании систем, закономерности функционирования которых сложны или практически не поддаются описанию, целесообразно использовать элементы теории вероятностей. К числу таких систем можно отнести глобальные вычислительные сети, например Internet, или современные многозадачные, многопользовательские сетевые операционные системы.

5.2.6. МОДЕЛИ РОЛЕВОГО РАЗГРАНИЧЕНИЯ ДОСТУПА

5.2.6.1. Базовая модель ролевого разграничения доступа

Базовая модель ролевого разграничения доступа (РРД) определяет самые общие принципы построения моделей РРД.

Основными элементами базовой модели РРД являются:

U – множество пользователей;

R – множество ролей;

P – множество прав доступа на объекты системы;

S – множество сессий пользователей;

PA: $R \to 2^P$ – функция, определяющая для каждой роли множество прав доступа; при этом для каждого $p \in P$ существует $r \in R$ такая, что $p \in PA(r)$;

UA: $U \to 2^R$ – функция, определяющая для каждого пользователя множество ролей, на которые он может быть авторизован;

user: $S \to U$ – функция, определяющая для каждой сессии пользователя, от имени которого она активизирована;

roles: $S \to 2^R$ – функция, определяющая для пользователя множество ролей, на которые он авторизован в данной сессии; при этом в каждый момент времени для каждого $s \in S$ выполняется условие $roles(s) \subseteq UA(user(s))$. Принципиально могут существовать роли, на которые не авторизован ни один пользователь.

В базовой модели РРД предполагается, что множества U, R, P и функции PA, UA не изменяются с течением времени. Множество ролей, на которые авторизуется пользователь в течение одной сессии, модифицируется самим пользователем. В базовой модели РРД отсутствуют механизмы, позволяющие одной сессии активизировать другую сессию. Все сессии активизируются пользователем.

Для обеспечения соответствия реальным системам, каждый пользователь которых занимает определенное положение в служебной иерархии, на множестве ролей реализуется иерархическая структура.

Иерархией ролей в базовой модели РРД называется заданное на множестве ролей R отношение частичного порядка «≤» (отношение «≤» обладает свойствами рефлексивности, антисимметричности и транзитивности). При этом выполняется условие для

$$u \in U, \text{ если } r, r' \in R, r \in UA(u) \text{ и } r' \le r, \text{ то } r' \in UA(u).$$

Таким образом, наряду с данной ролью пользователь должен быть авторизован на все роли, в ее низших уровнях иерархии.

Другим важным механизмом базовой модели РРД являются ограничения, накладываемые на множества ролей, на которые может быть авторизован пользователь или на которые он авторизуется в течение одной сессии.

В базовой модели РРД заданы ограничения статического взаимного исключения ролей или прав доступа, если выполняются условия:

$R = R_1 \cup \ldots \cup R_n$, где $R_i \cap R_j = \varnothing$ для $1 \le i < j \le n$;

$|UA(u) \cap R_1| \le 1$ для $u \in U$, $i \in 1, 2, \ldots, n$;

$P = P_1 \cup \ldots \cup R_m$, где $P_i \cap P_j = \varnothing$ для $1 \le i < j \le m$;

$|PA(r) \cap P_1| \le 1$ для $p \in U$, $i \in 1, 2, \ldots, m$.

Множество ролей и множество прав доступа разделяются на непересекающиеся подмножества. При этом каждый пользователь может обладать не более, чем одной ролью из каждого подмножества ролей, а каждая роль – не более, чем одним правом доступа из каждого подмножества прав доступа.

В базовой модели РРД задано ограничение динамического взаимного исключения ролей, если выполняются условия:

$R = R_1 \cup \ldots \cup R_n$, где $R_i \cap R_j = \varnothing$ для $1 \le i < j \le n$;

$|roles(s) \cap R_i| \le 1$ для $s \in S$, $i \in 1, 2, \ldots, n$.

Множество ролей разделяется на непересекающиеся подмножества. При этом в каждой сессии пользователь может обладать не более, чем одной ролью из каждого подмножества ролей.

В базовой модели РРД заданы статические количественные ограничения на обладание ролью или правом доступа, если определены две функции:

$$\alpha : R \to N_0; \qquad \beta : P \to N_0,$$

где N_0 – множество натуральных чисел с нулем, и выполняются условия:

$$|UA^{-1}(r)| \le \alpha(r) \text{ для } r \in R;$$

$$|PA^{-1}(p)| \le \beta(p) \text{ для } p \in P.$$

Для каждой роли устанавливается максимальное число пользователей, которые могут быть на нее авторизованы, а для каждого права доступа устанавливается максимальное число ролей, которые могут им обладать.

В базовой модели РРД задано динамическое количественное ограничение на обладание ролью, если определена функция

$$\gamma : R \to N_0$$

и выполняется условие

$$|\, roles^{-1}(r)\,| \le \gamma\,(r) \text{ для } r \in R.$$

Для роли устанавливается максимальное число сессий пользователей, которые могут одновременно быть на нее авторизованы.

В базовой модели РРД заданы статические ограничения необходимого обладания ролью или правом доступа, если определены две функции:

$$\alpha : R \to 2^R; \qquad \beta : P \to 2^P,$$

и выполняются условия:

- для $u \in U$, если $r, r' \in R$, $r \in UA(u)$ и $r' \in \alpha\,(r)$, то $r' \in UA(u)$;
- для $r \in R$, если $p, p' \in P$, $p \in PA(r)$ и $p' \in \beta\,(p)$, то $p' \in PA(r)$.

Для каждой роли для того чтобы на нее мог быть авторизован пользователь, могут быть определены роли, на которые пользователь также должен быть авторизован. Для каждого права доступа, для того чтобы им обладала роль, могут быть определены права доступа, которыми данная роль также должна обладать.

В базовой модели РРД задано динамическое ограничение необходимого обладания ролью, если определена функция

$$\gamma : R \to 2^R$$

и выполняется условие

для $s \in S$, если $r, r' \in R$, $r \in roles(s)$ и $r' \in \gamma\,(r)$, то $r' \in roles(s)$.

Для каждой роли, для того чтобы на нее мог быть авторизован пользователь в некоторой сессии, могут быть определены роли, на которые пользователь также должен быть авторизован в данной сессии.

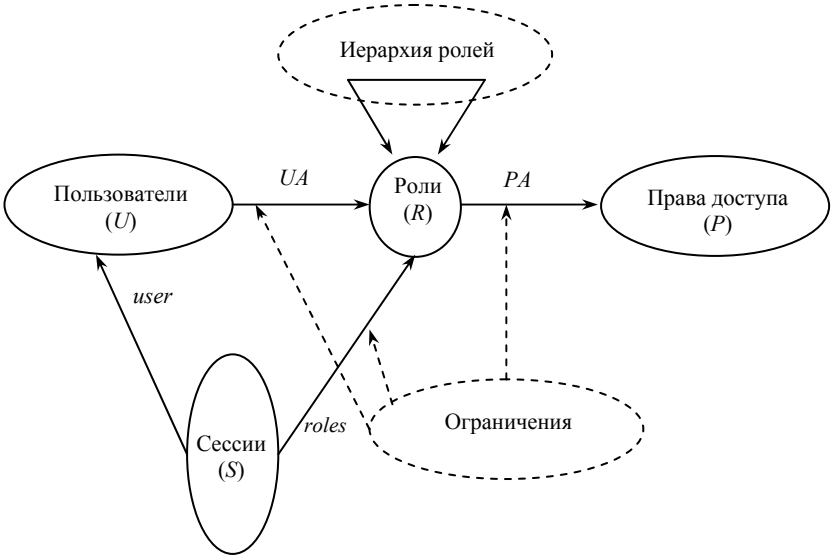


Рис. 5.7. Структура элементов базовой модели РРД

Общая структура элементов базовой модели РРД имеет вид, представленный на рис. 5.7.

5.2.6.2. Модель администрирования ролевого разграничения доступа

Структура модели администрирования ролевого разграничения доступа. Как уже отмечалось, в базовой модели РРД предполагается, что множества U, R, P и функции PA и UA не изменяются с течением времени или существует единственная роль – «офицер безопасности», которая предоставляет возможность изменять эти множества и функции. В реальных компьютерных системах, в которых одновременно могут работать сотни и тысячи пользователей, а структура ролей и прав доступа может быть очень сложной, проблема администрирования является чрезвычайно важной задачей. Для решения этой задачи рассматривается построенная на основе базовой модели РРД модель администрирования РРД.

В дополнение к используемым элементам базовой модели РРД в модели администрирования РРД рассматриваются следующие элементы (рис. 5.8):

AR – множество административных ролей ($AR \cap R = \varnothing$);

AP – множество административных прав доступа ($AP \cap P = \varnothing$);

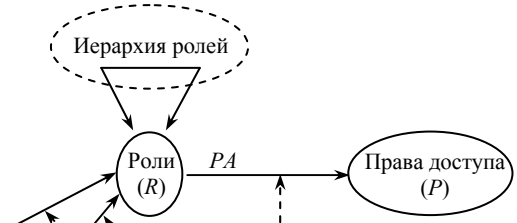


Рис. 5.8. Структура элементов модели администрирования РРД

$APA : AR \to 2^{AP}$ – функция, определяющая для каждой административной роли множество административных прав доступа; при этом для каждого $p \in AP$ существует $r \in AR$ такая, что $p \in APA(r)$;

$AUA : U \to 2^{AR}$ – функция, определяющая для каждого пользователя множество административных ролей, на которые он может быть авторизован.

Кроме того, переопределяется следующая функция:

roles: $S \to 2^R \cup 2^{AR}$ – функция, определяющая для пользователя множество ролей, на которые он авторизован в данной сессии; при этом в каждый момент времени для каждого $s \in S$ выполняется условие $roles(s) \subseteq UA(user(s))$ и $AUA(user(s))$.

Как и в базовой модели РРД, в модели администрирования РРД реализуются иерархия административных ролей и механизмы ограничений.

Иерархией ролей в модели администрирования РРД называется заданное на множестве ролей *AR* отношение частичного порядка «≤». При этом выполняется условие

• для $u \in U$, если $r, r' \in AR$, $r \in AUA(u)$ и $r' \le r$, то $r' \in AUA(u)$.

Административные роли могут быть разделены на три группы по своему назначению:

1) администрирование множеств авторизованных ролей пользователей;
2) администрирование множеств прав доступа, которыми обладают роли;
3) администрирование иерархии ролей.

Как правило, каждой административной роли назначают подмножество иерархии ролей, параметры которых данная административная роль позволяет изменять. Рассмотрим каждую из групп административных ролей подробнее.

Администрирование множеств авторизованных ролей пользователей. **При администрировании множеств авторизованных ролей пользователей изменяются значения функции (*UA*) Для изменения значений функции (*UA*) определяются специальные административные роли из множества *AR*.**

Для администрирования множеств авторизованных ролей пользователей необходимо определять:

• для каждой административной роли множество ролей, множества авторизованных пользователей которых она позволяет изменять;

• для каждой роли предварительное условие, которому должны соответствовать пользователи, прежде чем они будут включены в множество ее авторизованных пользователей.

Пусть задана иерархия ролей (рис. 5.9, *а*) и иерархия административных ролей (рис. 5.9, *б*).

Минимальная роль в иерархии – служащий (*E*). Иерархия ролей разработчиков проектов имеет максимальную роль – директор *(DIR*), минимальную роль – инженер (*ED*). В

управлении выполняются работы по двум проектам. В каждом проекте определены максимальная роль – руководитель проекта (*PL*1, *PL*2 соответственно), минимальная роль – инженер проекта (*E*1, *E*2 соответственно) и не сравнимые между собой роли – инженер по производству (*PE*1, *PE*2 соответственно) и инженер по контролю (*QE*1, *QE*2 соответственно). Иерархия административных ролей состоит из четырех ролей с максимальной ролью – старший офицер безопасности (*SSO*).

В рассматриваемом примере административная роль *PSO*1 позволяет включать в множества авторизованных ролей пользователя роли *PE*1, *QE*1, *E*1. При этом, для того чтобы любая из перечисленных ролей могла быть включена в множество авторизованных ролей пользователя, он уже должен обладать ролью *ED*.

Для администрирования множеств авторизованных ролей пользователей на множестве административных ролей задаются следующие функции:

• *can-assign*: $AR \to CR \times 2^R$ – функция, определяющая для каждой административной роли множество ролей, которые могут быть включены в множество авторизованных ролей пользователя с использованием данной административной роли при выполнении заданных предварительных условий;

• *can-revoke:* $AR \to 2^R$ – функция, определяющая для каждой административной роли множество ролей, которые могут быть исключены из множества авторизованных ролей пользователя с использованием данной административной роли.

Как правило, множество ролей, являющееся значением функции *can-assign*() или *can-revoke*(), задается интервалом ролей одного из четырех видов:

$[x, y] = \{x' \in R, \text{где } x \le x' \text{ и } x' \le y\}$;

$(x, y] = \{x' \in R, \text{где } x < x' \text{ и } x' \le y\}$;

$[x, y) = \{x' \in R, \text{где } x \le x' \text{ и } x' < y\}$;

$(x, y) = \{x' \in R, \text{где } x < x' \text{ и } x' < y\}$, где $x, y \in R$.

Иерархия ролей и иерархия административных ролей представлена в табл. 5.5 и 5.6.

5.5. Значение функции *can-assign*( )

Административная роль	Предварительное условие	Множество ролей
*PSO*1	*ED*	[*E*1, *PL*1)
*PSO*2	*ED*	[*E*2, *PL*2)
DSO	*ED and* (*not PL1*)	[*PL*2, *PL*2]
DSO	*ED and* (*not PL*2)	[*PL*1, *PL*1]

5.6. Значение *функции can-revoke*( )

Административная роль	Множество ролей
*PSO*1	[*E*1, *PL*1)
*PSO*2	[*E*2, *PL*2)
DSO	(*ED*, *DIR*)

Таким образом, административная роль *PSO1* позволяет включить для пользователя, уже обладающего ролью *ED*, в множество его авторизованных ролей роли *E*1, *PE*1 и *QE*1. Административная роль *DSO* позволяет включить для пользователя, уже обладающего ролью *ED*, в множество его авторизованных ролей роль *PL*1; при этом данный пользователь не должен обладать ролью *PL*2.

Следует отметить, что в общем случае значения функций *can-assign*() и *can-revoke*() определяются независимо друг от друга, т.е. удаление роли из множества авторизованных ролей пользователя может быть выполнено независимо от того, каким образом эта роль была включена в это множество, и наоборот.

Администрирование множеств прав доступа ролей. **При администрировании множеств прав доступа ролей изменяются значения функции *PA*(), для чего определяются специальные административные роли из множества *AR*.**

Подход к определению порядка администрирования множеств прав доступа ролей аналогичен подходу, использованному для администрирования множеств авторизованных ролей пользователя.

При этом предварительные условия определяются для прав доступа.

Для администрирования множеств прав доступа, которыми обладают роли, на множестве административных ролей задаются:

• *can-assign*$_p$: $AR \to CR \times 2^R$ – функция, определяющая для каждой административной роли множество ролей, для которых разрешено включать права доступа в множества прав доступа с использованием данной административной роли при выполнении заданных предварительных условий;

• *can-revoke*$_p$: $AR \to 2^R$ – функция, определяющая для каждой административной роли множество ролей, для которых разрешено удаление прав доступа из множеств прав доступа с использованием данной административной роли.

Иерархия ролей и иерархия административных ролей представлена в табл. 5.7 и 5.8.

Таким образом, административная роль *DSO* позволяет включить права доступа, входящие в множество прав доступа роли *DIR*, в множества прав доступа ролей *PL*1, *PL*2. Причем данные права доступа могут быть включены в множества прав доступа ролей, находящихся ниже *PL*1, *PL*2 по иерархии. Административная роль *PSO*1 позволяет

включить права доступа, входящие в множество прав доступа роли *PL*1, в множества прав доступа ролей *PE*1 и *QE*1, но только в каждую по отдельности. Административная роль *DSO* позволяет удалять права доступа из множеств прав доступа ролей, находящихся в иерархии между ролями *ED* и *DIR*, а административная роль *PSO*1 позволяет удалять права доступа из множеств прав доступа ролей *PE*1 и *QE*1.

5.7. Значение функции *can-assign*( )

Административная роль	Предварительное условие	Множество ролей
	DIR	[*PL*1, *PL*1]
	DIR	[*PL*2, *PL*2]
DSO	*PL1 and*	[*PE*1, *PE*1]
DSO	(*not QE*1)	[*QE*1, *QE*1]
*PSO*1	*PL1 and*	[*PE*2, *PE*2]
*PSO*1	(*not PE*1)	[*QE*2, *QE*2
*PSO*2	*PL2 and*	
*PSO*2	(*not QE*2)	
	PL2 and	
	(*not PE*2)	

5.8. Значение функции *can-revoke*( )

Административная роль	Множество ролей
DSO	(*ED*,
*PSO*1	*DIR*)
*PSO*1	[*QE*1,
*PSO*2	*QE*1]
*PSO*2	[*PE*1,
	*PE*1]
	[*QE*2,
	*QE*2]
	[*PE*2,
	*PE*2]

В общем случае значения функции $can\text{-}revoke_p()$ определяются без учета иерархии ролей. В связи с этим необходимо решать проблему удаления прав доступа ролей, аналогичную рассмотренной ранее проблеме удаления ролей из множеств авторизованных ролей пользователей.

Администрирование иерархии ролей. **Определение правил администрирования, позволяющих изменять иерархию ролей, является самой сложной задачей, рассматриваемой в модели администрирования РРД. Для решения данной задачи используются подходы, реализованные при определении правил администрирования множеств авторизованных ролей пользователей и прав доступа ролей. Задаются три иерархии, элементами которых соответственно являются:**

- возможности – множества прав доступа и других возможностей;
- группы – множества пользователей и других групп;
- объединения – множества пользователей, прав доступа, групп, возможностей и других объединений.

Иерархия объединений является наиболее общей и может включать в себя иерархии возможностей и групп. Определение возможностей и групп требуется для обеспечения соответствия правил администрирования ролей в модели и используемых на практике технологий обработки информации и создания административных структур организаций. Например, для выполнения своих функций пользователю может быть необходим некоторый набор прав доступа, причем отсутствие в этом наборе некоторого права доступа может сделать бессмысленным обладание имеющимися правами.

На основе иерархий возможностей, групп и объединений задается иерархия ролей, элементами которой являются (*UP*-роли):

- роли-возможности – роли, которые обладают только определенными в соответствующей возможности правами доступа;
- роли-группы – роли, на которые могут быть авторизованы одновременно только все пользователи соответствующей группы;
- роли-объединения – роли, которые обладают возможностями, правами доступа и на которые могут быть авторизованы группы пользователей и отдельные пользователи.

Роли-объединения являются общим случаем ролей, рассматриваемых в модели администрирования РРД.

Для администрирования возможностей и групп пользователей на множестве административных ролей задаются:

- $can\text{-}assign_a$: $AR \rightarrow CR \times 2^{UPR}$ – функция, определяющая для каждой административной роли множество ролей-объединений, в множество прав доступа которых разрешено включать возможности с использованием данной административной роли при выполнении заданных предварительных условий;
- $can\text{-}revoke_a$: $AR \rightarrow 2^{UPR}$ – функция, определяющая для каждой административной роли множество ролей-объединений, из множества прав доступа которых разрешено удаление возможностей с использованием данной административной роли;
- $can\text{-}assign_g$: $AR \rightarrow CR \times 2^{UPR}$ – функция, определяющая для каждой административной роли множество ролей-объединений, которые разрешено включать в множество авторизованных ролей групп пользователей с использованием данной административной

роли при выполнении заданных предварительных условий;

• *can-revoke*$_g$: $AR \rightarrow 2^{UPR}$ – функция, определяющая для каждой административной роли множество ролей-объединений, которые разрешено удалять из множества авторизованных ролей групп пользователей с использованием данной административной роли.

Необходимость определения иерархий возможностей и групп обусловлена также тем, что построенные на их основе правила администрирования отличны друг от друга. Предварительные условия, определенные в функции *can-assign*$_a$(), используются аналогично предварительным условиям, определенным в функции *can-assign*$_g$(), а предварительные условия, определенные в функции *can-assign*$_g$(), используются аналогично предварительным условиям, определенным в функции *can-assign*$_a$(). Например, для того чтобы в множество прав доступа роли-объединения была включена возможность, данная возможность должна входить в множества прав доступа ролей, указанных в предварительном условии функции *can-assign*$_a$(), а для того чтобы роль-объединение была включена в множество авторизованных ролей группы пользователей, пользователи данной группы должны уже обладать ролями в соответствии с предварительным условием функции *can-assign*$_g$().

Включение возможности в множество прав доступа роли-объединения означает, что соответствующая роль-возможность в иерархии ролей станет непосредственно «ниже» роли-объединения. Наоборот, включение роли-объединения в множество авторизованных ролей группы пользователей означает, что соответствующая роль-группа в иерархии ролей станет непосредственно «выше» роли-объединения.

Для администрирования иерархии ролей (добавления и удаления ролей, включения или удаления отношений иерархии) на множестве административных ролей задается *can-modify*: $AR \rightarrow 2^{UPR}$ – функция, определяющая для каждой административной роли интервал ролей (исключая границы интервала), на котором возможно изменение иерархии ролей с использованием данной административной роли.

5.2.6.3. Модель мандатного ролевого разграничения доступа

Защита от угрозы конфиденциальности информации. **Ролевое разграничение доступа является развитием дискреционного разграничения доступа. В то же время оно является достаточно гибким, и, используя механизм ролей, возможна реализация мандатной политики безопасности, ориентированной на защиту от угрозы конфиденциальности информации.**

Рассмотрим подход, реализующий мандатное разграничение доступа на основе базовой модели РРД.

Используем следующие обозначения:

U – множество пользователей (субъектов);

O – множество объектов;

$(L, \leq)$ – решетка уровней конфиденциальности;

c: $U \rightarrow L$ – функция уровня доступа пользователя;

c: $O \rightarrow L$ – функция уровня конфиденциальности объекта;

$A = \{read, write\}$ – виды доступа.

Будем различать два вида мандатного разграничения доступа: либеральный и строгий (Белла–ЛаПадула).

Доступ (u, o, r) является безопасным для либерального мандатного разграничения доступа, если выполняется одно из условий:

$r = read$ и $c(u) \geq c(o)$ (*ss* – свойство);

$r = write$ и, если существует доступ $(u, o', read)$, то $c(o) \geq c(o')$ (либеральное *– свойство).

Доступ (u, o, r) является безопасным для строгого мандатного разграничения доступа, если выполняется одно из условий:

$r = read$ и $c(u) \geq c(o)$ (*ss* – свойство);

$r = write$ и, если существует доступ $(u, o', read)$, то $c(o) = c(o')$ (строгое *– свойство).

Построим систему РРД. Пусть

$R = \{x\_read \mid x \in L\} \cup \{x\_write \mid x \in L\}$ – множество ролей;

$P = \{(o, read) \mid o \in 0\} \cup \{(o, write) \mid o \in O\}$ – множество прав доступа.

Зададим на множестве ролей R иерархию; при этом иерархии ролей на множествах $\{x\_read \mid x \in L\}$ и $\{x\_write \mid x \in L\}$ будут независимы.

Иерархией на множестве ролей R в соответствии с требованиями либерального мандатного разграничения доступа называется отношение частичного порядка «≤», где для $r, r' \in R$ справедливо неравенство $r \leq r'$, если выполняется одно из условий:

$r = x\_read$, $r' = x'\_read$ и $x \leq x'$;

$r = x\_write$, $r' = x\_write$ и $x' \le x$.

Иерархией на множестве ролей R в соответствии с требованиями строгого мандатного разграничения доступа называется отношение частичного порядка «≤», где для r, $r' \in R$ справедливо неравенство $r \le r'$, если выполняется одно из условий:

$r = x\_read$, $r' = x'\_read$ и $x \le x'$;

$r = x\_write$, $r' = x'\_write$ и $x = x'$ (каждая роль вида x_write сравнима только сама с собой).

Модель РРД соответствует требованиям либерального мандатного разграничения доступа и выполняются ограничения:

- ограничение функции $UA()$ – для каждого пользователя $u \in U$ роль $x\_read = \oplus (UA(u) \cap \{y\_read \mid y \in L\}) \in UA\{u)$ (здесь $x = c(u)$) и $x\_write = \oplus \{y\_write \mid y \in L\} \in UA(u)$ (здесь $x = \oplus L$);
- ограничение функции $roles()$ – для каждой сессии $s \in S$ множество $roles(s) = \{x\_read, x\_write\}$;
- ограничение функции $PA()$ – должно выполняться:

– для каждого $x \in L$ доступ (о, $read$) $\in PA(x\_read)$ тогда и только тогда, когда доступ (o, $write$) $\in PA(x\_write)$;

– для каждого доступа (o, $read$) существует единственная роль x_read: (o, $read$) $\in PA(x\_read)$ (здесь $x = c(o)$).

Модель РРД соответствует требованиям строгого мандатного разграничения доступа и выполняются ограничения:

- ограничение функции $UA()$ – для каждого пользователя $u \in U$ роль $x\_read = \oplus (UA(u) \cap \{y\_read \mid y \in L\}) \in UA(u)$ (здесь $x = c(u)$) и $UA(u) = \{yjwrite \mid y \in L\}$;
- ограничение функции $roles()$ – для каждой сессии $s \in S$ множество $roles(s) = \{x\_read, x\_write\}$;
- ограничение функции $PA()$ – должно выполняться:

– для каждого $x \in L$ доступ (o, $read$) $\in PA(x\_read)$ тогда и только тогда, когда доступ (o, $write$) $\in PA(x\_write)$;

– для каждого доступа (o, $read$) существует единственная роль x_read: (o, $read$) $\in PA(x\_read)$ (здесь $x = c(o)$).

Таким образом, требования соответствия либеральному и строгому мандатному разграничению доступа для моделей РРД совпадают во всем, кроме требований к соответствующей иерархии ролей и ограничениям на функцию UA.

Контрольные вопросы

1. Что понимается под политикой безопасности?
2. Дайте определения компонентам, связанным с понятием «политика безопасности»?
3. Какие модели основных типов политик безопасности вы знаете?
4. Охарактеризуйте дискреционные, мандатные и ролевые модели политик безопасности.

6. КРИПТОГРАФИЧЕСКАЯ ЗАЩИТА ИНФОРМАЦИИ

С распространением письменности в человеческом обществе появилась потребность в обмене письмами и сообщениями, что вызвало необходимость сокрытия содержимого письменных сообщений от посторонних. Методы сокрытия содержимого письменных сообщений можно разделить на три группы. К первой группе относятся методы маскировки или *стеганографии*, которые осуществляют сокрытие самого факта наличия сообщения; вторую группу составляют различные методы тайнописи или *криптографии* (от греческих слов *ktyptos* – тайный и *grapho* – пишу); методы третьей группы ориентированы на создание специальных технических устройств, засекречивания информации.

Практически одновременно с криптографией стал развиваться и криптоанализ – наука о раскрытии шифров (ключей) по шифртексту.

Вторая мировая война дала новый толчок развитию криптографии и криптоанализа, что было вызвано применением технических средств связи и боевого управления. Для разработки новых шифров и работы в качестве криптоаналитиков привлекались ведущие ученые. В годы Второй мировой войны был разработан ряд механических устройств для

шифрования сообщений.

В 1949 г. была опубликована статья Клода Шеннона «Теория связи в секретных системах», которая подвела научную базу под криптографию и криптоанализ. С этого времени стали говорить о КРИПТОЛОГИИ (от греческого *kryptos* – тайный и *logos* – сообщение) – науке о преобразовании информации для обеспечения ее секретности. Этап развития криптографии и криптоанализа до 1949 г. стали называть донаучной криптологией.

Криптография является одним из наиболее мощных средств обеспечения конфиденциальности и контроля целостности информации. Во многих отношениях она занимает центральное место среди программно-технических регуляторов безопасности. Например, для портативных компьютеров, физически защитить которые крайне трудно, только криптография позволяет гарантировать конфиденциальность информации даже в случае кражи.

6.1. ОСНОВНЫЕ ОПРЕДЕЛЕНИЯ КРИПТОЛОГИИ

Защита данных с помощью шифрования – одно из возможных решений проблемы безопасности. Зашифрованные данные становятся доступными только тем, кто знает, как их расшифровать, и поэтому похищение зашифрованных данных абсолютно бессмысленно для несанкционированных пользователей.

Коды и шифры использовались задолго до появления ЭВМ. С теоретической точки зрения не существует четкого различия между кодами и шифрами. Однако в современной практике различие между ними является достаточно четким. *Коды* оперируют *лингвистическими элементами*, разделяя шифруемый текст на такие смысловые элементы, как слова и слоги. В *шифре* всегда различают два элемента: *алгоритм и ключ*.

Алгоритм позволяет использовать сравнительно короткий ключ для шифрования сколь угодно большого текста.

Определим ряд терминов, используемых в криптологии.

Под *шифром* понимается совокупность обратимых преобразований множества открытых данных на множество зашифрованных данных, заданных алгоритмом криптографического преобразования.

Ключ – конкретное секретное состояние некоторых параметров алгоритма криптографического преобразования данных, обеспечивающее выбор одного варианта из совокупности всевозможных для данного алгоритма.

Гаммирование – процесс наложения по определенному закону гаммы шифра на открытые данные.

Гамма шифра – псевдослучайная двоичная последовательность, вырабатываемая по заданному алгоритму, для зашифрования открытых данных и расшифрования зашифрованных данных.

Зашифрованием данных называется процесс преобразования открытых данных в зашифрованные с помощью шифра, а *расшифрованием* данных – процесс преобразования закрытых данных в открытые с помощью шифра.

Шифрованием называется процесс зашифрования или расшифрования данных.

Дешифрованием называется процесс преобразования закрытых данных в открытые при неизвестном ключе и, возможно, неизвестном алгоритме.

Криптостойкостью называется характеристика шифра, определяющая его стойкость к дешифрованию. Обычно эта характеристика определяется периодом времени, необходимым для дешифрования.

Имитозащита – защита от навязывания ложных данных. Для обеспечения имитозащиты к зашифрованным данным добавляется имитовставка, представляющая собой последовательность данных фиксированной длины, полученную по определенному правилу из открытых данных и ключа.

Криптографическая защита – это защита данных с помощью криптографического преобразования, под которым понимается преобразование данных шифрованием и (или) выработкой имитовставки.

Синхропосылка – исходные открытые параметры алгоритма криптографического преобразования.

Уравнение зашифрования (*расшифрования*) – соотношение, описывающее процесс образования зашифрованных (открытых) данных из открытых (зашифрованных) данных в результате преобразований, заданных алгоритмом криптографического преобразования.

Современные методы шифрования должны отвечать следующим требованиям:

1. Стойкость шифра противостоять криптоанализу должна быть такой, чтобы вскрытие его могло быть осуществлено только путем решения задачи полного перебора ключей.

2. Криптостойкость обеспечивается не секретностью алгоритма, а секретностью

ключа.

3. Шифртекст не должен существенно превосходить по объему исходную информацию.

4. Ошибки, возникающие при шифровании, не должны приводить к искажениям и потерям информации.

5. Время шифрования не должно быть большим.

6. Стоимость шифрования должна быть согласована со стоимостью закрываемой информации.

6.2. КЛАССИФИКАЦИЯ МЕТОДОВ КРИПТОГРАФИЧЕСКОГО ЗАКРЫТИЯ ИНФОРМАЦИИ

В настоящее время известно большое число методов криптографического закрытия информации. Классификация методов шифрования (криптоалгоритмов) может быть осуществлена по следующим признакам:

- по типу ключей: симметричные и асимметричные криптоалгоритмы;
- по размеру блока информации: потоковые и блочные шифры;
- по характеру воздействий, производимых над данными: метод замены (перестановки), метод подстановки; аналитические методы, аддитивные методы (гаммирование), комбинированные методы.

Кодирование может быть смысловое, символьное, комбинированное.

Закрытие информации другими способами может достигаться с помощью стеганографии, сжатия/расширения, рассечения/разнесения

Классификация методов криптографического закрытия информации приведена на рис. 6.1.

6.3. ОСНОВЫ ТЕОРИИ К. ШЕННОНА

Клод Шеннон рассмотрел модель (см. рис. 6.2), в которой источник сообщений порождает открытый текст X. Источник ключей генерирует ключ Z.

Шифратор преобразовывает открытый текст X с помощью ключа Z в шифртекст Y: $Y = TzX$.

Дешифратор, получив зашифрованное сообщение Y, выполняет обратную операцию: $X = Tz^{(-1)}Y$.

Задачей криптоаналитика противника является получение открытого текста и ключа на основе анализа шифртекста.

Шеннон рассмотрел вопросы теоретической и практической секретности. Для определения теоретической секретности Шеннон сформулировал следующие вопросы:

1. Насколько устойчива система, если криптоаналитик противника не ограничен временем и обладает всеми необходимыми средствами для анализа криптограмм?

2. Имеет ли криптограмма единственное решение?

3. Какой объем шифртекста необходимо перехватить криптоаналитику, чтобы решение стало единственным?

Для ответа на эти вопросы Шеннон ввел понятие совершенной секретности с помощью следующего условия: для всех Y апостериорные вероятности равны априорным вероятностям, т.е. перехват зашифрованного сообщения не дает криптоаналитику противника никакой информации.

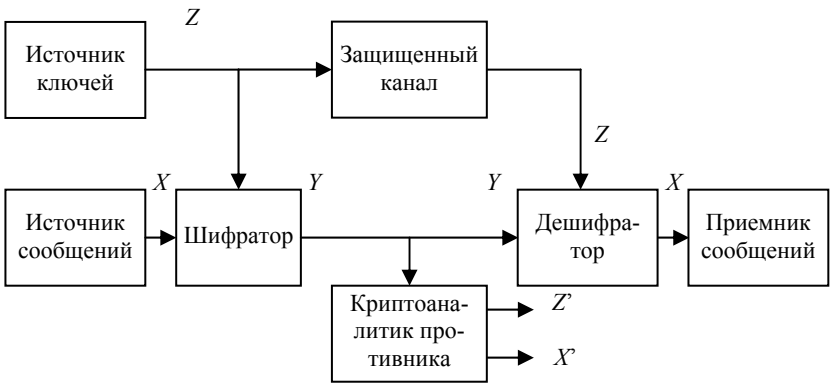


Рис. 6.2. Общая схема передачи шифрованных сообщений

По теореме Байеса

$$P_y(X) = P(X)P_x(Y)/P(Y),$$

где $P(X)$ – априорная вероятность сообщения X; $P_x(Y)$ – условная вероятность криптограммы Y при условии, что выбрано сообщение X, т.е. сумма вероятностей всех тех ключей, которые переводят сообщение X в криптограмму Y; $P(Y)$ – вероятность получения криптограммы Y; $P_y(X)$ – апостериорная вероятность сообщения X при условии, что перехвачена криптограмма Y. Для совершенной секретности значения $P_y(X)$ и $P(X)$ должны быть равны для всех X и Y.

Для противодействия методам статистического анализа криптограмм Шеннон предложил использовать два метода: рассеивание и перемешивание.

6.4. ОСНОВНЫЕ КРИПТОГРАФИЧЕСКИЕ МОДЕЛИ И АЛГОРИТМЫ ШИФРОВАНИЯ

6.4.1. СИММЕТРИЧНЫЕ МЕТОДЫ ШИФРОВАНИЯ

В данных методах один и тот же ключ (хранящийся в секрете) используется и для шифровки, и для расшифровки сообщений. Существуют весьма эффективные методы симметричного шифрования. Существует и стандарт на подобные методы – ГОСТ 28147-89 «Системы обработки информации. Защита криптографическая. Алгоритм криптографического преобразования».

Иллюстрация использования симметричного шифрования показана на рис. 6.3. Для определенности будем вести речь о защите сообщений, хотя события могут развиваться не только в пространстве, но и во времени, когда шифруются и расшифровываются никуда не перемещающиеся файлы.

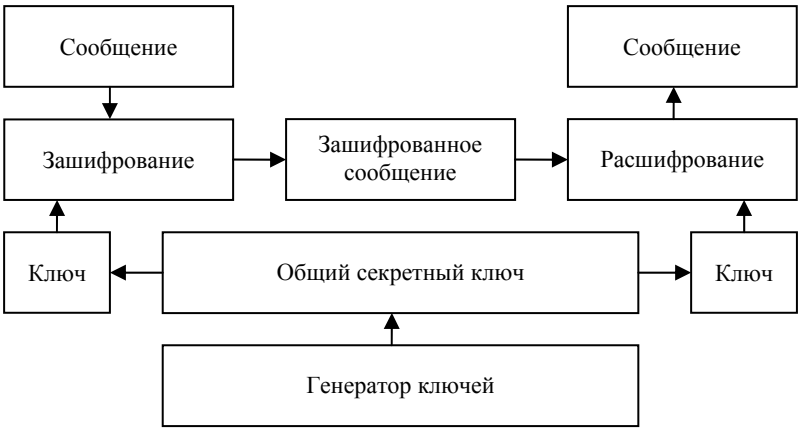


Рис. 6.3. Использование симметричного метода шифрования

Основным недостатком симметричного шифрования является то, что секретный ключ должен быть известен и отправителю, и получателю. С одной стороны, это ставит новую проблему рассылки ключей. С другой стороны, получатель на основании наличия шифрованного и расшифрованного сообщения не может доказать, что он получил это сообщение от конкретного отправителя, поскольку такое же сообщение он мог сгенерировать и сам.

6.4.1.1. Методы замены

Шифрование методом замены (подстановки) основано на алгебраической операции, называемой подстановкой. *Подстановкой* называется взаимнооднозначное отображение некоторого конечного множества M на себя. Число N элементов этого множества называется степенью подстановки. Природа множества M роли не играет, поэтому можно считать, что $M = \{1, 2, ..., N\}$.

Если при данной подстановке S число j переходит в Ij, то подстановка обозначается символом S:

$$S = \begin{bmatrix} 1 & 2 & & n \\ I_1 & I_2 & & I_n \end{bmatrix}$$

В этой записи числа 1, 2, ..., n можно произвольным образом переставлять, соответственно переставляя числа I_1, I_2, ..., I_n. Результат последовательного выполнения двух подстановок S_1 и S_2 одной и той же степени также является подстановкой, которая называется произведением подстановок S_1 и S_2 и обозначается S_1S_2.

Пусть S – произвольная подстановка степени n. Если для некоторого j число I_j отлично от j, то говорят, что подстановка S действительно перемещает число j; в противном случае – подстановка S оставляет число j на месте.

Количество m чисел, действительно перемещаемых подстановкой S, называется *длиной цикла подстановки*.

Подстановка S называется *транспозицией*, если существует пара (j_1, j_2) различных элементов из M, удовлетворяющих условиям:

$Ij_1 = j_2$, $Ij_2 = j_2$, $Ij = j$ для каждого $j \in \{M \backslash \{j_1, j_2\}\}$. Любая подстановка разлагается в произведение транспозиций.

В криптографии рассматриваются четыре типа подстановки (замены): моноалфавитная, гомофоническая, полиалфавитная и полиграммная.

Далее всюду в примерах, где необходимо, будем использовать кодирование букв русского алфавита, приведенное в табл. 6.1. Знак «_» в табл. 6.1. и далее означает пробел.

6.1. Коды букв русского алфавита

Буква	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	Ъ	Ы	Ь	Э	Ю	Я	–
Код	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33

При моноалфавитной замене каждой букве алфавита открытого текста ставится в соответствие одна буква шифртекста из этого же алфавита.

Пример. Открытый текст: «ШИФРОВАНИЕ_ЗАМЕНОЙ». Подстановка задана табл. 6.2.

6.2. Подстановка

ИТ	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	Ъ	Ы	Ь	Э	Ю	Я	–
ШТ	–	Я	Ю	Э	Ь	Ы	Ъ	Щ	Ш	Ч	Ц	Х	Ф	У	Т	С	Р	П	О	Н	М	Л	К	Й	И	З	Ж	Е	Д	Г	В	Б	А

ИТ – алфавит исходного текста; ШТ – алфавит шифртекста.

Шифртекст: «ИШМРТЮ_УШЫАЩ_ФЫУТЧ».

Основным недостатком рассмотренного метода является сохранение статистических свойств открытого текста (частота повторения букв) в шифртексте.

Общая формула моноалфавитной замены выглядит в виде

$$Y_i = k_1 X_i + k_2 (\mathrm{mod}\ N),$$

где Y_i – i-й символ алфавита; k_1 и k_2 – константы; X_i – i-й символ открытого текста (номер буквы в алфавите); N – длина используемого алфавита.

Шифр, задаваемый формулой

$$Y_i = X_i + k_i (\mathrm{mod}\ N),$$

где k_i – i-ая буква ключа, в качестве которого используются слово или фраза, называется *шифром Вижинера*.

Пример. Открытый текст: «ЗАМЕНА». Ключ: «КЛЮЧ».

Открытый текст	Ключ	Преобразование	Шифр
З	К	$y_1 = 8 + 11 (\mathrm{mod}\ 33) = 19$	Т
А	Л	$y_2 = 1 + 12 (\mathrm{mod}\ 33) = 13$	М
М	Ю	$y_3 = 13 + 31 (\mathrm{mod}\ 33) = 11$	К
Е	Ч	$y_4 = 6 + 24 (\mathrm{mod}\ 33) = 30$	Э
Н	К	$y_5 = 14 + 11 (\mathrm{mod}\ 33) = 25$	Ш
А	Л	$y_6 = 1 + 12 (\mathrm{mod}\ 33) = 13$	М

Шифртекст: «ТМКЭШМ».

Шифры Бофорта используют формулы

$$y_i = k_i - x_i (\mathrm{mod}\ n) \quad \text{и} \quad y_i = x_i - k_i (\mathrm{mod}\ n).$$

Гомофоническая замена одному символу открытого текста ставит в соответствие несколько символов шифртекста. Этот метод применяется для искажения статистических свойств шифртекста.

Пример. Открытый текст: «ЗАМЕНА». Подстановка задана табл. 6.3.

6.3. Гомофоническая подстановка

Алфавит открытого текста	А	Б	…	Е	Ж	З	…	М	Н	…
Алфавит шифртекста	17	23		97	47	76		32	55	
	31	44		51	67	19		28	84	
	48	63		15	33	59		61	34	

Шифртекст: «76 17 32 97 55 31».

Таким образом, при гомофонической замене каждая буква открытого текста заменяется по очереди цифрами соответствующего столбца.

Полиалфавитная подстановка использует несколько алфавитов шифртекста. Пусть используется k алфавитов. Тогда открытый текст

$$X = X_1X_2 \ldots X_kX_{k+1} \ldots X_{2k}X_{2k+1}\ldots$$

заменяется шифртекстом

$$Y = F_1(X_1)F_2(X_2)\ldots F_k(X_k)F_{k+1}(X_{k+1})\ldots F_{2k}(X_{2k})F_{2k+1}(X_{2k+1}),$$

где $F_i(X_j)$ – символ шифртекста алфавита i для символа открытого текста X_j.

Полиграммная замена формируется из одного алфавита с помощью специальных правил. В качестве примера рассмотрим *шифр Плэйфера*. В этом шифре алфавит располагается в матрице. Открытый текст разбивается на пары символов X_iX_{i+1}. Каждая пара символов открытого текста заменяется на пару символов из матрицы следующим образом:

– если символы находятся в одной строке, то каждый из символов пары заменяется на стоящий правее его (за последним символом в строке следует первый);

– если символы находятся в одном столбце, то каждый символ пары заменяется на символ, расположенный ниже его в столбце (за последним нижним символом следует верхний);

– если символы пары находятся в разных строках и столбцах, то они считаются противоположными углами прямоугольника. Символ, находящийся в левом углу, заменяется на символ, стоящий в другом левом углу; замена символа, находящегося в правом углу, осуществляется аналогично;

– если в открытом тексте встречаются два одинаковых символа подряд, то перед шифрованием между ними вставляется специальный символ (например, тире).

Пример. Открытый текст: «ШИФР_ПЛЭЙФЕРА». Матрица алфавита представлена в табл. 6.4.

6.4. Матрица алфавита

А	Х	Б	М	Ц	В
Ч	Г	Н	Ш	Д	О
Е	Щ	,	Ж	У	П
.	З	Ъ	Р	И	Й
С	Ь	К	Э	Т	Л
Ю	Я	_	Ы	Ф	-

Шифртекст: «РДЫИ,-СТ-И.ЖЮБ».

При рассмотрении этих видов шифров становится очевидным, что чем больше длина ключа, тем лучше шифр. Существенного улучшения свойств шифртекста можно достигнуть при использовании шифров с автоключом.

Шифр, в котором сам открытый текст или получающаяся криптограмма используются в качестве «ключа», называется шифром с *автоключом*. Шифрование в этом случае начинается с ключа, называемого первичным, и продолжается с помощью открытого текста или криптограммы, смещенной на длину первичного ключа.

Пример. Открытый текст: «ШИФРОВАНИЕ_ЗАМЕНОЙ». Первичный ключ: «КЛЮЧ».

Шифрование с автоключом при использовании открытого текста представлено в табл. 6.5.

6.5. Шифрование с автоключом при использовании открытого текста

ИТ	Ш	И	Ф	Р	О	В	А	Н	И	Е	_	З	А	М	Е	Н	О	Й

Кл	К	Л	Ю	Ч	Ш	И	Ф	Р	О	В	А	Н	И	Е	_	З	А	М
Код	03	21	19	08	07	12	22	31	24	09	01	22	10	19	06	22	16	23
ШТ	В	Ф	Т	З	Ж	Л	Х	Ю	Ч	И	А	Х	Й	Т	Е	Х	П	Ц

ИТ – алфавит исходного текста; Кл – ключ; ШТ – алфавит шифртекста.

Шифрование с автоключом при использовании криптограммы представлено в табл. 6.6.

6.6. Шифрование с автоключом при использовании криптограммы

ИТ	Ш	И	Ф	Р	О	В	А	Н	И	Е	_	З	А	М	Е	Н	О	Й
Кл	К	Л	Ю	Ч	В	Ф	Т	З	С	Ч	У	Х	Ъ	Э	У	Э	Ы	Й
Код	03	21	19	08	18	24	20	22	27	30	20	30	28	10	26	11	10	20
ШТ	В	Ф	Т	З	С	Ч	У	Х	Ъ	Э	У	Э	Ы	Й	Щ	К	Й	У

ИТ – алфавит исходного текста; Кл – ключ; ШТ – алфавит шифртекста.

6.4.1.2. Методы перестановки

При использовании для шифрования информации методов перестановки символы открытого текста переставляются в соответствии с некоторыми правилами.

Пример. Открытый текст: «ШИФРОВАНИЕ_ПЕРЕСТАНОВКОЙ». Ключ (правило перестановки): группы из 8 букв с порядковыми номерами 1, 2, ..., 8 переставить в порядок 3-8-1-5-2-7-6-4.

Шифртекст: «ФНШОИАВР_СИЕЕЕРПННТВАОКО».

Можно использовать и усложненную перестановку. Для этого открытый текст записывается в матрицу по определенному ключу k_1. Шифртекст образуется при считывании из этой матрицы по ключу k_2.

Пример. Открытый текст: «ШИФРОВАНИЕ_ПЕРЕСТАНОВКОЙ». Матрица из четырех столбцов. Ключи: k_1 {5-3-1-2-4-6}; k_2 {4-2-3-1}. Запись по строкам производится в соответствии с ключом k_1. Чтение по столбцам в соответствии с ключом k_2 (табл. 6.7.).

6.7. Шифрование перестановкой

1	И	Е	_	П
2	Е	Р	Е	С
3	О	В	А	Н
4	Т	А	Н	О
5	Ш	И	Ф	Р
6	В	К	О	Й
k_1/k_2	1	2	3	4

Шифртекст: «ПСНОРЙЕРВАИК_ЕАНФОИЕОТШВ».

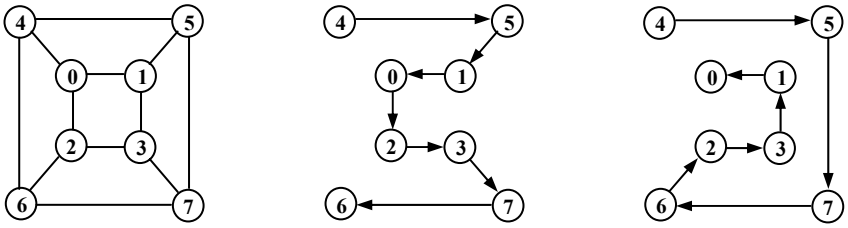
Наиболее сложные перестановки осуществляются по гамильтоновым путям, которых в графе может быть несколько.

Пример. Открытый текст: «ШИФРОВАНИЕ_ПЕРЕСТАНОВКОЙ». Ключ: гамильтонов путь на графе.

Шифртекст: «ШАОНИРФВИЕЕСЕП_РТОВИАОНК»

Необходимо отметить, что для данного графа из восьми вершин можно предложить несколько маршрутов записи открытого текста и несколько гамильтоновых путей для чтения криптограмм (рис. 6.4).

В 1991 г. В.М. Кузьмич предложил схему перестановки, основанной на кубике Рубика. Согласно этой схеме открытый текст записывается в ячейки граней куба по строкам. После осуществления заданного



Общий вид Маршрут 1 Маршрут 2

Рис. 6.4. Гамильтоновы пути на графе

числа заданных поворотов слоев куба считывание шифртекста осуществляется по столбикам. Сложность расшифрования в этом случае определяется количеством ячеек на гранях куба и сложностью выполненных поворотов слоев. Перестановка, основанная на кубике Рубика, получила название объемной (многомерной) перестановки.

В 1992–1994 г.г. идея применения объемной перестановки для шифрования открытого текста получила дальнейшее развитие. Усовершенствованная схема перестановок по принципу кубика Рубика, в которой наряду с открытым текстом перестановке подвергаются и функциональные элементы самого алгоритма шифрования, легла в основу секретной системы «Рубикон». В качестве прообразов пространственных многомерных структур, на основании объемных преобразований которых осуществляются перестановки, в системе «Рубикон» используются трехмерные куб и тэтраэдр. Другой особенностью системы «Рубикон» является генерация уникальной версии алгоритма и ключа криптографических преобразований на основании некоторого секретного параметра (пароля). Это обеспечивает как дополнительные трудности для криптоанализа перехваченных сообщений нарушителем (неопределенность примененного алгоритма), так и возможность априорного задания требуемой криптостойкости. Криптостойкость данной системы определяется длиной ключа, криптостойкостью отдельных функциональных элементов алгоритма криптографических преобразований, а также количеством таких преобразований.

Использование уникальных алгоритма и ключа шифрования для каждого пользователя системы соответствует положению теории К. Шеннона о том, что абсолютно стойкий шифр может быть получен только при использовании "ленты однократного применения", т.е. уникальных параметров при каждом осуществлении шифрования.

6.4.1.3. Методы аналитических преобразований

Шифрование методами аналитических преобразований основано на понятии односторонней функции. Будем говорить, что функция $y = f(x)$ является односторонней, если она за сравнительно небольшое число операций преобразует элемент открытого текста X в элемент шифртекста Y для всех значений X из области определения, а обратная операция (вычисление $X = F^{-1}(Y)$ при известном шифртексте) является вычислительно трудоемкой.

В качестве односторонней функции можно использовать следующие преобразования: умножение матриц; решение задачи об укладке ранца; вычисление значения полинома по модулю; экспоненциальные преобразования и др.

Метод умножения матриц использует преобразование вида $Y = CX$, где $Y = \|y_1, y_2, ..., y_n\|^T$; $C = \|C_{ij}\|$; $X = \|x_1, x_2, ..., x_n\|$.

Пример. Открытый текст: «ПРИКАЗ» («16 17 09 11 01 08» согласно табл. 6.1).

Матрица C: $\begin{vmatrix}1&3&2\\2&1&5\\3&2&1\end{vmatrix}$; $\begin{vmatrix}1&3&2\\2&1&5\\3&2&1\end{vmatrix}\begin{vmatrix}16\\17\\09\end{vmatrix} = \begin{vmatrix}85&94&91\end{vmatrix}$; $\begin{vmatrix}1&3&2\\2&1&5\\3&2&1\end{vmatrix}\begin{vmatrix}11\\01\\08\end{vmatrix} = \begin{vmatrix}30&63&43\end{vmatrix}$.

Шифртекст: «85 94 91 30 63 43».

Задача об укладке ранца формулируется следующим образом.

Задан вектор $C = |c_1, c_2, ..., c_n|$, который используется для шифрования сообщения, каждый символ s_i которого представлен последовательностью из n бит $s_i = |x_1, x_2, ..., x_n|^T$, $X_k \subset \{0, 1\}$. Шифртекст получается как скалярное произведение Cs_i.

Пример. Открытый текст: «ПРИКАЗ» («16 17 09 11 01 08» согласно табл. 6.1). Вектор $C = \{1, 3, 5, 7, 11\}$. Запишем символы открытого текста пятиразрядным двоичным кодом.

П	Р	И	К	А	З
10000	10001	01001	01011	00001	01000

Произведем соответствующие операции:

$y_1 = 1 * 1 = 1;$

$y_2 = 1 * 1 + 1 * 11 = 12;$

$y_3 = 1 * 3 + 1 * 11 = 14;$

$y_4 = 1 * 3 + 1 * 7 + 1 * 11 = 21;$

$y_5 = 1 * 11 = 11;$

$y_6 = 1 * 3 = 3.$

Шифртекст: «01 12 14 21 11 03».

Метод полиномов основан на преобразовании

$$y_i = x_i^n + a_1 * x_i^{(n-1)} + \ldots + a_n * x_i (\mathrm{mod}\ p),$$

где n, a_1, a_2, ..., a_n – целые неотрицательные числа, не превосходящие p, $1 \le x_i, y_i \le p$; p – большое простое число.

Пример. Открытый текст: «ПРИКАЗ». («16 17 09 11 01 08» согласно табл. 6.1.)

Полином:

$y_i = x_i^3 + 2x_i^2 + 3xi + 4(\mathrm{mod}\ 991);$

$y_1 = 16^3 + 2*16^2 + 3*16 + 4(\mathrm{mod}\ 991) = 696;$

$y_2 = 17^3 + 2*17^2 + 3*16 + 4(\mathrm{mod}\ 991) = 591;$

$y_3 = 9^3 + 2*9^2 + 3*9 + 4(\mathrm{mod}\ 991) = 922;$

$y_4 = 11^3 + 2*11^2 + 3*11 + 4(\mathrm{mod}\ 991) = 619;$

$y_5 = 1^3 + 2*1^2 + 3*1 + 4(\mathrm{mod}\ 991) = 10;$

$y_6 = 8^3 + 2*8^2 + 3*8 + 4(\mathrm{mod}\ 991) = 668.$

Шифртекст: «96 591 922 619 010 668».

Экспоненциальный шифр использует преобразование вида

$$y_i = a^{(xi)} (\mathrm{mod}\ p),$$

где x_i – целое, $1 \le x_i \le p - 1$; p – большое простое число; a – целое, $1 \le a \le p$.

Пример. Открытый текст: «ПРИКАЗ» («16 17 09 11 01 08» согласно табл. 6.1); $a = 3$; $p = 991$.

$y_1 = 3^{16}(\mathrm{mod}\ 991) = 43046721\ (\mathrm{mod}\ 991) = 654;$

$y_2 = 3^{17}(\mathrm{mod}\ 991) = 129140163\ (\mathrm{mod}\ 991) = 971;$

$y_3 = 3^{9}(\mathrm{mod}\ 991) = 19683\ (\mathrm{mod}\ 991) = 854;$

$y_4 = 3^{11}(\mathrm{mod}\ 991) = 177147\ (\mathrm{mod}\ 991) = 749;$

$y_5 = 3^{1}\ (\mathrm{mod}\ 991) = 3;$

$y_6 = 3^{8}\ (\mathrm{mod}\ 991) = 6561\ (\mathrm{mod}\ 991) = 615.$

Шифртекст: «654 971 854 749 003 615».

6.4.1.4. Гаммирование (шифрование с помощью датчика псевдослучайных чисел)

Различают два случая: метод конечной гаммы и метод бесконечной гаммы. В качестве конечной гаммы может использоваться фраза, а в качестве бесконечной – последовательность, вырабатываемая датчиком псевдослучайных чисел.

Принцип зашифрования заключается в генерации гаммы шифра с помощью датчика псевдослучайных чисел (ПСЧ) и наложении полученной гаммы на открытые данные обратимым образом (например, при использовании логической операции «исключающее ИЛИ»).

Процесс расшифрования данных сводится к повторной генерации гаммы шифра при известном ключе и наложению такой гаммы на зашифрованные данные. Полученный зашифрованный текст является достаточно трудным для раскрытия в том случае, когда гамма шифра не содержит повторяющихся битовых последовательностей. По сути дела гамма шифра должна изменяться случайным образом для каждого шифруемого слова. Фактически если период гаммы превышает длину всего зашифрованного текста и неизвестна никакая часть исходного текста, то шифр можно раскрыть только прямым перебором (подбором ключа). В этом случае криптостойкость определяется размером ключа.

Чтобы получить линейные последовательности элементов гаммы, длина которых превышает размер шифруемых данных, используются датчики ПСЧ. На основе теории групп было разработано несколько типов таких датчиков. В настоящее время наиболее доступными и эффективными являются *конгруэнтные генераторы* ПСЧ.

Они вырабатывает последовательности псевдослучайных чисел $T(i)$, описываемые соотношением

$$T(i+1) = (A * T(i) + C) \bmod M,$$

где A и C – константы; $T(0)$ – исходная величина, выбранная в качестве порождающего

числа.

Для шифрования данных с помощью датчика ПСЧ может быть выбран ключ любого размера. Например, пусть ключ состоит из набора чисел $X(j)$ размерностью b, где $j = 1, 2, ..., N$. Тогда создаваемую гамму шифра G можно представить как объединение непересекающихся множеств $H(j)$:

$$G = H(1) \cup H(2) \cup \ldots \cup H(N),$$

где $H(j)$ – множество соответствующих j-му сегменту данных и полученных на основе порождающего числа $Y(j)$, определенного как функция от $X(j)$ (например, ПСЧ, полученное на основе $X(j)$).

Разумеется, возможны и другие, более изощренные варианты выбора порождающих чисел для гаммы шифра. Более того, гамму шифра необязательно рассматривать как объединение непересекающихся множеств. Например, гамма шифра может быть представлена в виде

$$G = H(1) (+) H(2) (+) \ldots (+) H(N),$$

где символ (+) обозначает операцию «Исключающее ИЛИ».

Шифрование с помощью датчика ПСЧ является довольно распространенным криптографическим методом, а качество шифра определяется не только и не столько характеристиками датчика, сколько алгоритмом получения гаммы. Хорошие результаты дает метод гаммирования с обратной связью, который заключается в том, что для получения сегмента гаммы используется контрольная сумма определенного участка шифруемых данных.

6.4.1.5. Комбинированные методы

Шифрование комбинированными методами основывается на результатах, полученных К. Шенноном. Наиболее часто применяются такие комбинации, как подстановка и гамма, перестановка и гамма, подстановка и перестановка, гамма и гамма. При составлении комбинированных шифров необходимо проявлять осторожность, так как неправильный выбор составлявших шифров может привести к исходному открытому тексту.

В качестве примера можно привести шифр, предложенный Д. Френдбергом, который комбинирует многоалфавитную подстановку с генератором ПСЧ. Особенность данного алгоритма состоит в том, что при большом объеме шифртекста частотные характеристики символов шифртекста близки к равномерному распределению независимо от содержания открытого текста.

Комбинация методов подстановки и перестановки была применена в 1974 г. фирмой *IBM* при разработке системы ЛЮЦИФЕР.

Система ЛЮЦИФЕР строится на базе блоков подстановки (*S*-блоков) и блоков перестановки (*P*-блоков). Блок подстановки включает линейные и нелинейные преобразования.

Первый преобразователь *S*-блока осуществляет развертку двоичного числа из n разрядов в число по основанию 2^n. Второй преобразователь осуществляет свертку этого числа.

Блок перестановки осуществляет преобразование n разрядного входного числа в n разрядное число.

Входные данные (открытый текст) последовательно проходят через чередующиеся слои 32-разрядных *P*-блоков и 8-разрядных *S*-блоков.

Реализация шифрования данных в системе ЛЮЦИФЕР программными средствами показала низкую производительность, поэтому *P* и *S*-блоки были реализованы аппаратно, что позволило достичь скорости шифрования до 100 Кбайт/с. Опыт, полученный при разработке и эксплуатации системы, дал возможность создать стандарт шифрования данных *DES*.

DES (*Data Encryption Standard*) является одним из наиболее распространенных криптографических стандартов на шифрование данных, применяемых в США. Первоначально метод, лежащий в основе данного стандарта, был разработан фирмой *IBM* для своих целей. Он был проверен Агентством Национальной Безопасности США, которое не обнаружило в нем статистических или математических изъянов. Это означало, что дешифрование данных, защищенных с помощью *DES*, не могло быть выполнено статистическими (например, с помощью частотного словаря) или математическими («прокручиванием» в обратном направлении) методами.

После этого метод фирмы *IBM* был принят в качестве федерального стандарта. Стандарт *DES* используется федеральными департаментами и агентствами для защиты всех достаточно важных данных в компьютерах (исключая некоторые данные, методы защиты которых определяются специальными актами). Его применяют многие негосу-

дарственные институты, в том числе большинство банков и служб обращения денег. Оговоренный в стандарте алгоритм криптографической защиты данных опубликован для того, чтобы большинство пользователей могли использовать проверенный и апробированный алгоритм с хорошей криптостойкостью. Однако, с одной стороны, публикация алгоритма нежелательна, поскольку может привести к попыткам дешифрования закрытой информации, но, с другой стороны, это не столь существенно поскольку стандартный алгоритм шифрования данных должен обладать такими характеристиками, чтобы его опубликование не сказалось на его криптостойкости.

DES имеет блоки по 64 бит и основан на 16 кратной перестановке данных, также для шифрования использует ключ в 56 бит. Существует несколько режимов *DES*: *Electronic Code Book* (*ECB*) и *Cipher Block Chaining* (*CBC*).

56 бит – это 8 семибитовых ASCII символов, т.е. пароль не может быть больше чем 8 букв. Если вдобавок использовать только буквы и цифры, то количество возможных вариантов будет существенно меньше максимально возможных 2^{56}.

Суть данного алгоритма состоит в следующем (рис. 6.5).

Входной блок данных делится пополам на левую (L_0) и правую (R_0) части. После этого формируется выходной массив так, что его левая часть L_1 представлена правой частью R_0 входного, а правая R_1 формируется как сумма L_0 и R_0 операций XOR. Далее, выходной массив шифруется перестановкой с заменой. Можно убедиться, что все проведенные операции могут быть обращены и расшифровывание осуществляется за число операций, линейно зависящее от размера блока. После нескольких таких взбиваний можно считать, что каждый бит выходного блока шифровки может зависеть от каждого бита сообщения.

ГОСТ 28147–89 – *отечественный стандарт на шифрование данных*. В нашей стране установлен единый алгоритм криптографического преобразования данных для систем обработки информации в сетях ЭВМ, отдельных вычислительных комплексах и ЭВМ, который определяется указанным ГОСТом.

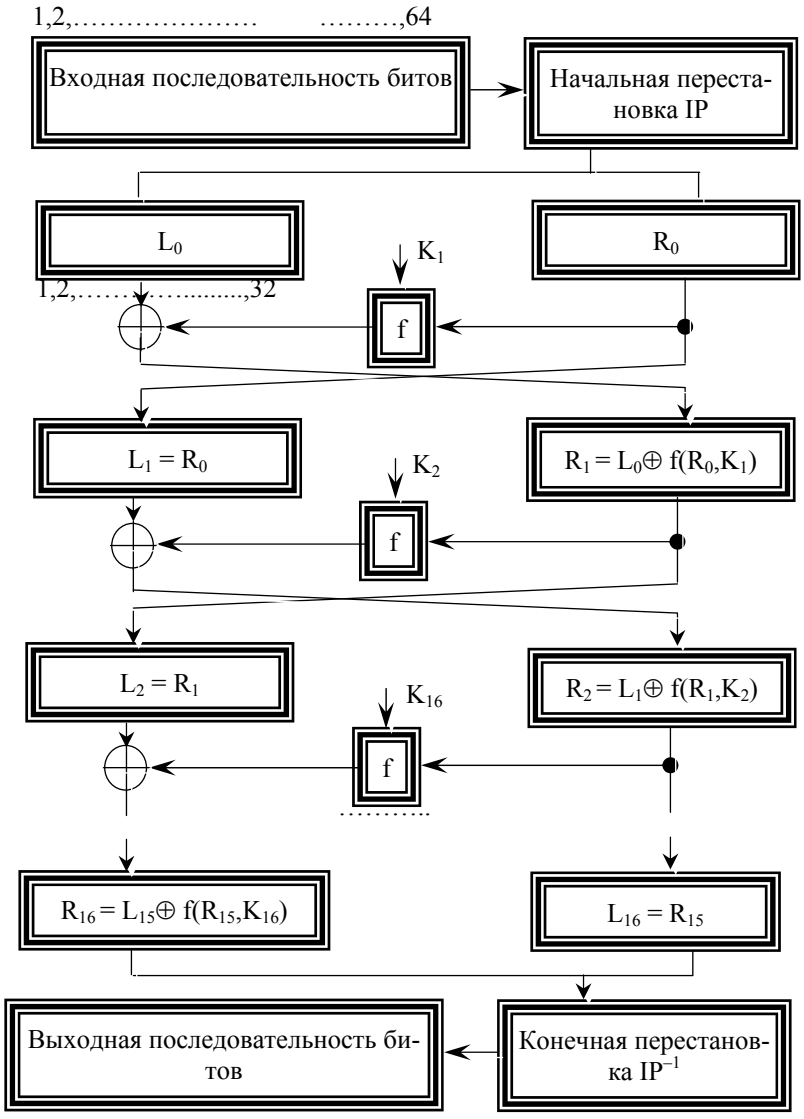


Рис. 6.5. Схема алгоритма шифрования DES

Этот алгоритм криптографического преобразования данных предназначен для аппаратной или программной реализации, удовлетворяет криптографическим требованиям и не накладывает ограничения на степень секретности защищаемой информации.

Из-за сложности алгоритма приведем только основные его идеи. Чтобы получить подробные спецификации алгоритма криптографического преобразования, следует обратиться к оригиналу.

При описании алгоритма используются следующие обозначения. Если L и R – это последовательности бит, то LR будет обозначать конкатенацию последовательностей L и R. Под конкатенацией последовательностей L и R понимается последовательность бит, размерность которой равна сумме размерностей L и R. В этой последовательности биты последовательности R следуют за битами последовательности L. Конкатенация битовых строк является ассоциативной, т.е. запись ABCDE обозначает, что за битами последовательности A следуют биты последовательности B, затем C и т.д.

Символом (+) будет обозначаться операция побитового сложения по модулю 2, символом [+] – операция сложения по модулю 2^{32} двух 32-разрядных чисел, символом {+} – операция сложения по модулю $2^{32} - 1$ двух 32 разрядных чисел.

Алгоритм криптографического преобразования предусматривает несколько режимов работы. Но в любом случае для шифровки данных используется ключ, который имеет разрядность 256 бит и представляется в виде восьми 32-разрядных чисел $X(i)$. Если обозначить ключ через W, то

$$W = X(7)X(6)X(5)X(4)X(3)X(2)X(1)X(0).$$

Расшифрование выполняется по тому же ключу, что и зашифрование, но этот процесс является инверсией процесса зашифрования данных.

Общая структурная схема ГОСТ 28147–89 приведена на рис. 6.6.

Стандартом определены следующие алгоритмы криптографического преобразования информации: *замена, гаммирование, гаммирование с обратной связью, выработка имитовставки*.

Первый и самый простой режим – замена. Открытые данные, подлежащие зашифрованию, разбивают на блоки по 64 бит в каждом, которые можно обозначить $T(j)$.

Очередная последовательность бит $T(j)$ разделяется на две последовательности $B(0)$ (левые или старшие биты) и $A(0)$ (правые или младшие биты), каждая из которых содержит 32 бита. Затем выполняется итеративный процесс шифрования, который описывается следующими формулами:

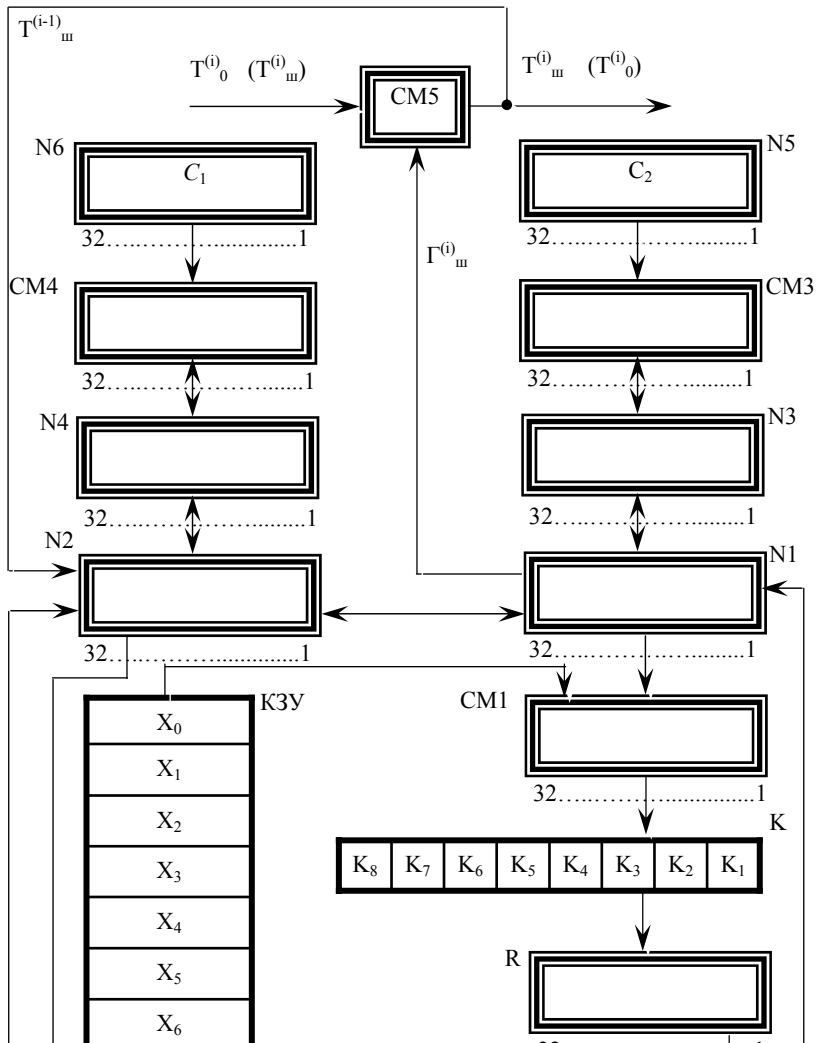


Рис. 6.6. Общая структурная схема алгоритма ГОСТ 28147–89

$$\begin{cases} A(i) = f(A(i-1)[+]X(j)(+)B(i-1); \\ B(i) = A(i-1), \ \text{если} \ \ i = 1,\ 2,\ ...,\ 24,\ j = (i-1) \bmod 8; \end{cases}$$

$$\begin{cases} A(i) = f(A(i-1)[+]X(j)(+)B(i-1)); \\ B(i) = A(i-1), \ \text{если} \ \ i = 25,\ 26,\ ...,\ 31,\ j = 32 - i; \end{cases}$$

$$\begin{cases} A(32) = A(31); \\ B(32) = f(A(31)[+]X(0))(+)B(31), \ \text{если} \ \ i = 32. \end{cases}$$

Здесь i обозначает номер итерации (i = 1, 2, ..., 32). Функция f называется функцией шифрования. Ее аргументом является сумма по модулю 2^{32} числа A(i), полученного на предыдущем шаге итерации, и числа $X(j)$ ключа (размерность каждого из этих чисел равна 32 знакам).

Функция шифрования включает две операции над полученной 32-разрядной суммой. Первая операция называется подстановкой К. Блок подстановки К состоит из восьми узлов замены К(1)...К(8) с памятью 64 бит каждый. Поступающий на блок подстановки 32-разрядный вектор разбивается на восемь последовательно идущих 4-разрядных векторов, каждый из которых преобразуется в 4-разрядный вектор соответствующим узлом замены, представляющим собой таблицу из шестнадцати целых чисел в диапазоне 0...15.

Входной вектор определяет адрес строки в таблице, число из которой является выходным вектором. Затем 4-разрядные выходные векторы последовательно объединяются в 32-разрядный вектор. Таблицы блока подстановки К содержат ключевые элементы, общие для сети ЭВМ и редко изменяемые.

Вторая операция – циклический сдвиг влево 32-разрядного вектора, полученного в результате подстановки К. 64-разрядный блок зашифрованных данных $T_ш$ представляется в виде

$$T_ш = A(32)B(32).$$

Остальные блоки открытых данных в режиме простой замены зашифровываются аналогично.

Следует иметь в виду, что режим простой замены допустимо использовать для шифрования данных только в ограниченных случаях. К этим случаям относится выработка ключа и зашифрование его с обеспечением имитозащиты для передачи по каналам связи иди хранения в памяти ЭВМ.

Следующий режим шифрования называется режимом гаммироваиия. Отрытые данные, разбитые на 64-разрядные блоки T(i) (i = 1, 2, …, m, где m определяется объемом шифруемых данных), зашифровываются в режиме гаммирования путем поразрядного сложения по модулю 2 с гаммой шифра $Γ_ш$, которая вырабатывается блоками по 64 бит, т.е.

$$Γ_ш = (Γ(1), Γ(2), …, Γ(i), …, Γ(m)).$$

Число двоичных разрядов в блоке T(m) может быть меньше 64, при этом неиспользованная для шифрования часть гаммы из блока Γ(m) отбрасывается.

Уравнение зашифрования данных в режиме гаммирования может быть представлено в следующем виде:

$$Ш(i) = A(Y(i-1)\ [+]\ C2,\ Z(i-1)\ \{+\}\ C(1)\ (+)\ T(i) = Γ(i)\ (+)\ T(i).$$

В этом уравнении Ш(i) обозначает 64-разрядный блок зашифрованного текста, А – функцию шифрования в режиме простой замены (аргументами этой функции являются два 32-разрядных числа), С1 и С2 – константы, заданные в ГОСТ 28147–89. Величины Y(i) и Z(i) определяются итерационно по мере формирования гаммы, следующим образом:

$$(Y(0), Z(0)) = A(S),$$

где S – 64-разрядная двоичная последовательность (синхропосылка);

$$(Y(i), Z(i)) = (Y(i-1)\ [+]\ C2,\ Z(i-1)\ \{+\}\ C(1) \text{ для } \ i = 1, 2, …, m.$$

Расшифрование данных возможно только при наличии синхропосылки, которая не является секретным элементом шифра и может храниться в памяти ЭВМ или передаваться по каналам связи вместе с зашифрованными данными.

Режим гаммирования с обратной связью очень похож на режим гаммирования. Как и в режиме гаммирования, отрытые данные, разбитые на 64-разрядные блоки T(1) (I = 1, 2, …, m, где m определяется объемом шифруемых данных), зашифровываются путем поразрядного сложения по модулю 2 с гаммой шифра Гш, которая вырабатывается блоками по 64 бит:

$$Γ_ш = (Γ(1), Γ(2), …, Γ(i), ..., Γ(m)).$$

Число двоичных разрядов в блоке T(m) может быть меньше 64, при этом неисполь-

зованная для шифрования часть гаммы из блока Г(m) отбрасывается.

Уравнение зашифрования данных в режиме гаммирования с обратной связью может быть представлено в следующем виде:

Ш(1) = A(S) (+) T(1) = Г(1) (+) T(1),

Ш(i) = A(Ш(i – 1)) (+) T(i) = Г(i) (+) T(I) для i = 2, 3, …, m.

Здесь Ш(i) обозначает 64-разрядный блок зашифрованного текста, А – функцию шифрования в режиме простой замены. Аргументом функции на первом шаге итеративного алгоритма является 64-разрядная синхропосылка, а на всех последующих – предыдущий блок зашифрованных данных Ш(i – 1).

В ГОСТ 28147–89 определяется процесс выработки имитовставки, который единообразен для любого из режимов шифрования данных. Имитовставка – это блок из *p* бит (имитовставка И*p*), который вырабатывается либо перед шифрованием всего сообщения, либо параллельно с шифрованием по блокам. Первые блоки открытых данных, которые участвуют в выработке имитовставки, могут содержать служебную информацию (например, адресную часть, время, синхропосылку) и не зашифровываться. Значение параметра *p* (число двоичных разрядов в имитовставке) определяется криптографическими требованиями с учетом того, что вероятность навязывания ложных помех равна $1/2^p$.

Для получения имитовставки отрытые данные представляются в виде 64-разрядных блоков T(i) (i = 1, 2, …, m, где m определяется объемом шифруемых данных). Первый блок открытых данных T(1) подвергается преобразованию, соответствующему первым 16 циклам алгоритма зашифрования в режиме простой замены. Причем в качестве ключа для выработки имитовставки используется ключ, по которому шифруются данные.

Полученное после 16 циклов работы 64-разрядное число суммируется по модулю 2 со вторым блоком открытых данных T(2). Результат суммирования снова подвергается преобразованию, соответствующему первым 16 циклам алгоритма зашифрования в режиме простой замены.

Полученное 64-разрядное число суммируется по модулю 2 с третьим блоком открытых данных T(3) и т.д. Последний блок T(m), при необходимости дополненный до полного 64-разрядного блока нулями, суммируется по модулю 2 с результатом работы на шаге m – 1, после чего зашифровывается в режиме простой замены по первым 16 циклам работы алгоритма. Из полученного 64-разрядного числа выбирается отрезок И*p* длиной *p* бит.

Имитовставка И*p* передается по каналу связи или в память ЭВМ после зашифрованных данных. Поступившие зашифрованные данные расшифровываются и из полученных блоков открытых данных T(i) вырабатывается имитовставка И*p*, которая затем сравнивается с имитовставкой И*p*, полученной из канала связи или из памяти ЭВМ. В случае несовпадения имитовставок все расшифрованные данные считают ложными.

6.4.2. АСИММЕТРИЧНЫЕ МЕТОДЫ ШИФРОВАНИЯ

Наиболее перспективными системами криптографической защиты данных являются системы, основанные на асимметричных методах шифрования. В таких системах для зашифрования данных используется один ключ, а для расшифрования другой. Первый

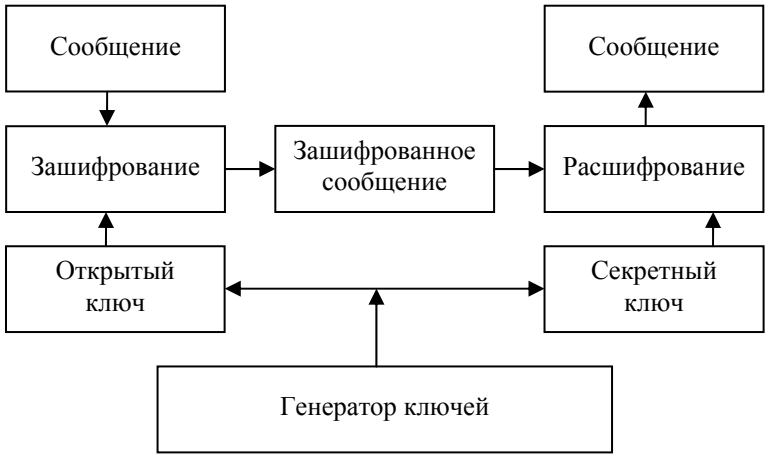


Рис. 6.7. Использование асимметричного метода шифрования

ключ не является секретным и может быть опубликован для использования всеми пользователями системы, которые зашифровывают данные. Расшифрование данных с помощью известного ключа невозможно. Для расшифрования данных получатель зашифрованной информации использует второй ключ, который является секретным. Разумеется, ключ расшифрования не может быть определен из ключа зашифрования.

Применение таких шифров стало возможным благодаря К. Шеннону, предложив-

шему строить шифр таким способом, чтобы его раскрытие было эквивалентно решению математической задачи, требующей выполнения объемов вычислений, превосходящих возможности современных ЭВМ (например, операции с большими простыми числами и их произведениями).

Принцип применения асимметричного шифрования показан на рис. 6.7. Рассмотрим наиболее распространенные алгоритмы асимметричного шифрования.

6.4.2.1. Алгоритм *RSA*

В настоящее время наиболее развитым методом криптографической защиты информации с известным ключом является *RSA*, названный так по начальным буквам фамилий ее изобретателей (*Rivest*, *Shamir* и *Adleman*). Перед тем как приступить к изложению концепции метода *RSA*, необходимо определить некоторые термины.

Под *простым числом* будем понимать такое число, которое делится только на 1 и на само себя. Взаимно простыми числами будем называть такие числа, которые не имеют ни одного общего делителя, кроме 1.

Под *результатом операции* $i \bmod j$ будем считать остаток от целочисленного деления i на j. Чтобы использовать алгоритм *RSA*, надо сначала сгенерировать открытый и секретный ключи, выполнив следующие шаги.

Выберем два очень больших простых числа p и q,

Определим n как результат умножения p на q ($n = pq$).

Выберем большое случайное число, которое назовем d. Это число должно быть взаимно простым с m результатом умножения $(p - 1)(q - 1)$.

Определим такое число e, для которого является истинным следующее соотношение $(e\,d) \bmod (m) = 1$ или $e = (1 \bmod (m))/d$.

Открытым ключом будут числа e и n, а секретным ключом – числа d и n.

Теперь, чтобы зашифровать данные по известному ключу $\{e, n\}$, необходимо сделать следующее:

– разбить шифруемый текст на блоки, каждый из которых может быть представлен в виде числа $M(i) = 0, 1, \ldots, n - 1$;

– зашифровать текст, рассматриваемый как последовательность чисел $M(i)$ по формуле $C(i) = (M(i)^e) \bmod n$.

Чтобы расшифровать данные, используя секретный ключ $\{d, n\}$, необходимо выполнить следующие вычисления: $M(i) = (C(i)^d) \bmod n$. В результате получится множество чисел $M(i)$, которые представляют собой исходный текст.

Пример. Применим метод *RSA* для шифрования сообщения «ГАЗ». Для простоты будем использовать очень маленькие числа (на практике используются намного большие числа).

Выберем $p = 3$ и $q = 11$.

Определим $n = 3 \cdot 11 = 33$.

Найдем $(p - 1)(q - 1) = 20$. Следовательно, в качестве d выберем любое число, которое является взаимно простым с 20, например $d = 3$.

Выберем число e. В качестве такого числа может быть взято любое число, для которого удовлетворяется соотношение $(e \times 3) \bmod 20 = 1$, например 7.

Представим шифруемое сообщение как последовательность целых чисел в диапазоне 0…32. Пусть буква *А* изображается числом 1, буква Г – числом 4, а буква З – числом 9. Тогда сообщение можно представить в виде последовательности чисел 4 1 9. Зашифруем сообщение, используя ключ {7, 33}:

$C_1 = (4^7) \bmod 33 = 16384 \bmod 33 = 16$,

$C_2 = (1^7) \bmod 33 = 1 \bmod 33 = 1$,

$C_3 = (9^7) \bmod 33 = 4782969 \bmod 33 = 15$.

Шифртекст: «16 1 15».

Попытаемся расшифровать сообщение {16, 1, 15}, полученное в результате зашифрования по известному ключу, на основе секретного ключа {3, 33}:

$M_1 = (16^3) \bmod 33 = 4096 \bmod 33 = 4$,

$M_2 = (1^3) \bmod 33 = 1 \bmod 33 = 1$,

$M_3 = (15^3) \bmod 33 = 3375 \bmod 33 = 9$.

Таким образом, в результате расшифрования сообщения получено исходное сообщение «ГАЗ».

Криптостойкость алгоритма *RSA* основывается на предположении, что исключительно трудно определить секретный ключ по известному, поскольку для этого необходимо решить задачу о существовании делителей целого числа. Данная задача является *NP*-полной и, как следствие этого факта, не допускает в настоящее время эффективного (полиномиального) решения. Более того, сам вопрос существования эффективных алгоритмов решения *NP*-полных задач является до настоящего времени открытым. В связи с этим для чисел, состоящих из 200 цифр (а именно такие числа рекомендуется использо-

вать), традиционные методы требуют выполнения огромного числа операций (около 10^{23}).

6.4.2.2. Алгоритм Эль-Гамаля

Система Эль-Гамаля – это криптосистема с открытым ключом, основанная на проблеме вычисления логарифма. Данный алгоритм используется как для шифрования, так и для цифровой подписи.

Множество параметров системы включает простое число p и целое g, степени которого по модулю p порождают большое число элементов Z_p. У пользователя A есть секретный ключ a и открытый ключ y, вычисляемый

$$y = g^a \bmod p.$$

Предположим, что пользователь B желает послать сообщение m пользователю A. Сначала пользователь B выбирает случайное число k, меньшее p. Затем он вычисляет

$$y_1 = g^k \bmod p \text{ и } y_2 = m(+)(y^k(\bmod p)),$$

где (+) обозначает побитовое «исключающее ИЛИ». Пользователь B посылает пользователю A пару (y_1, y_2). После получения шифрованного текста пользователь A вычисляет

$$m = (y_1{}^a \bmod p)(+)y_2.$$

Иногда операция побитового «исключающего ИЛИ» может быть заменена на умножение по модулю p. Уравнение расшифрования в этом случае принимает вид

$$m = y_2/y_1{}^k \bmod p.$$

Существенным недостатком асимметричных методов является их низкое быстро-

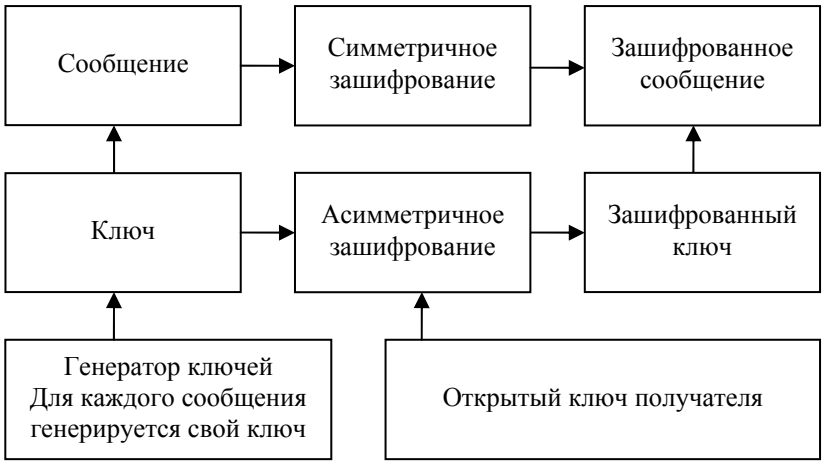


Рис. 6.8. Эффективное шифрование сообщения

действие, поэтому их приходится сочетать с симметричными (асимметричные методы на 3–4 порядка медленнее симметричных). Так, для решения задачи рассылки ключей сначала сообщение симметрично шифруют случайным ключом, затем этот ключ шифруют открытым асимметричным ключом получателя, после чего сообщение и ключ отправляются по сети.

Эффективное шифрование, реализованное путем сочетания симметричного и асимметричного методов, иллюстрирует рис. 6.8, а рис. 6.9 – расшифрование эффективно зашифрованного сообщения.

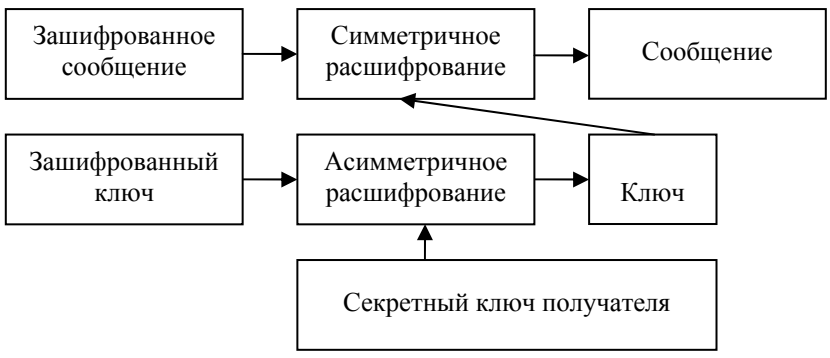


Рис. 6.9. Расшифрование эффективно зашифрованного сообщения

Применение асимметричных методов позволило решить важную задачу совместной выработки секретных ключей, обслуживающих сеанс взаимодействия, при изначальном отсутствии общих секретов. Для этого используется алгоритм Диффи-Хелмана.

Определенное распространение получила разновидность симметричного шифрования, основанная на использовании составных ключей. Идея состоит в том, что секретный ключ делится на две части, хранящиеся отдельно. Каждая часть сама по себе не позволяет выполнить расшифрование. Если у правоохранительных органов появляются подозрения относительно лица, использующего некоторый ключ, они могут в установленном порядке получить половинки ключа и дальше действовать обычным для симметричной расшифровки образом.

Составные ключи – отличный пример следования принципу разделения обязанностей. Они позволяют сочетать право граждан на тайну с возможностью эффективно следить за нарушителями закона, хотя, здесь очень много тонкостей и технического, и юридического плана.

Криптографические методы позволяют надежно контролировать целостность информации, определять ее подлинность, гарантировать невозможность отказаться от совершенных действий. В отличие от традиционных методов контрольного суммирования, способных противостоять только случайным ошибкам, криптографическая контрольная сумма (имитовставка), вычисленная с применением секретного ключа, практически исключает все возможности незаметным образом изменить данные.

6.4.2.3. Электронная цифровая подпись

В основе криптографического контроля целостности лежат два понятия: *хэш-функция*; *электронная цифровая подпись* (ЭЦП).

Хэш-функция – это труднообратимое преобразование данных (односторонняя функция), реализуемое, как правило, средствами симметричного шифрования со связыванием блоков. Результат шифрования последнего блока (зависящий от всех предыдущих) и служит результатом хэш-функции.

Пусть имеются данные, целостность которых должна быть проверена, хэш-функция и ранее вычисленный результат ее применения к исходным данным (дайджест). Хэш-функцию обозначим через h, исходные данные – через T, проверяемые данные – через T'. Контроль целостности данных сводится к проверке равенства $h(T) = h(T')$. Если оно выполняется, считается, что $T = T'$. Совпадение дайджестов для различных данных называется *коллизией.* В принципе коллизии возможны (так как мощность множества дайджестов меньше множества хэшируемых данных), однако, исходя из определения хэш-функции, специально организовать коллизию за приемлемое время невозможно.

Асимметричные методы позволяют реализовать так называемую *электронную цифровую подпись*, или электронное заверение сообщения. Идея состоит в том, что отправитель посылает два экземпляра сообщения – открытое и дешифрованное его секретным ключом (естественно, дешифровка незашифрованного сообщения на самом деле есть форма шифрования). Получатель может зашифровать с помощью открытого ключа отправителя дешифрованный экземпляр и сравнить с открытым. Если они совпадут, личность и подпись отправителя можно считать установленными.

Пусть $E(T)$ обозначает результат шифрования текста T с помощью открытого ключа, а $D(T)$ – результат дешифровки текста T с помощью секретного ключа. Чтобы асимметричный метод мог применяться для реализации электронной подписи, необходимо выполнение тождества

$$E(D(T)) = D(E(T)) = T.$$

Проиллюстрируем рис. 6.10 процедуру эффективной генерации электронной подписи, состоящую в шифровании преобразованием D дайджеста $h(T)$, а проверка эффективно сгенерированной электронной подписи может быть реализована способом, изображенным на рис. 6.11.

Из равенства $E(S') = h(T)$ следует, $S' = D(h(T'))$. Следовательно, ЭЦП защищает целостность сообщения, удостоверяет личность отправителя и служит основой неотказуемости.

Два российских стандарта, «Процедуры выработки и проверки электронной цифровой подписи на базе асимметричного криптографического алгоритма» и «Функция хэширования», объединенные общим заголовком «Информационная технология. Криптографическая защита информации», регламентируют вычисление дайджеста и реализацию электронной подписи.

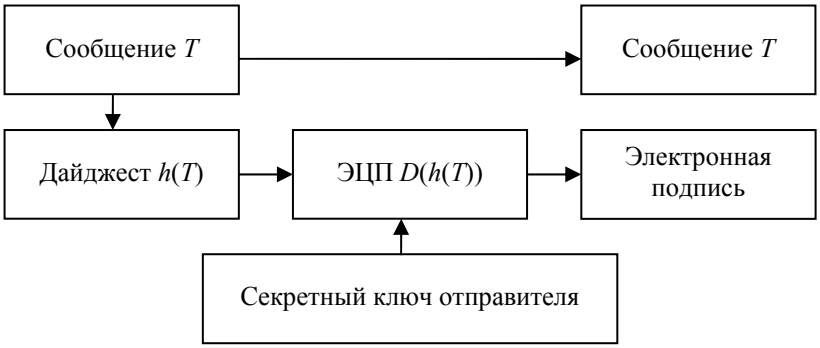


Рис. 6.10. Выработка электронной цифровой подписи

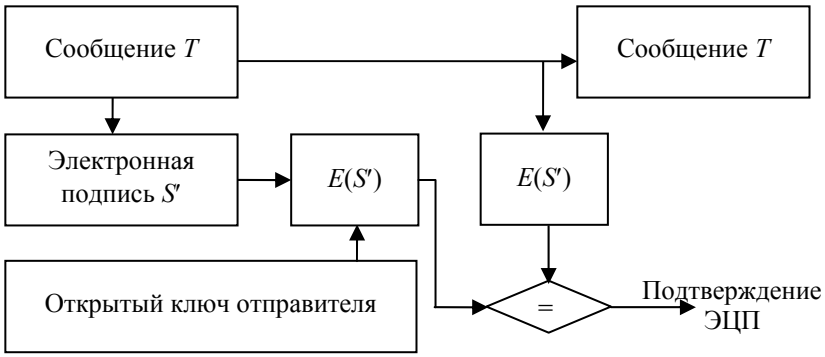


Рис. 6.11. Проверка электронной цифровой подписи

В сентябре 2001 г. утвержден, а с 1 июля 2002 г. вступил в силу новый стандарт ЭЦП – ГОСТ Р 34.10–2001.

Для контроля целостности последовательности сообщений (т.е. защиты от кражи, дублирования и переупорядочения сообщений) применяют временные штампы и нумерацию элементов последовательности, при этом штампы и номера включают в подписываемый текст.

Обратим внимание на то, что при использовании асимметричных методов шифрования (в частности ЭЦП) необходимо иметь гарантию подлинности пары (имя, открытый ключ) адресата. Для решения этой задачи в спецификациях X.509 вводятся понятия цифрового сертификата и сертификационного центра. Сертификационный центр – это компонент глобальной службы каталогов, отвечающий за управление криптографическими ключами пользователей, заверяющий подлинность пары имя, открытый ключ адресата своей подписью.

Цифровые сертификаты в формате X.509 стали не только формальным, но и фактическим стандартом, поддерживаемым многочисленными сертификационными центрами.

Отметим, что услуги, характерные для асимметричного шифрования, можно реализовать и с помощью симметричных методов, если имеется надежная третья сторона, знающая секретные ключи своих клиентов. Эта идея положена, например, в основу сервера аутентификации *Kerberos*.

6.4.3. СРАВНЕНИЕ КРИПТОГРАФИЧЕСКИХ МЕТОДОВ

Метод шифрования с использованием датчика ПСЧ наиболее ча-сто используется в программной реализации системы криптографической защиты данных. Это объясняется сочетанием в нем: простоты программирования и высокую криптостойкость. При этом системы, основанные на методе шифрования с использованием датчика ПСЧ, позволяют зашифровать в секунду от нескольких десятков до сотен кБ данных (для ПЭВМ).

Однако этот алгоритм шифрования данных чувствителен к простым воздействиям и перед его применением должен быть подвергнут всестороннему математическому, статистическому и криптографическому анализам. Иначе, результаты могут быть катастрофическими.

Основными преимуществами метода *DES* являются, по утверждениям Национального Бюро Стандартов США:

– высокий уровень защиты данных против дешифрования и возможной модификации данных;

– простота в понимании;

– высокая степень сложности, которая делает его раскрытие дороже получаемой при этом прибыли;

– метод защиты основывается на ключе и не зависит ни от какой «секретности» алгоритма;

– экономичен в реализации и эффективен в быстродействии.

DES обладает и рядом недостатков.

Самым существенным недостатком *DES* признается размер ключа, который считается слишком малым. Стандарт в настоящем виде не является неуязвимым, хотя и очень труден для раскрытия. Для дешифрования информации методом подбора ключей достаточно выполнить 2^{56} операций расшифрования и хотя в настоящее время нет аппаратуры, которая могла бы выполнить в обозримый период времени подобные вычисления, никто не гарантирует ее появление в будущем. Некоторые специалисты предлагают простую модификацию (Тройной *DES*) для устранения этого недостатка: исходный текст P зашифровывается сначала по ключу $K1$, затем по ключу $K2$ и, наконец, по ключу $K3$. В результате время, требующееся для дешифрования, возрастает до 2^{128} операций (приблизительно, до 10^{34} операций).

Метод *DES* может быть реализован и программно. В зависимости от быстродействия и типа процессора персонального компьютера программная система, шифруем данные с использованием метода *DES*, может обрабатывать от нескольких килобайт до десятков кБ данных в секунду. В то же время необходимо отметить, базовый алгоритм все же рассчитан на реализации электронных устройствах специального назначения.

Алгоритмы криптографического преобразования, являющиеся отечественным стандартом и определяемый ГОСТ 28147–89, свободен от недостатка стандарта *DES* и в то же время обладает всеми его преимуществами. Кроме того, в стандарт уже заложен метод, с помощью которого можно зафиксировать необнаруженную случайную или умышленную модификацию зашифрованной информации.

Однако у алгоритма есть очень существенный недостаток, который заключается в том, что его программная реализация очень сложна и практически лишена всякого смысла из-за крайне низкого быстродействия. По оценкам, за одну секунду на персональном компьютере может быть обработано всего лишь несколько десятков (максимально сотен) байт данных, а подобная производительность вряд ли удовлетворит кого-либо из пользователей. Хотя сейчас уже разработанные программные средства, реализующие данный алгоритм криптографического преобразования данных, которые демонстрируют приемлемую производительность.

Теперь остановимся на методе *RSA*. Он является очень перспективным, поскольку для зашифрования информации не требует передачи ключа другим пользователям. Это выгодно отличает его от всех вышеописанных методов криптографической зашиты данных. Но в настоящее время по этому методу относятся с подозрительностью, поскольку в ходе дальнейшего развития может быть найден эффективный алгоритм определения делителей целых чисел, в результате чего метод шифрования станет абсолютно не защищенным. Кроме того, *не существует строго доказательства*, что не существует другого способа определения секретного ключа по известному, кроме как определения делителя целых чисел.

В остальном метод *RSA* обладает только достоинствами. К числу этих достоинств следует отнести очень высокую криптостойкость, довольно простую программную и аппаратную реализации. Правда, следует заметить что использование этого метода для криптографической защиты данных неразрывно связано с очень высоким уровнем развития техники.

6.4.4. МЕТОДЫ КОДИРОВАНИЯ

Как уже отмечалось, под кодированием понимается замена элементов открытого текста (букв, слов, фраз и т.п.) кодами. Различают символьное и смысловое кодирование.

При *символьном кодировании* каждый знак алфавита открытого текста заменяется соответствующим символом. Примером символьного кодирования служит азбука Морзе, а также методы шифрования заменой и перестановкой. Рассмотрим метод символьного кодирования, который использует предыдущие символы открытого текста. Этот метод, называемый методом стопки книг, был предложен Б.Я. Рябко.

Предположим, что нужно передать сообщение X из алфавита A, в котором буквы алфавита отождествлены с числами 1, 2, …, L, где L – число элементов алфавита A. Каждой букве алфавита соответствует код k_i, $1 = 1…L$. При появлении в сообщении X очередной буквы x_j ее код представляется кодом номера позиции j, занимаемой в данный момент буквой x_j в списке. Это дает возможность на приемном конце по коду номера позиции j определить букву x_j. После кодирования буквы x_j одновременно на приемном и передающих концах перемещают букву x_j в начало списка, увеличивая тем самым на единицу номера букв, стоявших на позициях от 1 до $j – 1$. Номера букв, стоявших на позициях от $j + 1$ до L, остаются без изменений. В результате кодирования открытого текста в начале списка будут находиться буквы, которые наиболее часто встречались в открытом тексте.

Интересный метод кодирования в 1992 г. предложил С.П. Савчук. В отличие от метода стопки книг перемещению подвергается список кодов. Пусть алфавит $A = \{a_1, a_2, ..., a_n\}$. Данному порядку расположения букв соответствует начальный список кодов $K_0 = \{k_1, k_2, …, k_n\}$. При появлении в кодируемом сообщении буквы a_i в качестве кода выбирается соответствующий ее местоположению код k_i. После этого осуществляется сдвиг списка кодов:

$$\{k_1, k_2, ..., k_i, ..., k_n\} \rightarrow \{k_2, k_3, ..., k_n, k_1\}.$$

Таким образом, список кодов образует замкнутое кольцо.

Смысловое кодирование – это кодирование, в котором в качестве исходного алфавита используются не только отдельные символы (буквы), но и слова и даже наиболее часто встречающиеся фразы.

Рассмотрим пример одноалфавитного и многоалфавитного смыслового кодирования.

Пример. Открытый текст: «19.9.1992 ГОДА».

6.8. Таблица кодирования

Элементы открытого текста	Коды
1	089 146 214 417
2	187 226 045 361
9	289 023 194 635
ГОД	031 155 217 473
.	786 432 319 157

Закодированное сообщение при одноалфавитном кодировании:
«089 289 786 289 786 089 289 289 187 031».

Закодированное сообщение при многоалфавитном кодировании:
«089 289 786 023 432 146 194 635 187 031» (при многоалфавитном кодировании одинаковые символы заменяются кодами из следующего столбца).

Среди различных кодов, применяемых для кодирования естественных языков, особый интерес вызывает код Хаффмена, который позволяет сжимать открытый текст. Суть его состоит в присваивании наиболее часто встречающимся буквам наиболее коротких кодов.

Строка двоичных символов кодов Хаффмена единственным образом разлагается на коды символов (такие коды называются префиксными).

Пример. Закодированное кодом Хаффмена сообщение имеет вид: «01101000100000010101111000100000».

Пользуясь деревом для английского языка, получаем 0110 = *S*.

Далее снова начинаем движение из вершины: 100 = *E*; 01000 = *C*;
00010 = *U*; 1011 = *R*; 1010 = *I*; 001 = *T*; 00000 = *Y*.

Открытый текст: «SECURITY».

6.4.5. ДРУГИЕ МЕТОДЫ

Широкое применение персональная ЭВМ (ПЭВМ) сделало актуальной задачу защиты хранящихся данных (файлов). Для защиты файлов могут быть применены рассмотренные методы шифрования и кодирования.

Специфика применения ПЭВМ позволяет реализовать дополнительные методы кодирования для надежного закрытия содержимого файлов. Примером такого кодирования является метод рассечения-разнесения, в соответствии с которым содержимое одного файла разбивается на блоки, которые разносятся по нескольким файлам. Каждый такой файл не несет никакой информации, а сбор данных в единое целое осуществляется простой программой.

Пример. Блок (файл открытого текста) начинается словами: «МЕТОД_РАССЕЧЕНИЯ-РАЗНЕСЕНИЯ».

Для рассечения блока открытого текста на 8 частей запишем открытый текст в следующем виде (табл. 6.9).

6.9. Рассечения открытого текста

	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	М	Е	Т	О	С	С	Е	Ч	-	Р	А	З	Н	И	Я	_
2	Д	_	Р	А	Е	Н	И	Я	Н	Е	С	Е				

Для рассечения текста на 8 частей выбраны 2 строки и 4 столбца. Пусть столбцы s_j выбираются в последовательности {4, 1, 3, 2}, а строки r_i – в последовательности {2, 1}. Тогда номер k блока Φ_k, куда записывается очередной символ открытого текста, определяется по формуле

$$k = (r_i - 1)n + s_j,$$

где n – число столбцов.

Первый символ М запишется в блок с номером ($r_i = 2$, $s_j = 4$): $k = (2 - 1) \cdot 4 + 4 = 8$; второй символ Е – в блок с номером ($r_i = 2$, $s_j = 1$): $k = (2 - 1) \cdot 4 + 1 = 5$, и т.д.

Тогда блоки Φ_k, записанные в порядке номеров, будут содержать следующие символы: Φ_1 = (_НЕ...), Φ_2 = (АЯЕ...), Φ_3 = (РИС..,), Φ_4 = = {ДЕН...), Φ_5 = {ЕСРИ...}, Φ_6 = {ОЧЗ...), Φ_7 = {ТЕАЯ...), Φ_8 = {МС-Н...}. Таким образом, один блок открытого текста заменяется восемью блоками, которые в сумме дают длину исходного блока.

Одной из важных проблем при использовании ПЭВМ является проблема хранения

больших массивов данных. Для этой цели применяют различные методы сжатия данных (сжатие рассматривается как метод кодирования).

Методы сжатия данных осуществляют такое преобразование повторяющихся символов и строк символов, которое позволяет использовать для хранения данных меньший объем памяти. Методы сжатия можно разделить на два класса: *статические и динамические* (адаптивные).

Методы статического сжатия данных эффективны, когда частоты появления символов изменяются незначительно. Методы динамического сжатия адаптивно отслеживают неравномерности частот появления символов с сохранением последовательности изменений вероятностей появления символов.

Адаптивные методы сжатия могут динамично реагировать на изменения в открытом тексте, происходящие по мере кодирования. Первые такие методы являлись модификацией кодов Хаффмена и использовали счетчики для хранения текущих частот появления каждого символа. При таких методах наиболее часто встречающиеся символы сдвигаются ближе к корню дерева и, следовательно, получают более короткие кодовые слова.

Кодирование Лемпеля-Зива использует синтаксический метод для динамического источника. Этот метод осуществляет синтаксический анализ символьных потоков, которые не превышают заданной длины, и строит таблицу отображения этих потоков в кодированные слова фиксированной длины. Длина кодового слова зависит от размера таблицы, используемой для хранения кодового отображения поток-слово. Например, размер таблицы в 4096 слов требует 12-битового кодового слова. Кодовое слово является просто табличным адресом соответствующих слов в таблице.

При кодировании по методу Лемпеля-Зива-Уэлча таблица инициализируется символьным множеством и содержит вместо потоков заданной длины пары (кодовое слово, символ) фиксированной длины. Таблица строится на основе синтаксического анализа самого длинного опознанного в таблице потока и использовании последующего символа для формирования нового входа в таблицу. Это позволяет уменьшить размеры таблицы.

В последнее время широкое распространение получили методы сжатия на основе расширяющихся деревьев. Префиксный код переменной длины в этих методах строится на основе положения символов в дереве. Для получения оптимальных кодов дерево балансируется.

Несомненно, криптография должна стать обязательным компонентом защиты всех сколько-нибудь развитых систем. К сожалению, этому мешает огромное количество самых разных барьеров.

Контрольные вопросы

1. Что понимается под шифрованием, расшифрованием и дешифрованием данных?
2. Дайте определения ключа и алгоритма шифрования?
3. Какие методы шифрования Вы знаете? Назовите их достоинства и недостатки.
4. Назовите принципы, лежащие в основе известных Вам методах шифрования.
5. Охарактеризуйте основные алгоритмы симметричного шифрования.
6. Какие ассиметричные алгоритмы шифрования Вам известны?
7. Что такое электронная цифровая подпись и где она применима?

III. Защита компьютерной информации в локальных ЭВМ и информационно-вычислительных сетях

7. МОДЕЛИ БЕЗОПАСНОСТИ ОСНОВНЫХ ОПЕРАЦИОННЫХ СИСТЕМ

7.1. МЕХАНИЗМЫ ЗАЩИТЫ ОПЕРАЦИОННЫХ СИСТЕМ

Операционная система (ОС) есть специально организованная совокупность программ, которая управляет ресурсами системы (ЭВМ, вычислительной системы, других компонентов ИВС) с целью наиболее эффективного их использования и обеспечивает интерфейс пользователя с ресурсами.

Операционные системы, подобно аппаратуре ЭВМ, на пути своего развития прошли несколько поколений.

ОС первого поколения были направлены на ускорение и упрощение перехода с одной задачи пользователя на другую задачу (другого пользователя), что поставило проблему обеспечения безопасности данных, принадлежащих разным задачам.

Второе поколение ОС характеризовалось наращиванием программных средств обеспечения операций ввода-вывода и стандартизацией обработки прерываний. Надежное обеспечение безопасности данных в целом осталось нерешенной проблемой.

К концу 60-х гг. XX в. начал осуществляться переход к мультипроцессорной организации средств ВТ, поэтому проблемы распределения ресурсов и их защиты стали более острыми и трудноразрешимыми. Решение этих проблем привело к соответствующей организации ОС и широкому применению аппаратных средств защиты (защита памяти, аппаратный контроль, диагностика и т.п.).

Основной тенденцией развития вычислительной техники была и остается идея максимальной доступности ее для пользователей, что входит в противоречие с требованием обеспечения безопасности данных.

Под *механизмами защиты* ОС будем понимать все средства и механизмы защиты данных, функционирующие в составе ОС. Операционные системы, в составе которых функционируют средства и механизмы защиты данных, часто называют защищенными системами.

Под *безопасностью* ОС будем понимать такое состояние ОС, при котором невозможно случайное или преднамеренное нарушение функционирования ОС, а также нарушение безопасности находящихся под управлением ОС ресурсов системы. Укажем следующие особенности ОС, которые позволяют выделить вопросы обеспечения безопасности ОС в особую категорию:

- управление всеми ресурсами системы;
- наличие встроенных механизмов, которые прямо или косвенно влияют на безопасность программ и данных, работающих в среде ОС;
- обеспечение интерфейса пользователя с ресурсами системы;
- размеры и сложность ОС.

Большинство ОС обладают дефектами с точки зрения обеспечения безопасности данных в системе, что обусловлено выполнением задачи обеспечения максимальной доступности системы для пользователя.

Рассмотрим *типовые функциональные дефекты* ОС, которые могут привести к созданию каналов утечки данных.

1. *Идентификация*. Каждому ресурсу в системе должно быть присвоено уникальное имя – идентификатор. Во многих системах пользователи не имеют возможности удостовериться в том, что используемые ими ресурсы действительно принадлежат системе.

2. *Пароли*. Большинство пользователей выбирают простейшие пароли, которые легко подобрать или угадать.

3. *Список паролей*. Хранение списка паролей в незашифрованном виде дает возможность его компрометации с последующим НСД к данным.

4. *Пороговые значения*. Для предотвращения попыток несанкционированного входа в систему с помощью подбора пароля необходимо ограничить число таких попыток, что в некоторых ОС не предусмотрено.

5. *Подразумеваемое доверие*. Во многих случаях программы ОС считают, что другие программы работают правильно.

6. *Общая память*. При использовании общей памяти не всегда после выполнения программ очищаются участки оперативной памяти (ОП).

7. *Разрыв связи*. В случае разрыва связи ОС должна немедленно закончить сеанс работы с пользователем или повторно установить подлинность субъекта.

8. *Передача параметров по ссылке, а не по значению* (при передаче параметров по ссылке возможно сохранение параметров в ОП после проверки их корректности, нарушитель может изменить эти данные до их использования).

9. *Система может содержать много элементов* (например, программ), имеющих различные привилегии.

Основной проблемой обеспечения безопасности ОС является проблема создания механизмов контроля доступа к ресурсам системы. Процедура контроля доступа заключается в проверке соответствия запроса субъекта предоставленным ему правам доступа к ресурсам. Кроме того, ОС содержит вспомогательные средства защиты, такие как средства мониторинга, профилактического контроля и аудита. В совокупности механизмы контроля доступа и вспомогательные средства защиты образуют механизмы управления доступом.

Средства профилактического контроля необходимы для отстранения пользователя от непосредственного выполнения критичных с точки зрения безопасности данных операций и передачи этих операций под контроль ОС. Для обеспечения безопасности данных работа с ресурсами системы осуществляется с помощью специальных программ ОС, доступ к которым ограничен.

Средства мониторинга осуществляют постоянное ведение регистрационного журнала, в который заносятся записи о всех событиях в системе. В ОС могут применяться средства сигнализации о НСД, которые используются при обнаружении нарушения безопасности данных или попыток нарушения.

7.2. СИСТЕМА БЕЗОПАСНОСТИ ОС WINDOWS NT

Операционная система Windows NT всегда обладала прекрасными и широко применимыми на практике возможностями защиты. Однократная регистрация в домене Windows NT предоставляет пользователям доступ к ресурсам всей корпоративной сети.

Полноценный набор инструментов Windows NT Server облегчает администраторам управление системой защиты и ее поддержку. Например, администратор может контролировать круг пользователей, имеющих права доступа к сетевым ресурсам: файлам, каталогам, серверам, принтерам и приложениям. Учетные записи пользователей и правами для каждого ресурса можно управлять централизованно.

С помощью простых графических инструментов администратор задает принадлежность к группам, допустимое время работы, срок действия и другие параметры учетной записи. Администратор получает возможность аудита всех событий, связанных с защитой доступа пользователей к файлам, каталогам, принтерам и иным ресурсам. Система также способна блокировать учетную запись пользователя, если число неудачных попыток регистрации превышает заранее определенное. Администраторы вправе устанавливать срок действия паролей, принуждать пользователей к периодической смене паролей и выбору паролей, затрудняющих несанкционированный доступ.

С точки зрения пользователя система защиты Windows NT Server полноценна и несложна в обращении. Простая процедура регистрации обеспечивает доступ к соответствующим ресурсам. Для пользователя невидимы такие процессы, как шифрование пароля на системном уровне. Пользователь сам определяет права доступа к тем ресурсам, которыми владеет. Например, чтобы разрешить совместное использование своего документа, он указывает, кто и как может с ним работать. Разумеется, доступ к ресурсам предприятия контролируется только администраторами с соответствующими полномочиями.

Более глубокий уровень безопасности – то, как Windows NT Server защищает данные, находящиеся в физической памяти компьютера. Доступ к ним предоставляется только имеющим на это право программам. Если данные больше не содержатся на диске, система предотвращает несанкционированный доступ к той области диска, где они содержались. При такой системе защиты никакая программа не «подсмотрит» в виртуальной памяти машины информацию, с которой оперирует в данный момент другое приложение.

Удаленный доступ через открытые сети и связь предприятий через Интернет стимулируют постоянное и быстрое развитие технологий безопасности. В качестве примера можно выделить сертификаты открытых ключей и динамические пароли. Архитектура безопасности Windows NT однозначно оценивается как превосходящая и эти, и многие будущие технологии. Перечислим функции безопасности Windows NT:

- информация о доменных правилах безопасности и учетная информация хранятся в каталоге *Active Directory* (служба каталогов *Active Directory* обеспечивает тиражирование и доступность учетной информации на многих контроллерах домена, а также позволяет удаленное администрирование);
- в *Active Directory* поддерживается иерархичное пространство имен пользователей, групп и учетных записей машин (учетные записи могут быть сгруппированы по организационным единицам);
- административные права на создание и управление группами учетных записей пользователей могут быть делегированы на уровень организационных единиц (возможно установление дифференцированных прав доступа к отдельным свойствам пользовательских объектов);
- тиражирование *Active Directory* позволяет изменять учетную информацию на любом контроллере домена, а не только на первичном (копии *Active Directory*, хранящиеся на других контроллерах домена, обновляются и синхронизируются автоматически);
- доменная модель изменена и использует *Active Directory* для поддержки многоуровневого дерева доменов (управление доверительными отношениями между доменами упрощено в пределах всего дерева доменов);
- в систему безопасности включены новые механизмы аутентификации, такие как *Kerberos v5 и TLS* (*Transport Layer Security*), базирующиеся на стандартах безопасности Интернета;
- протоколы защищенных каналов (SSL 3.0/TLS) обеспечивают поддержку надежной аутентификации клиента (осуществляется сопо-ставление мандатов пользователей в форме сертификатов открытых ключей с существующими учетными записями Windows NT);

– дополнительно к регистрации посредством ввода пароля может поддерживаться аутентификация с использованием смарт-карт.

В состав Windows NT входит Microsoft Certificate Server, позволяющий выдавать сотрудникам и партнерам сертификаты X.509 версии 3. Системные администраторы могут указывать, сертификаты каких уполномоченных являются доверяемыми в системе и, таким образом, контролировать аутентификацию доступа к ресурсам.

Внешние пользователи, не имеющие учетных записей Windows NT, могут быть аутентифицированы с помощью сертификатов открытых ключей и соотнесены с существующей учетной записью. Права доступа, назначенные для этой учетной записи, определяют права внешних пользователей на доступ к ресурсам.

В распоряжении пользователей простые средства управления парами закрытых (открытых) ключей и сертификатами, используемыми для доступа к ресурсам системы.

Технология шифрования встроена в операционную систему и позволяет использовать цифровые подписи для идентификации потоков.

Протокол аутентификации Kerberos определяет взаимодействие между клиентом и сетевым сервисом аутентификации, известным как *KDC* (*Key Distribution Center*). В Windows NT KDC используется как сервис аутентификации на всех контроллерах домена. Домен Windows NT эквивалентен области Kerberos, но к ней обращаются как к домену. Реализация протокола Kerberos в Windows NT основана на определении Kerberos в RFC1510, Клиент Kerberos реализован в виде ПФБ (поставщика функций безопасности) Windows NT, основанном на SSPI. Начальная аутентификация Kerberos интегрирована с процедурой WinLogon. Сервер *Kerberos* (*KDC*) интегрирован с существующими службами безопасности Windows NT, исполняемыми на контроллере домена. Для хранения информации о пользователях и группах он использует службу каталогов *Active Directory*.

Протокол *Kerberos* усиливает существующие функции безопасности Windows NT и добавляет новые:

– повышенная скорость аутентификации при установлении начального соединения (сервер приложений не обращается к контроллеру домена для аутентификации клиента);

– делегирование аутентификации в многоярусных архитектурах клиент-сервер (при подключении клиента к серверу, последний имперсонирует (олицетворяет) клиента в этой системе, но если серверу для завершения транзакции нужно выполнить сетевое подключение к другому серверу, протокол *Kerberos* позволяет делегировать аутентификацию первого сервера и подключиться ко второму от имени клиента);

– транзитивные доверительные отношения для междудоменной аутентификации (т.е. пользователь может быть аутентифицирован в любом месте дерева доменов) упрощают управление доменами в больших сетях с несколькими доменами.

Основы Kerberos. Протокол *Kerberos* является протоколом аутентификации с совместным секретом – и пользователю, и *KDC* известен пароль (*KDC* – зашифрованный пароль). *Kerberos* определяет серию обменов между клиентами, *KDC* и серверами для получения билетов *Kerberos*. Когда клиент начинает регистрацию в Windows NT, поставщик функций безопасности Kerberos получает начальный билет *Kerberos TGT* (Ticket grantticket), основанный на зашифрованном представлении пароля. Windows NT хранит *TGT* в кэше билетов на рабочей станции, связанной с контекстом регистрации пользователя. При попытке клиентской программы обратиться к сетевой службе проверяется кэш билетов: есть ли в нем верный билет для текущего сеанса работы с сервером. Если такого билета нет, на *KDC* посылается запрос с *TGT* для получения сеансового билета, разрешающего доступ к серверу.

Сеансовый билет добавляется в кэш и может впоследствии быть использован повторно для доступа к тому же самому серверу в течение времени действия билета. Время действия билета устанавливается доменными правилами и обычно равно восьми часам. Если время действия билета истекает в процессе сеанса, то поставщик функций безопасности *Kerberos* возвращает соответствующую ошибку, что позволяет клиенту и серверу обновить билет, создать новый сеансовый ключ и возобновить подключение.

Сеансовый билет *Kerberos* предъявляется удаленной службе в сообщении о начале подключения. Части сеансового билета зашифрованы секретным ключом, используемым совместно службой и *KDC.* Сервер может быстро аутентифицировать клиента, проверив его сеансовый билет и не обращаясь к сервису аутентификации, так как на сервере в кэше хранится копия секретного ключа. Соединение при этом происходит гораздо быстрее, чем при аутентификации *NTLM*, где сервер получает мандаты пользователя, а затем проверяет их, подключившись к контроллеру домена.

Сеансовые билеты *Kerberos* содержат уникальный сеансовый ключ, созданный *KDC* для симметричного шифрования информации об аутентификации, а также данных, передаваемых от клиента к серверу. В модели *Kerberos KDC* используется в качестве интерактивной доверенной стороны, генерирующей сеансовый ключ.

Интеграция Kerberos. Протокол *Kerberos* полностью интегрирован с системой безопасности и контроля доступа Windows NT. Начальная регистрация в Windows NT обеспечивается процедурой WinLogon, использующей ПФБ *Kerberos* для получения начального билета *TGT*. Другие компоненты системы, например, *Redirector*, применяют интерфейс *SSPI* к ПФБ *Kerberos* для получения сеансового билета для удаленного доступа к файлам сервера SMB.

Взаимодействие Kerberos. Протокол *Kerberos* версии 5 реализован в различных системах и используется для единообразия аутентификации в распределенной сети.

Под взаимодействием *Kerberos* подразумевается общий протокол, позволяющий учетным записям аутентифицированных пользователей, хранящимся в одной базе осуществлять доступ ко всем сервисам в гетерогенной среде. Взаимодействие *Kerberos* основывается на следующих характеристиках:

– общий протокол аутентификации пользователя или сервиса по основному имени при сетевом подключении;

– возможность определения доверительных отношений между областями *Kerberos* и создания ссылочных запросов билетов между областями;

– поддержка определенных в RFC 1510 требований к взаимодействию, относящихся к алгоритмам шифрования и контрольных сумм, взаимной аутентификации и другим возможностям билетов;

– поддержка форматов маркера безопасности *Kerberos* версии 5 для установления контекста и обмена сообщениями.

Поддержка Kerberos открытых ключей. В Windows NT также реализованы расширения протокола Kerberos, поддерживающие дополнительно к аутентификации с совместно используемым секретным ключом аутентификацию, основанную на парах открытого (закрытого) ключа. Поддержка открытых ключей позволяет клиентам запрашивать начальный ключ *TGT* с помощью закрытого ключа, в то время как *KDC* проверяет запрос с помощью открытого ключа, полученного из сертификата X.509 (хранится в пользовательском объекте в каталоге *Active Directory*), Сертификат пользователя может быть выдан как сторонним уполномоченным сертификации (*Certification Authority*), так и *Microsoft Certificate Server*, входящим в Windows NT. После начальной аутентификации закрытым ключом используются стандартные протоколы Kerberos для получения сеансовых билетов на доступ к сетевым службам,

Модель безопасности Windows NT обеспечивает однородный и унифицированный механизм контроля за доступом к ресурсам домена на основе членства в группах. Компоненты безопасности Windows NT доверяют хранимой в каталоге информации о защите. Например, сервис аутентификации Windows NT хранит зашифрованные пароли пользователей в безопасной части каталога объектов пользователя. По умолчанию операционная система «считает», что правила безопасности защищены и не могут быть изменены кем-либо несанкционированно. Общая политика безопасности домена также хранится в каталоге Active Directory.

Делегирование административных полномочий – гибкий инструмент ограничения административной деятельности рамками части домена. Этот метод позволяет предоставить отдельным сотрудникам возможность управления пользователями или группами в заданных пределах и, в то же время, не дает им прав на управление учетными записями, относящимися к другим подразделениям.

Права на определение новых пользователей или создание групп пользователей делегируются на уровне OU или контейнера, в котором создана учетная запись.

Существует три способа делегирования административных полномочий:

1) на изменение свойств определенного контейнера, например, LocalDomainPolicies самого домена;

2) на создание и удаление дочерних объектов определенного типа (пользователи, группы, принтеры и пр.) внутри OU;

3) на обновление определенных свойств некоторых дочерних объектов внутри OU (например, право устанавливать пароль для объектов типа User).

Делегировать полномочия просто. Достаточно выбрать лицо, которому будут делегированы полномочия, и указать, какие именно полномочия передаются. Интерфейс программы администрирования Active Directory позволяет без затруднений просматривать информацию о делегировании, определенную для контейнеров.

Наследование прав доступа означает, что информация об управлении доступом, определенная в высших слоях контейнеров в каталоге, распространяется ниже – на вложенные контейнеры и объекты-листья. Существуют две модели наследования прав доступа: динамическая и статическая. При динамическом наследовании права определяются путем оценки разрешений на доступ, назначенных непосредственно для объекта, а также для всех родительских объектов в каталоге. Это позволяет эффективно управлять досту-

пом к части дерева каталога, внося изменения в контейнер, влияющий на все вложенные контейнеры и объекты-листья. Обратная сторона такой гибкости – недостаточно высокая производительность из-за времени определения эффективных прав доступа при запросе пользователя.

В Windows NT реализована статическая форма наследования прав доступа, иногда также называемая наследованием в момент создания. Информация об управлении доступом к контейнеру распространяется на все вложенные объекты контейнера. При создании нового объекта наследуемые права сливаются с правами доступа, назначаемыми по умолчанию. Любые изменения наследуемых прав доступа, выполняемые в дальнейшем на высших уровнях дерева, должны распространяться на все дочерние объекты. Новые наследуемые права доступа распространяются на объекты Active Directory в соответствии с тем, как эти новые права определены. Статическая модель наследования позволяет увеличить производительность.

Элементы безопасности системы. Далее будут рассмотрены вопросы реализации политики безопасности: управлению учетными записями пользователей и групп, исполнению и делегированию административных функций.

Учетные записи пользователей и групп. Любой пользователь Windows NT характеризуется определенной учетной записью. Под учетной записью понимается совокупность прав и дополнительных параметров, ассоциированных с определенным пользователем. Кроме того, пользователь принадлежит к одной или нескольким группам. Принадлежность к группе позволяет быстро и эффективно назначать права доступа и полномочия.

К встроенным учетным записям пользователей относятся:

- Guest – учетная запись, фиксирующая минимальные привилегии гостя;
- Administrator – встроенная учетная запись для пользователей, наделенных максимальными привилегиями;
- Krbtgt – встроенная учетная запись, используемая при начальной аутентификации Kerberos.

Кроме них имеются скрытые встроенные учетные записи:

- System – учетная запись, используемая операционной системой;
- Creator owner – создатель (файла или каталога).

Перечислим встроенные группы:

- локальные (Account operators; Administrators; Backup operators; Guests; Print operators; Replicator; Server operators; Users);
- глобальные (Domain guests – гости домена; Domain Users – пользователи домена; Domain Admins – администраторы домена).

Помимо этих встроенных групп имеется еще ряд специальных групп:

- Everyone – в эту группу по умолчанию включаются вообще все пользователи в системе;
- Authenticated users – в эту группу включаются только аутентифицированные пользователи домена;
- Self – сам объект.

Для просмотра и модификации свойств учетной записи достаточно щелкнуть имя пользователя или группы и на экране появится диалоговое окно User Properties.

- General – общее описание пользователя;
- Address – домашний и рабочий адрес пользователя;
- Account – обязательные параметры учетной записи;
- Telephone/notes – необязательные параметры;
- Organization – дополнительные необязательные сведения;
- Membership – обязательная информация о принадлежности пользователя к группам;
- Dial-in – параметры удаленного доступа;
- Object – идентификационные сведения о пользовательском объекте;
- Security – информация о защите объекта.

Локальная политика безопасности – регламентирует правила безопасности на локальном компьютере. С ее помощью можно распределить административные роли, конкретизировать привилегии пользователей, назначить правила аудита.

По умолчанию поддерживаются следующие области безопасности:

- политика безопасности – задание различных атрибутов безопасности на локальном и доменном уровнях; так же охватывает некоторые установки на машинном уровне;
- управление группами с ограничениями – позволяет управлять членством в группах, которые, по мнению администратора, «чувствительны» с точки зрения безопасности системы;

• управление правами и привилегиями – позволяет редактировать список пользователей и их специфических прав и привилегий;

• деревья объектов – включают три области защиты: объекты каталога Active Directory, ключи реестра, локальную файловую систему; для каждого объекта в дереве шаблоны безопасности позволяют конфигурировать и анализировать характеристики дескрипторов защиты, включая владельцев объекта, списки контроля доступа и параметры аудита;

• системные службы (сетевые или локальные) – построенные соответствующим образом дают возможность независимым производителям программного обеспечения расширять редактор конфигураций безопасности для устранения специфических проблем.

Конфигурирование безопасности. Для конфигурирования параметров безопасности системы используются шаблоны.

Управление доступом к реестру. Реестр – это дерево объектов. Доступ к каждому объекту в дереве должен быть регламентирован. Выбрав в окне обзорного просмотра ветвь, соответствующую шаблону Custom, щелкните папку Registry. В правой части окна появится список ветвей реестра, доступ к которым можно ограничивать. В шаблоне, поставляемом с редактором, приведена ветвь MACHINE\HARDWARE, которую надо истолковывать как HKEY_LOCAL_MACHINE\Hardware. Чтобы добавить к дереву новые ветви, их надо в явном виде прописать в шаблоне с помощью любого текстового редактора. Для разграничения доступа к выбранной ветви реестра дважды щелкните ее имя и укажите нужный тип доступа и имя соответствующей учетной записи. Изменения будут занесены в шаблон.

7.3. ЗАЩИТА В ОПЕРАЦИОННОЙ СИСТЕМЕ UNIX

Операционная система UNIX относится к категории многопользовательских многопрограммных ОС, работающих в режиме разделения времени. Богатые возможности, заложенные в ОС UNIX, сделали ее наиболее популярной в мире. ОС UNIX поддерживается практически на всех типах ЭВМ.

Организация работ в ОС UNIX основана на понятии последовательного процесса как единицы работы, управления и потребления ресурсов. Взаимодействие процессов внутри ядра (процесс вызывает ядро как подпрограмму) происходит по принципу сопрограмм. Последовательность вычислений внутри процесса строго выдерживается: процесс, в частности, не может активизировать ввод–вывод и продолжать вычисление параллельно с ним. В этом случае требуется создать параллельный процесс.

Резидентная в ОП часть ОС называется *ядром*. Ядро ОС UNIX состоит из двух основных частей: управления процессами и управления устройствами. Управление процессами резервирует ресурсы, определяет последовательность выполнения процессов и принимает запросы на обслуживание. Управление устройствами контролирует передачу данных между ОП и периферийными устройствами.

В любой момент времени выполняется либо программа пользователя (процесс), либо команда ОС. В каждый момент времени лишь один пользовательский процесс активен, а все остальные приостановлены. Ядро ОС UNIX служит для удовлетворения потребностей процессов.

Процесс – это программа на этапе выполнения. В некоторый момент времени программе могут соответствовать один или несколько процессов, или не соответствовать ни одного. Считается, что процесс является объектом, учтенным в специальной таблице ядра системы. Наиболее важная информация о процессе хранится в двух местах: в таблице процессов и в таблице пользователя, называемой также контекстом процесса. Таблица процессов всегда находится в памяти и содержит на каждый процесс по одному элементу, в котором отражается состояние процесса: адрес в памяти или адрес своппинга, размер, идентификаторы процесса и запустившего его пользователя. Таблица пользователя существует для каждого активного процесса и к ней могут непосредственно адресоваться только программы ядра (ядро резервирует по одному контексту на каждый активный процесс). В этой таблице содержится информация, требуемая во время выполнения процесса: идентификационные номера пользователя и группы, предназначенные для определения привилегий доступа к файлам, ссылки на системную таблицу файлов для всех открытых процессом файлов, указатель на индексный дескриптор текущего каталога в таблице индексных дескрипторов и список реакций на различные ситуации. Если процесс приостанавливается, контекст становится недоступным и немодифицируемым.

Каталоги файловой системы ОС UNIX «спрятаны» от пользователей и защищены механизмами ОС. Скрытой частью файловой организации в ОС UNIX является индекс-

ный дескриптор файла, который описывает расположение файла, его длину, метод доступа к файлу, даты, связанные с историей создания файла, идентификатор владельца и т.д.

Работа с таблицами является привилегией ядра, что обеспечивает сохранность и безопасность системы. Структура данных ядра ОС, обеспечивающих доступ к файлам, приведена на рис. 7.1.

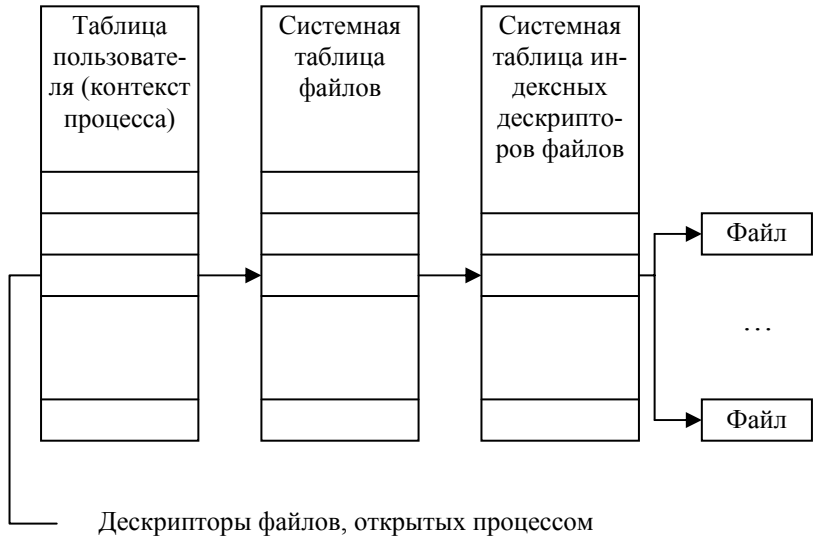


Рис. 7.1. Структура данных ядра ОС UNIX

При взаимодействии с ОС UNIX пользователь может обращаться к большому числу информационных объектов или файлов, объединенных в каталоги. Файловая система ОС UNIX имеет иерархическую структуру.

В ОС UNIX используется четыре типа файлов: обычные, специальные, каталоги, а в некоторых версиях ОС и FIFO-файлы (First In – First Out). Обычные файлы содержат данные пользователей. Специальные файлы предназначены для организации взаимодействия с устройствами ввода–вывода. Доступ к любому устройству реализуется как обслуживание запроса к специальному (дисковому) файлу. Каталоги используются системой для поддержания файловой структуры. Особенность каталогов состоит в том, что пользователь может читать их содержимое, но выполнять записи в каталоги (изменять структуру каталогов) может только ОС. В ОС UNIX, организуются именованные программные каналы, являющиеся соединительным средством между стандартным выводом одной программы и стандартным вводом другой.

Схема типичной файловой системы ОС UNIX приведена на рис. 7.2. Рассмотрим основные механизмы защиты данных, реализованные в ОС UNIX.

Управление доступом к системе. При включении пользователя в число абонентов

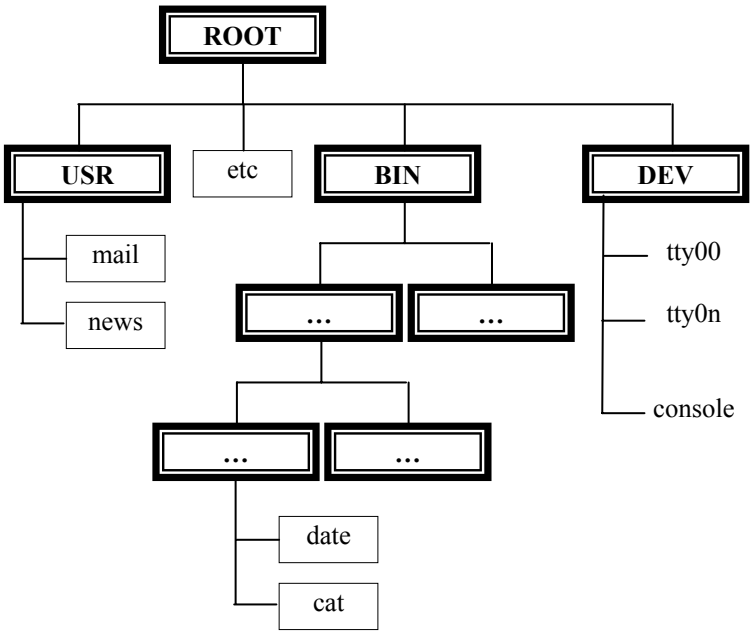


Рис. 7.2. Схема файловой системы ОС UNIX

ему выдается регистрационное имя (идентификатор) для входа в систему и пароль, который служит для подтверждения идентификатора пользователя. В отдельных версиях ОС UNIX, помимо идентификатора и пароля, требуется ввод номера телефона, с которого выполняется подключение к системе. Администратор системы и пользователь могут изменить пароль командой passwd. При вводе этой команды ОС запрашивает ввод текущего пароля, а затем требует ввести новый пароль. Если предложенный пароль не удовлетворяет требования системы, то запрос на ввод пароля может быть повторен. Если предложенный пароль удовлетворителен, ОС просит ввести его снова с тем, чтобы убедиться в корректности ввода пароля.

Пользователи, которым разрешен вход в систему, перечислены в учетном файле пользователей /etc/passwd. Этот текстовый файл содержит следующие данные: имя пользователя, зашифрованный пароль, идентификатор пользователя, идентификатор группы, начальный текущий каталог и имя исполняемого файла, используемого в качестве интерпретатора команд. Пароль шифруется, как правило, с использованием DES-алгоритма.

Управление доступом к данным. Операционная система UNIX поддерживает для любого файла комплекс характеристик, определяющих санкционированность доступа, тип файла, его размер и точное местоположение на диске. При каждом обращении к файлу система проверяет право пользоваться им. Операционная система UNIX допускает выполнение трех типов операций над файлами: чтение, запись и выполнение. Чтение файла означает, что доступно его содержимое, а запись – что возможны изменения содержимого файла. Выполнение приводит либо к загрузке файла в ОП либо к выполнению содержащихся в файле команд системного монитора Shell. Разрешение на выполнение каталога означает, что в нем допустим поиск с целью формирования полного имени на пути к файлу. Любой из файлов в ОС UNIX имеет определенного владельца и привязан к некоторой группе. Файл наследует их от процесса, создавшего файл. Пользователь и группа, идентификаторы которых связаны с файлом, считаются его владельцами.

Идентификаторы пользователя и группы, связанные с процессом, определяют его права при доступе к файлам. По отношению к конкретному файлу все процессы делятся на три категории:

1) владелец файла (процессы, имевшие идентификатор пользователя, совпадающий с идентификатором владельца файла);

2) члены группы владельца файла (процессы, имеющие идентификатор группы, совпадающий с идентификатором группы, которой принадлежит файл);

3) прочие (процессы, не попавшие в первые две категории).

Владелец файла обладает одними привилегиями на доступ к нему, члены группы, в которую входит файл – другими, все остальные пользователи – третьими. Каждый файл содержит код защиты, который присваивается файлу при его создании. Код защиты располагается в индексном дескрипторе файла и содержит десять символов, причем первый символ определяет тип файла, а последующие девять – право на доступ к нему. Три вида операций (чтение, запись и выполнение) и три категории (уровни привилегий на доступ: владельцев, групп и прочих пользователей) дают в совокупности девять возможных вариантов разрешений или запретов на доступ к файлу. Первые три символа определяют возможности чтения (*r*), записи (*w*) и выполнения (*e*) на уровне владельца, следующие три – на уровне группы, в которую входит владелец, и последние три – на уровне остальных пользователей. Наличие символов *r*, *w* и *e* указывает на соответствующее разрешение.

Если процесс требует доступа к файлу, то сначала определяется категория, в которую по отношению к этому файлу он попадает. Затем из кода защиты выбираются те три символа, которые соответствуют данной категории, и выполняется проверка: разрешен ли процессу требуемый доступ. Если доступ не разрешен, системный вызов, посредством которого процесс сделал запрос на доступ, отвергается ядром ОС.

По соглашению, принятому в ОС UNIX, привилегированный пользователь имеет идентификатор, равный нулю. Процесс, с которым связан нулевой идентификатор пользователя, считается привилеги-рованным. Независимо от кода защиты файла привилегированный процесс имеет право доступа к файлу для чтения и записи. Если в коде защиты хотя бы одной категории пользователей (процессов) есть разрешение на выполнение файла, привилегированный процесс тоже имеет право выполнять этот файл.

С помощью специальных команд владелец файла (и привилегированный пользователь) может изменять распределение привилегий. Команда Change mode позволяет изменить код защиты, команда Change owner меняет право на владение файлом, а команда Change group – принадлежность к той или иной группе. Пользователь может изменять режимы доступа только для тех файлов, которыми он владеет.

Защита хранимых данных. Для защиты хранимых данных в составе ОС UNIX имеется утилита crypt, которая читает данные со стандартного ввода, шифрует их и направляет на стандартный вывод. Шифрование применяется при необходимости предоставления абсолютного права владения файлом.

Восстановление файловой системы. Операционная система UNIX поддерживает три основных набора утилит копирования: программы volcopy/labelit, dump/restor и cpio. Программа volcopy целиком переписывает файловую систему, проверяя с помощью программы labelit соответствие меток требуемых томов. Программа dump обеспечивает копирование лишь тех файлов, которые были записаны позднее определенной даты (защита накоплением). Программа restor может анализировать данные, созданные программой dump, и восстанавливать отдельные файлы или всю файловую систему полностью. Программа cpio предназначена для создания одного большого файла, содержащего образ всей файловой системы или какой-либо ее части.

Для восстановления поврежденной, например, в результате сбоев в работе аппаратуры файловой системы используются программы fsck и fsdb.

За сохранность файловой системы, адаптацию программного обеспечения к конкретным условиям эксплуатации, периодическое копирование пользовательских файлов, восстановление потерянных данных и другие операции ответственность возложена на администратора системы.

Усложненное управление доступом. В составе утилит ОС UNIX находится утилита cron, которая предоставляет возможность запускать пользовательские программы в определенные моменты (промежутки) времени и, соответственно, ввести временные параметры для ограничения доступа пользователей.

Для управления доступом в ОС UNIX также применяется разрешение установки идентификатора владельца. Такое разрешение дает возможность получить привилегии владельца файла на время выполнения соответствующей программы. Владелец файлов может установить такой режим, в котором другие пользователи имеют возможность назначать собственные идентификаторы режима.

Доступ, *основанный на полномочиях*, использует соответствие меток. Для этого вводятся метки объектов (файлов) и субъектов (процессов), а также понятия доминанты и равенства меток (для выражения отношения между метками). Создаваемый файл наследует метку от создавшего его процесса. Вводятся соотношения, определяющие права процессов по отношению к файлам.

Интерфейс дискретного доступа существенно детализирует имеющиеся механизмы защиты ОС UNIX. Вводимые средства можно разделить на следующие группы:

- работа со списками доступа при дискретной защите;
- проверка права доступа;
- управление доступом на основе полномочий;
- работа привилегированных пользователей.

В рамках проекта Posix создан интерфейс системного администратора. Указанный интерфейс определяет объекты и множества действий, которые можно выполнить над объектами. В качестве классов субъектов и объектов предложены пользователь, группа пользователей, устройство, файловая система, процесс, очередь, вход в очередь, машина, система, администратор, программное обеспечение и др. Определены атрибуты таких классов, операции надклассами и события, которые могут с ними происходить.

7.4. ЗАЩИТА В ОПЕРАЦИОННОЙ СИСТЕМЕ NOVELL NETWARE

Авторизация доступа к данным сети. В NetWare реализованы три уровня защиты данных (рис. 7.3).

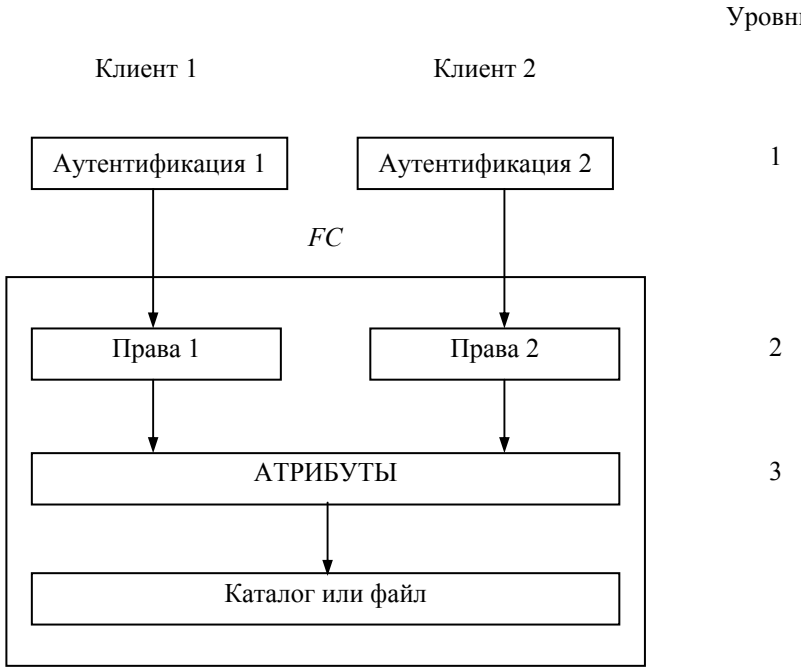


Рис. 7.3. Уровни защиты данных в NetWare

Здесь под аутентификацией понимается:

– процесс подтверждения подлинности клиента при его подключении к сети;

– процесс установления подлинности пакетов, передаваемых между сервером и рабочей станцией.

Права по отношению к файлу (каталогу) определяют, какие операции пользователь может выполнить с файлом (каталогом). Администратор может для каждого клиента сети определить права по отношению к любому сетевому файлу или каталогу.

Атрибуты определяют некоторые системные свойства файлов (каталогов). Они могут быть назначены администратором для любого сетевого файла или каталога. Например, чтобы записать данные в файл, клиент должен:

– знать свой идентификатор и пароль для подключения к сети;

– иметь право записи данных в этот файл;

– файл должен иметь атрибут, разрешающий запись данных.

Следует отметить, что атрибуты файла (каталога) имеют более высокий приоритет, чем права пользователей по отношению к этому файлу.

Аутентификация пользователей при подключении к сети. Подключение к сети выполняется с помощью утилиты LOGIN.EXE. Эта программа передает на сервер идентификатор, введенный пользователем.

По этому идентификатору NetWare выполняет поиск соответствующего объекта пользователя в системной базе данных сетевых ресурсов. Если в базе данных хранится значение пароля для этого клиента, то NetWare посылает на рабочую станцию зашифрованный с помощью пароля открытый ключ (симметричное шифрование). На рабочей станции этот ключ расшифровывается с помощью пароля, введенного пользователем, и используется для получения подписи запроса (пакета) к серверу о продолжении работы. Сервер расшифровывает эту подпись с помощью закрытого ключа (асимметричное шифрование), проверяет ее и посылает подтверждение на рабочую станцию. В дальнейшем каждый NCP-пакет снабжается подписью, получаемой в результате кодирования открытом ключом контрольной суммы содержимого пакета и случайного числа Nonce. Это число генерируется для каждого сеанса. Поэтому подписи пакетов не повторяются для разных сеансов, даже если пользователь выполняет те же самые действия

NCP-пакеты могут подписываться и рабочими станциями, и файловым сервером. Для инициирования включения подписи в NCP-пакеты администратор может задать один из следующих уровней:

0 – сервер не подписывает пакет;

1 – сервер подписывает пакет, если этого требует клиент (уровень на станции больше или равен 2);

2 – сервер подписывает пакет, если клиент также способен это сделать (уровень на станции больше или равен 1);

3 – сервер подписывает пакет и требует этого от всех клиентов (иначе подключение к сети невозможно).

Права пользователей по отношению к каталогам и файлам. Права, которые могут быть предоставлены пользователю (или группе пользователей) по отношению к каталогу или файлу, перечислены в табл. 7.1.

7.1. Список возможных прав по отношению к каталогу или файлу

Право	Обозначение	Описание
Supervisor	*S*	Предоставляет все права по отношению к каталогу или файлу, включая возможность назначения этого права другим пользователям. Не блокируется фильтром наследуемых прав IRF. Это право не может быть удалено ниже по дереву каталогов
Read	*R*	Чтение существующего файла (просмотр содержимого текстового файла, просмотр записей в файле базы данных и т.д.)
Write	*W*	Запись в существующий файл (добавление, удаление частей текста, редактирование записей базы данных)
Create	*C*	Создание в каталоге новых файлов (и запись в них) и подкаталогов. На уровне файла позволяет восстанавливать файл, если он был ошибочно удален
Erase	*E*	Удаление существующих файлов и каталогов
Modify	*M*	Изменение имен и атрибутов (файлов и каталогов), но не содержимого файлов
File Scan	*F*	Просмотр в каталоге имен файлов и подкаталогов. По отношению к файлу – возможность видеть структуру каталогов от корневого уровня до этого файла (путь доступа)
AccessControl	*A*	Возможность предоставлять другим пользователям все права, кроме Supervisor. Возможность изменять фильтр наследуемых прав IRF

Права и фильтры (маски) наследуемых прав назначаются администратором сети с помощью утилит NetWare. Но назначение прав для каждого пользователя по отношению ко всем требуемым файлам и каталогам – это утомительная задача. В NetWare предлагается механизм наследования прав. Прежде всего, введем некоторые определения.

Опекун (Trustees) – это пользователь (группа пользователей, другой объект), которому администратор с помощью утилиты (например, FILER) явно назначает права по отношению к какому-либо файлу или каталогу. Такие права называются опекунскими назначениями.

Фильтр наследуемых прав (IRF – Inherited Right Filter) – это свойство файла (каталога), определяющее, какие права данный файл (каталог) может унаследовать от родительского каталога. Фильтр назначается администратором с помощью утилиты (например, FILER).

Наследуемые права – права, передаваемые (распространяемые) от родительского каталога.

Эффективные права – права, которыми пользователь реально обладает по отношению к файлу или каталогу.

Права доступа к объектам NDS и их свойствам. Системная база данных сетевых ресурсов (СБДСР) представляет собой совокупность объектов, их свойств и значений этих свойств. В NetWare 4.x эта база данных называется NDS (NetWare Directory Services), а в NetWare 3.x – Bindery. Объекты NDS связаны между собой в иерархическую структуру, которую часто называют деревом NDS. На верхних уровнях дерева (ближе к корню [Root]) описываются логические ресурсы, которые принято называть *контейнерными* объектами. На самом нижнем (листьевом) уровне располагаются описания физических ресурсов, которые называют *оконечными* объектами.

В качестве контейнерных объектов используются объекты типа [Root] (корень), C (страна), O (организация), OU (организационная единица). Оконечные объекты – это User (пользователь), Group (группа), NetWare Server (сервер NetWare), Volume (том файлового сервера), Directories (директория тома) и т.д. Оконечные объекты имеют единое обозначение – CN.

В NetWare 4.x разработан механизм защиты дерева NDS. Этот механизм очень похож на механизм защиты файловой системы, который был рассмотрен ранее. Чтобы облегчить понимание этого механизма, оконечный объект можно интерпретировать как файл, а контейнерный объект – как каталог, в котором могут быть созданы другие контейнерные объекты (как бы подкаталоги) и оконечные объекты (как бы файлы). На рис. 7.4 представлена схема дерева NDS, где символами [Root], C, O, OU обозначены контейнерные объекты, а символами CN – оконечные объекты.

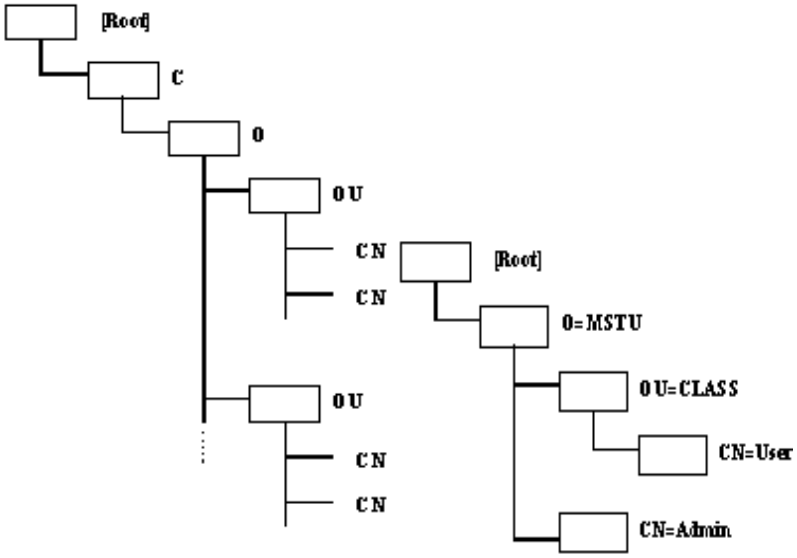


Рис. 7.4. Схема дерева NDS

В отличие от файловой системы здесь права по отношению к какому-либо объекту можно предоставить любому контейнерному или оконечному объекту дерева NDS. В частности, допустимо рекурсивное назначение прав объекта по отношению к этому же объекту.

Права, которые могут быть предоставлены объекту по отношению к другому или тому же самому объекту, перечислены в табл. 7.2.

7.2. Список возможных прав по отношению к объекту

Право	Обозначение	Описание
Supervisor	*S*	Гарантирует все привилегии по отношению к объекту и его свойствам. В отличие от файловой системы это право может быть блокировано фильтром наследуемых прав IRF, который может быть назначен для каждого объекта
Browse	*B*	Обеспечивает просмотр объекта в дереве NDS
Create	*C*	Это право может быть назначено только по отношению к контейнерному объекту (контейнеру). Позволяет создавать объекты в данном и во всех дочерних контейнерах
Delete	*D*	Позволяет удалять объект из дерева NDS
Rename	*R*	Позволяет изменять имя объекта

Администратор сети может для каждого объекта в дереве NDS определить значения свойств этого объекта. Для объекта User – это имя Login, требования к паролю, пароль пользователя, пользовательский сценарий подключения и т.д.

Контрольные вопросы

1. Сформулируйте список функциональных дефектов с точки зре-ния защиты в используемой ОС.
2. Какие элементы безопасности содержит ОС Windows NT?
3. Назовите элементы безопасности ОС UNIX?
4. Охарактеризуйте элементы безопасности ОС Novell NetWare?

8. СИСТЕМЫ ЗАЩИТЫ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

8.1. КЛАССИФИКАЦИЯ СИСТЕМ ЗАЩИТЫ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Системы защиты программного обеспечения (СЗПО) широко распространены и находятся в постоянном развитии, благодаря расширению рынка программного обеспечения (ПО) и телекоммуникационных технологий. Необходимость использования систем защиты СЗПО обусловлена рядом проблем, среди которых следует выделить: незаконное использование алгоритмов, являющихся интеллектуальной собственностью автора, при написании аналогов продукта (промышленный шпионаж); несанкционированное использование ПО (кража и копирование); несанкционированная модификация ПО с целью внедрения программных злоупотреблений; незаконное распространение и сбыт ПО (пиратство).

Существующие системы защиты программного обеспечения можно классифицировать по ряду признаков, среди которых можно выделить:

- метод установки;
- используемые механизмы защиты;
- принцип функционирования.

Системы защиты ПО по *методу установки* можно подразделить на:

1) системы, устанавливаемые на скомпилированные модули ПО;
2) системы, встраиваемые в исходный код ПО до компиляции;
3) комбинированные.

Системы первого типа наиболее удобны для производителя ПО, так как легко можно защитить уже полностью готовое и оттестированное ПО, а потому и наиболее популярны. В то же время стойкость этих систем достаточно низка (в зависимости от принципа действия СЗ), так как для обхода защиты достаточно определить точку завершения работы «конверта» защиты и передачи управления защищенной программе, а затем принудительно ее сохранить в незащищенном виде.

Системы второго типа неудобны для производителя ПО, так как возникает необходимость обучать персонал работе с программным интерфейсом (API) системы защиты с вытекающими отсюда денежными и временными затратами. Кроме того, усложняется процесс тестирования ПО и снижается его надежность, так как кроме самого ПО ошибки может содержать API системы защиты или процедуры, его использующие. Но такие системы являются более стойкими к атакам, потому что здесь исчезает четкая граница между системой защиты и как таковым ПО.

Наиболее живучими являются комбинированные системы защиты. Сохраняя достоинства и недостатки систем второго типа, они максимально затрудняют анализ и дезактивацию своих алгоритмов.

По *используемым механизмам защиты* СЗ можно классифицировать на:

1) системы, использующие сложные логические механизмы;
2) системы, использующие шифрование защищаемого ПО;
3) комбинированные системы.

Системы первого типа используют различные методы и приемы, ориентированные на затруднение дизассемблирования, отладки и анализа алгоритма СЗ и защищаемого ПО. Этот тип СЗ наименее стоек к атакам, так как для преодоления защиты достаточно проанализировать логику процедур проверки и должным образом их модифицировать.

Более стойкими являются системы второго типа. Для дезактивации таких защит необходимо определение ключа дешифрации ПО. Самыми стойкими к атакам являются комбинированные системы.

Для защиты ПО используется ряд методов:

1. *Алгоритмы запутывания* – используются хаотические переходы в разные части кода, внедрение ложных процедур – «пустышек», холостые циклы, искажение количества реальных параметров процедур ПО, разброс участков кода по разным областям ОЗУ и т.п.

2. *Алгоритмы мутации* – создаются таблицы соответствия операндов – синонимов и замена их друг на друга при каждом запуске программы по определенной схеме или случайным образом, случайные изменения структуры программы.

3. *Алгоритмы компрессии данных* – программа упаковывается, а затем распаковывается по мере выполнения.

4. *Алгоритмы шифрования данных* – программа шифруется, а затем расшифровывается по мере выполнения.

5. *Вычисление сложных математических выражений в процессе отработки механизма защиты* – элементы логики защиты зависят от результата вычисления значения какой-либо формулы или группы формул.

6. *Методы затруднения дизассемблирования* – используются различные приемы, направленные на предотвращение дизассемблирования в пакетном режиме.

7. *Методы затруднения отладки* – используются различные приемы, направленные на усложнение отладки программы.

8. *Эмуляция процессоров и операционных систем* – создается виртуальный процессор и/или операционная система (не обязательно существующие) и программа-переводчик из системы команд IBM в систему команд созданного процессора или ОС, после такого перевода ПО может выполняться только при помощи эмулятора, что резко затрудняет исследование алгоритма ПО.

9. *Нестандартные методы работы с аппаратным обеспечением* – модули системы защиты обращаются к аппаратуре ЭВМ, минуя процедуры ОС, и используют малоизвестные или недокументированные ее возможности.

В свою очередь, злоумышленники так же применяют ряд методов и средств для нарушения систем защиты. Ситуация противостояния разработчиков СЗПО и злоумышленников постоянно изменяется за счет комбинирования уже известных методов защиты и нападения, а так же за счет создания и использования новых методов. В целом это взаимодействие может быть описано схемой на рис. 8.1.

По *принципу функционирования СЗ* можно подразделить на следующие:

1) упаковщики/шифраторы;
2) СЗ от несанкционированного копирования;
3) СЗ от несанкционированного доступа (НСД).

8.2. ДОСТОИНСТВА И НЕДОСТАТКИ ОСНОВНЫХ СИСТЕМ ЗАЩИТЫ

Рассмотрим достоинства и недостатки основных систем защиты ПО исходя из принципов их функционирования.

8.2.1. УПАКОВЩИКИ. ШИФРАТОРЫ

Первоначально основной целью упаковщиков/шифраторов являлось уменьшение объема исполняемого модуля на диске без ущерба для функциональности программы, но позднее на первый план вышла цель защиты ПО от анализа его алгоритмов и несанкционированной модификации. Для достижения этого используются алгоритмы компрессии данных; приемы, связанные с использованием недокументированных особенностей операционных систем и процессоров; шифрование данных, алгоритмы мутации, запутывание логики программы, приведение ОС в нестабильное состояние на время работы ПО и др.

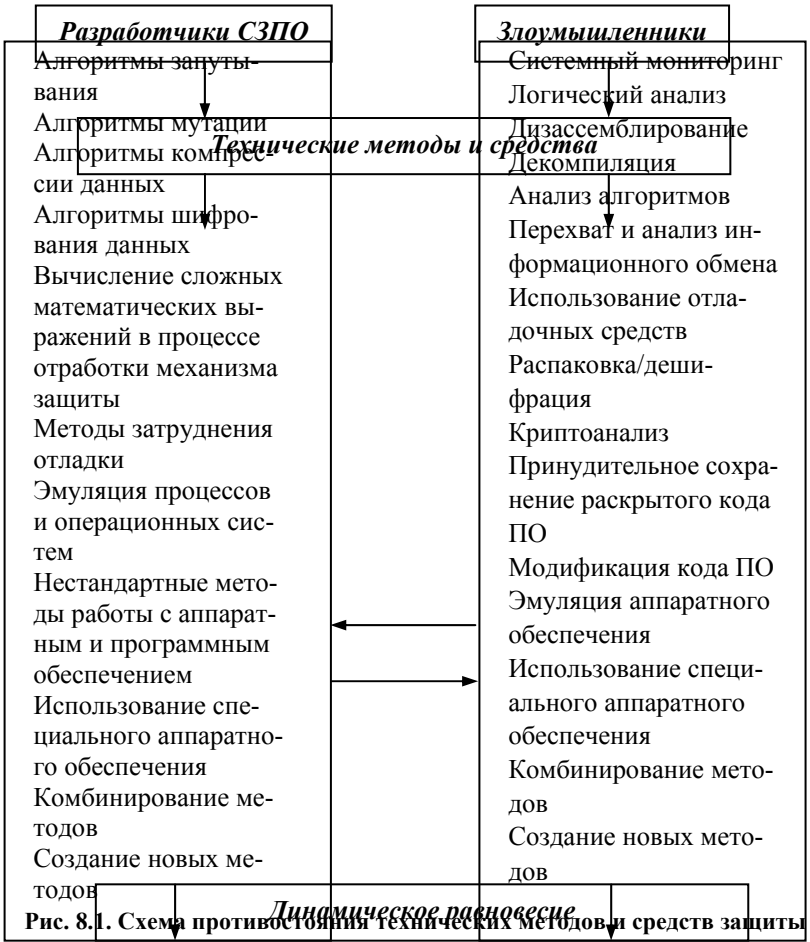


Рис. 8.1. Схема противостояния технических методов и средств защиты

Положительные стороны:

1. В рамках периода безопасного использования данные системы обеспечивают высокий уровень защиты ПО от анализа его алгоритмов.
2. Методы упаковки/шифрации намного увеличивают стойкость СЗ других типов.

Отрицательные стороны:

1. Практически все применяемы методы замедляют выполнение кода ПО.
2. Шифрование/упаковка кода ПО вызывает затруднения при обновлении (update) и исправлении ошибок (bugfix, servicepack).
3. Возможно повышение аппаратно-программных требований ПО.
4. В чистом виде данные системы не применимы для авторизации использования ПО.
5. Эти системы применимы лишь к продуктам небольшого объема.
6. Данный класс систем уязвим, так как программный код может быть, распакован или расшифрован для выполнения.
7. Обладают небольшим сроком безопасного использования, ввиду п. 4.
8. Упаковка и шифрование исполняемого кода вступает в конфликт с запрещением самомодифицирующегося кода в современных ОС.

8.2.2. СИСТЕМЫ ЗАЩИТЫ ОТ НЕСАНКЦИОНИРОВАННОГО КОПИРОВАНИЯ

Системы защиты от несанкционированного копирования осуществляют «привязку» ПО к дистрибутивному носителю (гибкий диск, CD и др.). Данный тип защит основывается на глубоком изучении работы контроллеров накопителей, их физических показателей, нестандартных режимах разбивки, чтения/записи и т.п. При этом на физическом уровне создается дистрибутивный носитель, обладающий предположительно неповторимыми свойствами (нестандартная разметка носителя информации или/и запись на него дополнительной информации – пароля или метки), а на программном – создается модуль, настроенный на идентификацию и аутентификацию носителя по его уникальным свойствам. При этом возможно применение приемов, используемых упаковщиками/шифраторами.

Положительные факторы:

1. Затруднение нелегального копирования и распространения ПО.
2. Защита прав пользователя на приобретенное ПО.

Отрицательные факторы:

1. Большая трудоемкость реализации системы защиты.
2. Замедление продаж из-за необходимости физической передачи дистрибутивного носителя информации.
3. Повышение системных требований из-за защиты (наличие накопителя).
4. Снижение отказоустойчивости ПО.
5. Несовместимость защиты и аппаратуры пользователя (накопитель, контроллер).
6. На время работы ПО занимается накопитель.
7. Угроза кражи защищенного носителя.

8.2.3. СИСТЕМЫ ЗАЩИТЫ ОТ НЕСАНКЦИОНИРОВАННОГО ДОСТУПА

Системы защиты от НСД осуществляют предварительную или периодическую аутентификацию пользователя ПО или его компьютерной системы путем запроса дополнительной информации. К этому типу СЗ можно отнести системы парольной защиты ПО, системы «привязки» ПО к компьютеру пользователя, аппаратно-программные системы с электронными ключами и системы с «ключевыми дисками». В первом случае «ключевую» информацию вводит пользователь, во втором – она содержится в уникальных параметрах компьютерной системы пользователя, в третьем – «ключевая» информация считывается с микросхем электронного ключа и в четвертом случае она хранится на диске.

8.2.3.1. Парольные защиты

На сегодняшний день этот класс СЗПО является самым распространенным. Основной принцип работы данных систем заключается в идентификации и аутентификации пользователя ПО путем запроса дополнительных данных, которыми могут быть название фирмы и/или имя и фамилия пользователя и его пароль либо только пароль/регистрационный код. Такая информация может запрашиваться в различных ситуациях, например, при старте программы, по истечении срока бесплатного использования ПО, при вызове процедуры регистрации либо в процессе установки на ПК пользователя. Процедуры парольной защиты просты в реализации. Большинство парольных СЗПО использует логические механизмы, сводящиеся к проверке правильности пароля/кода и запуске или не запуске ПО, в зависимости от результатов проверки. Существуют также системы, шифрующие защищаемое ПО и использующие пароль или производную от него

величину как ключ дешифрации. Обычно они реализованы в виде защитного модуля и вспомогательных библиотек и устанавливаются на уже скомпилированные модули ПО.

Слабым звеном парольных защит является блок проверки правильности введенного пароля/кода. Для такой проверки можно сравнивать введенный пароль с записанным в коде ПО правильным либо с правильно сгенерированным из введенных дополнительных данных паролем. Возможно также сравнение производных величин от введенного и правильного паролей, например их хэш-функций, в таком случае в коде можно сохранять только производную величину, что повышает стойкость защиты. Путем анализа процедур проверки можно найти реальный пароль, записанный в коде ПО, найти правильно сгенерированный пароль из введенных данных либо создать программу для перебора паролей для определения пароля с нужной хэш-суммой. Кроме того, если СЗПО не использует шифрования, достаточно лишь принудительно изменить логику проверки для получения беспрепятственного доступа к ПО.

Шифрующие системы более стойки к атакам, но при использовании простейших или некорректно реализованных криптоалгоритмов есть опасность дешифрации ПО.

Для всех парольных систем существует угроза перехвата пароля при его вводе авторизованным пользователем. Кроме того, в большинстве СЗПО данного типа процедура проверки используется лишь единожды, обычно при регистрации или установке ПО, затем система защиты просто отключается, что создает реальную угрозу для НСД при незаконном копировании ПО.

Положительные стороны:

1. Надежная защита от злоумышленника-непрофессионала.
2. Минимальные неудобства для пользователя.
3. Возможность передачи пароля/кода по сети.
4. Отсутствие конфликтов с системным и прикладным ПО и АО.
5. Простота реализации и применения.
6. Низкая стоимость.

Отрицательные стороны:

1. Низкая стойкость большинства систем защиты данного типа.
2. Пользователю необходимо запоминать пароль/код.

8.2.3.2. Системы «привязки» ПО

Системы этого типа при установке ПО на ПК пользователя осуществляют поиск уникальных признаков компьютерной системы либо они устанавливаются самой системой защиты. Модуль защиты в самом ПО настраивается на поиск и идентификацию данных признаков, по которым в дальнейшем определяется авторизованное или неавторизованное использование ПО, при котором возможно применение методик оценки скоростных и иных показателей процессора, материнской платы, дополнительных устройств, ОС, чтение/запись в микросхемы энергонезависимой памяти, запись скрытых файлов, настройка на наиболее часто встречаемую карту использования ОЗУ и т.п.

Слабым звеном таких защит является тот факт, что на ПК пользователя ПО всегда запускается на выполнение, что приводит к возможности принудительного сохранения ПО после отработки системы защиты, исследование самой защиты и выявление данных, используемым СЗПО для аутентификации ПК пользователя.

Положительные факторы:

1. Не требуется добавочных аппаратных средств для работы защиты.
2. Затруднение несанкционированного доступа к скопированному ПО.
3. Простота применения.
4. "Невидимость" СЗПО для пользователя.

Отрицательные факторы:

1. Ложные срабатывания СЗПО при любых изменениях в параметрах ПК.
2. Низкая стойкость при доступе злоумышленника к ПК пользователя.
3. Возможность конфликтов с системным ПО.

8.2.3.3. Программно-аппаратные средства защиты ПО с электронными ключами

В настоящее время данный класс СЗПО приобретает все большую популярность среди производителей программного обеспечения (ПО). Под программно-аппаратными средствами защиты понимаются средства, основанные на использовании так называемых «аппаратных (электронных) ключей». *Электронный ключ* – это аппаратная часть системы защиты, представляющая собой плату с микросхемами памяти и, в некоторых случаях, микропроцессором, помещенную в корпус и предназначенную для установки в один из стандартных портов ПК (COM, LPT, PCMCIA, USB) или слот расширения материнской платы. Так же в качестве такого устройства могут использоваться смарт-карты (Smart-

Card). По результатам проведенного анализа, программно-аппаратные средства защиты в настоящий момент являются одними из самых стойких систем защиты ПО от НСД.

Электронные ключи по архитектуре можно подразделить на *ключи с памятью* (без микропроцессора) и *ключи с микропроцессором* (и памятью).

Наименее стойкими являются системы с аппаратной частью первого типа. В таких системах критическая информация (ключ дешифрации, таблица переходов) хранится в памяти электронного ключа. Для дезактивации таких защит в большинстве случаев необходимо наличие у злоумышленника аппаратной части системы защиты (перехват диалога между программной и аппаратной частями для доступа к критической информации).

Наиболее стойкими являются системы с аппаратной частью второго типа. Такие комплексы содержат в аппаратной части не только ключ дешифрации, но и блоки шифрации/дешифрации данных, таким образом при работе защиты в электронный ключ передаются блоки зашифрованной информации, а принимаются оттуда расшифрованные данные. В системах этого типа достаточно сложно перехватить ключ дешифрации так как все процедуры выполняются аппаратной частью, но остается возможность принудительного сохранения защищенной программы в открытом виде после отработки системы защиты. Кроме того, к ним применимы методы криптоанализа.

Положительные факторы:

1. Значительное затруднение нелегального распространения и использования ПО.
2. Избавление производителя ПО от разработки собственной си-стемы защиты.
3. Высокая автоматизация процесса защиты ПО.
4. Наличие API системы для более глубокой защиты.
5. Возможность легкого создания демо-версий.
6. Достаточно большой выбор таких систем на рынке.

Отрицательные факторы:

1. Затруднение разработки и отладки ПО из-за ограничений со стороны СЗ.
2. Дополнительные затраты на приобретение системы защиты и обучение персонала.
3. Замедление продаж из-за необходимости физической передачи аппаратной части.
4. Повышение системных требований из-за защиты (совместимость, драйверы).
5. Снижение отказоустойчивости ПО.
6. Несовместимость систем защиты и системного или прикладного ПО пользователя.
7. Несовместимость защиты и аппаратуры пользователя.
8. Ограничения из-за несовместимости электронных ключей различных фирм.
9. Снижение расширяемости компьютерной системы.
10. Затруднения или невозможность использования защищенного ПО в переносных и блокнотных ПК.
11. Наличие у аппаратной части размеров и веса (для COM/LPT = = 5 × 3 × 2 см ~ 50 гр).
12. Угроза кражи аппаратного ключа.

8.2.3.4. Средства защиты ПО с «ключевыми дисками»

В настоящий момент этот тип систем защиты мало распространен, ввиду его морального устаревания. СЗПО этого типа во многом аналогичны системам с электронными ключами, но здесь критическая информация хранится на специальном, ключевом, носителе. Основной угрозой для таких СЗПО является перехват считывания критической информации, а так же незаконное копирование ключевого носителя.

Положительные и отрицательные стороны данного типа СЗПО практически полностью совпадают с таковыми у систем с электронными ключами.

8.3. ПОКАЗАТЕЛИ ЭФФЕКТИВНОСТИ СИСТЕМ ЗАЩИТЫ

Необходимо отметить, что пользователи явно ощущают лишь отрицательные стороны систем защит, а производители ПО рассматривают только относящиеся к ним «плюсы» и «минусы» систем защиты и практически не рассматривают факторы, относящиеся к конечному потребителю. По результатам исследований был разработан набор показателей применимости и критериев оценки СЗПО.

Показатели применимости

Технические – соответствие СЗПО функциональным требованиям производителя ПО и требованиям по стойкости, системные требования ПО и системные требования СЗПО, объем ПО и объем СЗПО, функциональная направленность ПО, наличие и тип СЗ у аналогов ПО – конкурентов.

Экономические – соотношение потерь от пиратства и общего объема прибыли, соот-

ношение потерь от пиратства и стоимости СЗПО и ее внедрения, соотношение стоимости ПО и стоимости СЗПО, соответствие стоимости СЗПО и ее внедрения поставленным целям.

Организационные – распространенность и популярность ПО, условия распространения и использования ПО, уникальность ПО, наличие угроз, вероятность превращения пользователя в злоумышленника, роль документации и поддержки при использовании ПО.

Критерии оценки

Защита как таковая – затруднение нелегального копирования и доступа, защита от мониторинга, отсутствие логических брешей и ошибок в реализации системы.

Стойкость к исследованию/взлому – применение стандартных механизмов, новые/нестандартные механизмы.

Отказоустойчивость (*надежность*) – вероятность отказа защиты (НСД), время наработки на отказ, вероятность отказа программы защиты (крах), время наработки на отказ, частота ложных срабатываний.

Независимость от конкретных реализаций ОС – использование недокументированных возможностей, «вирусных» технологий и «дыр» ОС.

Совместимость – отсутствие конфликтов с системным и прикладным ПО, отсутствие конфликтов с существующим АО, максимальная совместимость с разрабатываемым АО и ПО.

Неудобства для конечного пользователя ПО – необходимость и сложность дополнительной настройки системы защиты, доступность документации, доступность информации об обновлении модулей системы защиты из-за ошибок/несовместимости/нестойкости, доступность сервисных пакетов, безопасность сетевой передачи пароля/ключа, задержка из-за физической передачи пароля/ключа, нарушения прав потребителя.

Побочные эффекты – перегрузка трафика, отказ в обслуживании, замедление работы защищаемого ПО и ОС, захват системных ресурсов, перегрузка ОЗУ, нарушение стабильности ОС.

Стоимость – стоимость/эффективность, стоимость/цена защищаемого ПО, стоимость/ликвидированные убытки.

Доброкачественность – доступность результатов независимой экспертизы, доступность информации о побочных эффектах, полная информация о СЗ для конечного пользователя.

Общая картина взаимодействия агентов рынка программного обеспечения представлена на схеме на рис. 8.2. Из четырех указанных выше видов среды взаимодействия защищающейся стороне подконтрольны (или частично подконтрольны) три вида – организационная, техническая и экономическая среда. Важнейшей средой взаимодействия является несомненно экономическая среда, так как экономическое взаимодействие, в данном случае, является первопричиной и целью всего взаимодействия.

При разработке и анализе защиты программного обеспечения необходимо учитывать существующую законодательную базу, при этом нужно проводить подробный экономический анализ ситуации, применяя различные критерии оценки, а затем создавать стратегию защиты, включающую применение технических и организационных мер защиты программного обеспечения.

Контрольные вопросы

1. Приведите классификацию систем защиты программного обеспечения.
2. Сравните основные технические методы и средства защиты программного обеспечения.
3. Назовите отличия систем защиты от несанкционированного доступа от систем защиты от несанкционированного копирования.
4. Дайте характеристику показателей эффективности систем защиты.
5. Приведите примеры взаимодействия участников процесса создание и распространения ПО.

9. ЗАЩИТА ИНФОРМАЦИИ В КОРПОРАТИВНЫХ СЕТЯХ

9.1. ОСНОВЫ И ЦЕЛЬ ПОЛИТИКИ БЕЗОПАСНОСТИ В КОМПЬЮТЕРНЫХ СЕТЯХ

Межсетевой экран (МЭ) или брандмауэр (Firewall) – это средство защиты, которое можно использовать для управления доступом между надежной сетью и менее надежной. Межсетевой экран – это не одна компонента, а стратегия защиты ресурсов организации, доступных из глобальной сети.

Основная функция МЭ – централизация управления доступом. Если удаленные пользователи могут получить доступ к внутренним сетям в обход МЭ, его эффективность близка к нулю. МЭ обычно используются для защиты сегментов локальной сети организации.

Межсетевые экраны обеспечивают несколько типов защиты:

- блокирование нежелательного трафика;
- перенаправление входного трафика только к надежным внутренним системам;
- сокрытие уязвимых систем, которые нельзя обезопасить от атак из глобальной сети другим способом;
- протоколирование трафика в и из внутренней сети;
- сокрытие информации (имен систем, топологии сети, типов сетевых устройств и внутренних идентификаторов пользователей, от внешней сети;
- обеспечение более надежной аутентификации, чем та, которую представляют

стандартные приложения.

Как и для любого средства защиты, нужны определенные компромиссы между удобством работы и безопасностью. *Прозрачность* – это видимость МЭ как внутренним пользователям, так и внешним, осуществляющим взаимодействие через МЭ, который прозрачен для пользователей, если он не мешает им получить доступ к сети. Обычно МЭ конфигурируются так, чтобы быть прозрачными для внутренних пользователей сети (посылающим пакеты наружу), и, с другой стороны, МЭ конфигурируется так, чтобы быть непрозрачным для внешних пользователей, пытающихся получить доступ к внутренней сети извне. Это обычно обеспечивает высокий уровень безопасности и не мешает внутренним пользователям.

Система защиты доступа в ИВС должна включать три пояса защиты:

1) пояс, охватывающий территорию, на которой расположены элементы ИВС;

2) пояс, который охватывает сооружения и помещения с аппаратурой ИВС;

3) пояс, охватывавший ресурсы ИВС.

Защита доступа в первых двух поясах обеспечивается применением физических средств защиты, в третьем поясе – аппаратных, программных (программно-аппаратных) и криптографических средств защиты.

Под управлением доступом понимается процесс регулирования использования ресурсов ИВС.

Управление доступом включает решение следующих задач:

- идентификацию пользователей, персонала и ресурсов ИВС;
- установление подлинности субъектов и объектов, допускаемых к использованию ресурсов ИВС;
- проверку полномочий субъектов на доступ к защищаемым ресурсам;
- регистрацию (протоколирование) обращений к защищаемым ресурсам;
- реакцию на несанкционированные действия.

9.2. УПРАВЛЕНИЕ ДОСТУПОМ

9.2.1. ИДЕНТИФИКАЦИЯ И УСТАНОВЛЕНИЕ ПОДЛИННОСТИ

Для того чтобы установить подлинность субъектов и объектов системы, все субъекты и объекты, зарегистрированные в системе, должны иметь уникальные имена – идентификаторы. Когда какой-либо субъект обращается к ресурсам системы, необходимо установить его подлинность, опознать его (процесс авторизации или аутентификации).

Установление подлинности субъекта (объекта) заключается в подтверждении того, что обратившийся субъект (вызываемый объект) является именно тем, которому разрешено участвовать в данном процессе (выполнять действия).

В зависимости от сложности установления подлинности различают три основные группы операций: простое, усложненное и особое установление подлинности.

Простое установление подлинности сводится к сравнению предъявленного кода (характеристики) с эталонным кодом, который хранится в памяти устройства, выполняющего установление подлинности.

Усложненное установление подлинности требует от пользователя ввода дополнительной информации и осуществляется в режиме диалога.

Особое установление подлинности, кроме использования методов простого и усложненного установления подлинности, использует специальную совокупность опознавательных характеристик, которая выбирается для обеспечения надежного установления подлинности.

9.2.1.1. Установление подлинности субъектов

Для установления подлинности субъектов используются различные опознавательные характеристики. Классификация характеристик, применяемых для установления подлинности субъектов, приведена на рис. 9.1.

В литературе описаны устройства установления подлинности субъектов в реальном масштабе времени по почерку, голосу и отпечаткам пальцев.

Установление подлинности по почерку производится, например, с помощью специальной ручки-датчика. При этом используются методы сопоставления контуров, анализа специфических штрихов и гистограмм.

При установлении подлинности по голосу используются следующие параметры: тембр, высота звука, акцент, интонация, сила звука и скорость речи, основано на спектральных методах и не зависит от содержания речи.

Установление подлинности по отпечаткам пальцев производится путем сличения предъявленных отпечатков пальцев с эталонными. Устройство использует методы сопоставления бинарных образов и проекций для характерных точек и направлений штрихов

отпечатков пальцев.

Некоторые производители реализуют система установления подлинности на базе пластиковых карт, на которые кодовая информация записывается и считывается лазерно-голографическими методами. Такие карты могут использоваться в двух режимах: ключа и персонального идентификационного кода (ПИК). В режиме ключа карта служит для открывания специальных голографических электронно-механических замков, устанавливаемых на защищаемых объектах. В режиме ПИК карта используется для ограничения доступа к терминалам вычислительной системы и хранящимся в ней данным. Для этого на карту заносится ПИК пользователя, занимающий от 64 до 256 бит.

9.2.1.2. Методы паролирования

Методы паролирования требуют, чтобы пользователь ввел строку символов (пароль) для сравнения с эталонным паролем, хранящимся в памяти. При соответствии пароля эталонному пользователю разрешена работа с системой.

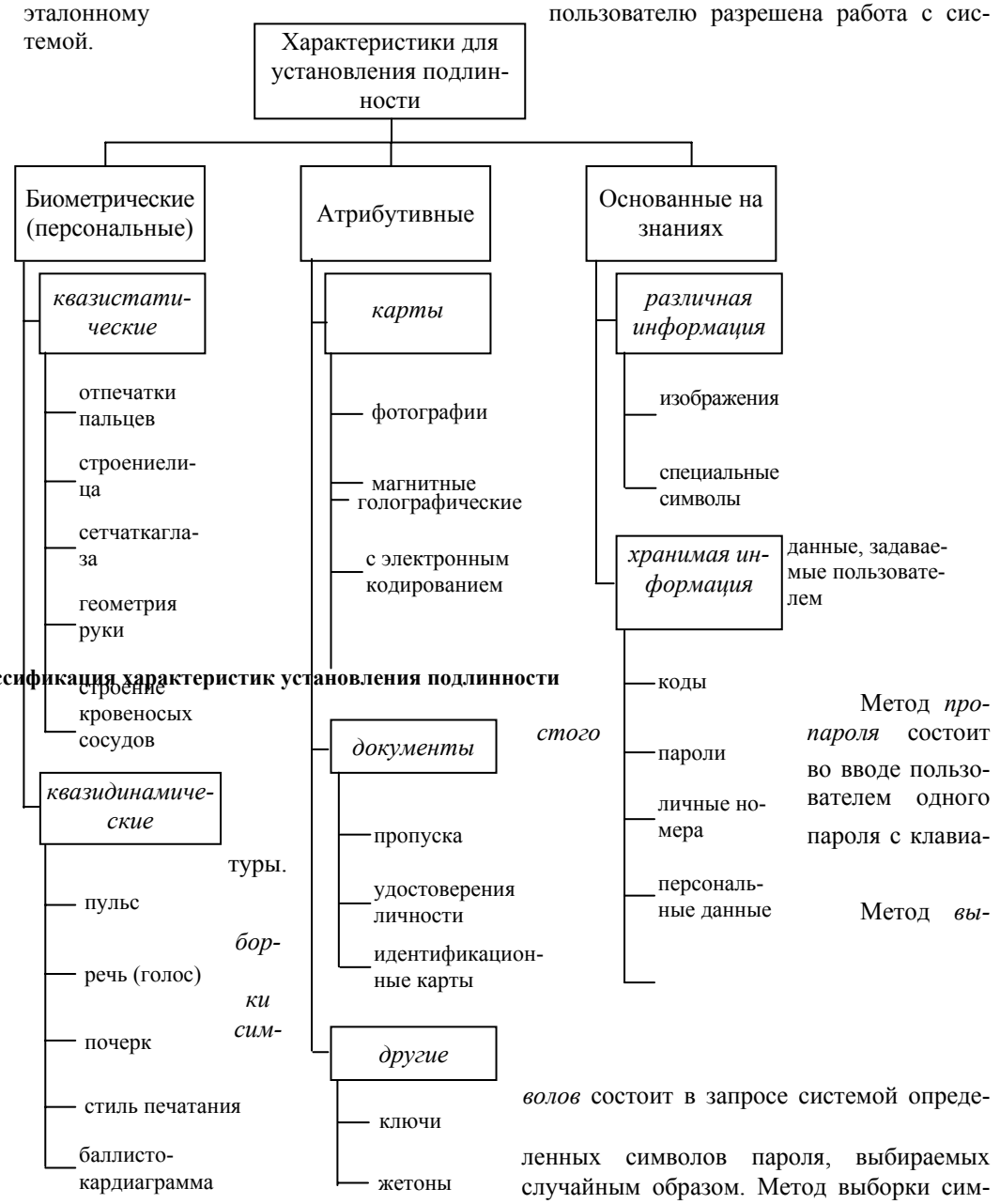


Рис. 9.1. Классификация характеристик установления подлинности

Метод *простого пароля* состоит во вводе пользователем одного пароля с клавиатуры.

Метод *выборки символов* состоит в запросе системой определенных символов пароля, выбираемых случайным образом. Метод выборки символов не позволяет нарушителю определить значение пароля по однократному наблюдению вводимых пользователем символов.

Метод *паролей однократного использования* предполагает наличие списка из N паролей, хранящегося в системе. При каждом обращении к системе пользователь вводит очередной пароль, который после окончания работы вычеркивается системой из списка. Основным недостатком рассмотренного метода является неоднозначность пароля.

Метод *групп паролей* основывается на том, что система для каждого пользователя может потребовать пароли из двух групп. Группы могут включать пароли, которые являются:

– ответами на общие для всех пользователей вопросы, например, имя, адрес, номер

телефона и т.п.;

– ответами на вопросы, которые устанавливаются администратором системы при регистрации персонально для каждого пользователя для работы с системой, например, любимый цвет, девичья фамилия матери и т.п.

При каждом обращении пользователя система случайно выбирает по нескольку вопросов из каждой группы.

Метод *функционального преобразования* предполагает, что пользователю при регистрации для работы в системе сообщается некоторое преобразование, которое он может выполнить в уме. Для усложнения вскрытия пароля в методе функционального преобразования в качестве аргументов могут использоваться числа месяца, часы суток или их комбинации.

При работе с паролями должны соблюдаться следующие правила:

– пароли должны храниться в памяти только в зашифрованном виде;

– символы пароля при вводе их пользователем не должны появляться в явном виде;

– пароли должны периодически меняться;

– пароли не должны быть простыми.

Для проверки сложности паролей обычно используют специальные контроллеры паролей, которые позволяет проверить уязвимость паролей. Контроллер осуществляет попытки взлома пароля по следующей методике.

1. Проверка использования в качестве пароля входного имени пользователя, его инициалов и их комбинаций.

2. Проверка использования в качестве пароля слов из различных словарей (60 000 слов): мужские и женские имена (16 000 имен); названия стран и городов; имена персонажей мультфильмов, кинофильмов, научно – фантастических произведений и т.п.; спортивные термины (названия команд, имена спортсменов, спортивный жаргон и т.п.); числа (цифрами и прописью, например, 2000, TWELVE); строки букв и цифр (например, AA, AAA, AAAA и т.д.); библейские имена и названия; биологические термины; жаргонные слова и ругательства; последовательности символов в порядке их расположения на клавиатуре (например, QWERTY, ASDF, ZXCVBN и т.д.); имена компьютеров (из файла /etc/hostc в ОС UNIX); персонажи и места действия из произведений Шекспира; часто употребляемые иностранные слова; названия астероидов.

3. Проверка различных перестановок слов из п. 2, включая: замену первой буквы на прописную; замену всех букв на прописные; инверсию всего слова; замену буквы О на цифру 0 и наоборот (цифру 1 на букву 1 и т.д.); превращение слов во множественное число.

Всего по п. 3 контроллер осуществляет проверку на совпадение приблизительно с одним миллионом слов.

4. Проверка различных перестановок слов из п. 2, не рассмотренных в п. 3: замена одной строчной буквы на прописную (около 400 000 слов); замена двух строчных букв на прописные (около 1 500 000 слов); замена трех строчных букв на прописные и т.д.

5. Для иностранных пользователей проверка слов на языке пользователя.

6. Проверка пар слов.

Проведенные эксперименты показали, что данный контроллер позволил определить 10 % паролей из пяти символов, 35 % паролей из шести символов, 25 % паролей из семи символов и 23 % паролей из восьми символов.

Приведенные примеры позволяют сформулировать следующие способы снижения уязвимости паролей:

– не использовать в качестве пароля слова, проверяемые контроллером Кляйна;

– проверять пароли перед их использованием контроллерами паролей;

– часто менять пароли;

– при формировании пароля использовать знаки препинания и различные регистры;

– использовать не осмысленные слова, а наборы букв (например, первых букв какой-нибудь известной пользователю фразы).

Из примеров, приведенных при рассмотрении контроллера паролей, видно, что важнейшими характеристиками пароля являются его длина и период смены (или период жизни). Естественно, что чем больше длина пароля, тем больше усилий придется приложить нарушителю для его определения. Чем больше период жизни пароля, тем более вероятно его раскрытие.

Для случая, когда пользователь вводит пароль через удаленный терминал, можно применить формулу Андерсена:

$$4{,}32 \cdot 10^4 (\mathrm{v}T)/(NP) <= A^s,$$

где v – скорость передачи данных через линию связи (в символах/мин); T – период времени, в течение которого могут быть предприняты попытки отгадывания пароля (в месяцах при работе 24 ч/сутки); N – число символов в каждом передаваемом сообщении при попытке получить доступ к системе; P – вероятность подбора нарушителем правильного пароля; A – число символов в алфавите, из которого составляется пароль; S – длина пароля (в символах).

9.2.1.3. Установление подлинности объектов

Одной из возможных стратегий действий нарушителя в ИВС является подключение к каналу связи. В этом случае нарушитель может имитировать механизм установления подлинности, что позволит ему получить пароль пользователя и доступ к его данным. Для предупреждения подобных действий нарушителя пользователь должен убедиться в подлинности системы, с которой он начинает работать.

Одним из методов решения этой задачи является так называемая процедура «рукопожатия». Для осуществления процедуры «рукопожатия» выбирается нетривиальное преобразование вида

$$y = f(x, k),$$

где x – аргумент; k – коэффициент. В качестве аргумента преобразования можно использовать элементы даты, времени и т.п. Преобразование y известно только пользователям и ЭВМ и должно сохраняться в тайне. Пользователь вместе с запросом на подключение к системе посылает выбранное им значение x. Получив значение x вместе с идентификатором пользователя, система вычисляет $y = f(x, k)$ и посылает его пользователю вместе с запросом о вводе пароля. Пользователь вычисляет или имеет вычисленное заранее значение y. Если значения y, полученные пользователем и системой, совпадают, то режим опознания системы заканчивается, и пользователь может вводить пароль. После подтверждения правильности пароля пользователя считается, что «рукопожатие» состоялось.

Аналогичным образом осуществляется установление подлинности ЭВМ ИВС при необходимости обмена данных между ними.

Проверка подлинности взаимодействующих субъектов и объектов системы может производиться не только перед началом сеанса, но и в ходе него. Такие проверки могут осуществляться через определенные промежутки времени, после определенного количества переданных данных и т.п.

9.2.2. ПРОВЕРКА ПОЛНОМОЧИЙ СУБЪЕКТОВ НА ДОСТУП К РЕСУРСАМ

После положительного установления подлинности пользователя (и системы со стороны пользователя) система должна осуществлять постоянную проверку полномочий поступающих от субъектов запросов. Проверка полномочий заключается в определении соответствия запроса субъекта предоставленным ему правам доступа к ресурсам. Такую процедуру часто называют «контроль полномочий» или «контроль доступа». Проверка полномочий основывается на различных методах разграничения доступа, которые были рассмотрены ранее.

9.2.3. РЕГИСТРАЦИЯ ОБРАЩЕНИЙ К ЗАЩИЩАЕМЫМ РЕСУРСАМ

Регистрация (протоколирование) обращений к защищенным ресурсам системы позволяет должностному лицу, ответственному за информационную безопасность, следить за использованием ресурсов и оперативно принимать меры по перекрытию обнаруженных каналов утечки данных. Все обращения к ресурсам системы должны фиксироваться в регистрационном журнале.

В регистрационный журнал обычно заносятся следующие данные:

- обращения (доступы) к защищаемым ресурсам;
- отказы в доступе;
- изменения полномочий;
- случаи неиспользования пользователями разрешенных запросов;
- изменения содержания памяти ЭВМ, производимые пользователями;
- любые подозрительные действия.

Типовая форма записи регистрационного журнала представлена в табл. 9.1.

9.1. Типовая форма записи регистрационного журнала

Тип записи	Дата	Время	Терминал	Пользователь	Событие

В системе должна быть предусмотрена возможность выводить содержимое регистрационного журнала на экран терминала и печатающее устройство, причем выводимую информацию необходимо сортировать по пользователям, терминалам, датам, идентификаторам заданий, элементам данных и т.п.

Следует отметить, что регистрационный журнал может быть также использован для решения следующих задач:

- настройка системы (по частоте обращений к различным ресурсам);
- помощь пользователям в случае их непреднамеренных ошибок;
- изменение полномочий пользователей (если пользователи ча-сто совершают ошибки, либо вообще никогда не обращаются к некоторым ресурсам);
- возврат системы в исходное состояние для восстановления;
- психологическое воздействие на потенциальных нарушителей.

Приведенный перечень задач, для решения которых может быть использован регистрационный журнал, еще раз подтверждает необходимость комплексного применения всех средств и механизмов защиты для обеспечения безопасности данных.

9.2.4. РЕАГИРОВАНИЕ НА НЕСАНКЦИОНИРОВАННЫЕ ДЕЙСТВИЯ

Реагирование на несанкционированные действия включает в себя:

- сигнализацию о НСД;
- блокировку (отключение терминала, группы терминалов, элементов ИВС и т.п.);
- задержку в работе;
- отказ в запросе;
- имитацию выполнения запрещенного действия для определения места подключения нарушителя и характера его действий.

Реагирование на НСД может осуществляться автоматически и с участием должностного лица, ответственного за информационную безопасность.

9.3. МНОГОУРОВНЕВАЯ ЗАЩИТА КОРПОРАТИВНЫХ СЕТЕЙ

9.3.1. АУТЕНТИФИКАЦИЯ

Межсетевые экраны (МЭ) на основе маршрутизаторов не обеспечивают аутентификации пользователей. МЭ, в состав которых входят прокси-сервера, обеспечивают следующие типы аутентификации.

Имя/*пароль* – это самый плохой вариант, так как эта информация может быть перехвачена в сети или получена путем подглядывания за ее вводом из-за спины и еще тысячей других способов.

Одноразовые пароли – используют программы или специальные устройства для генерации нового пароля для каждого сеанса. Это означает, что старые пароли не могут быть повторно использованы, если они были перехвачены в сети или украдены другим способом.

Электронные сертификаты – используют шифрование с открытыми ключами.

9.3.2. АНАЛИЗ ВОЗМОЖНОСТЕЙ МАРШРУТИЗАЦИИ И ПРОКСИ-СЕРВЕРОВ

В политике безопасности должно быть отражено, может ли МЭ маршрутизировать пакеты или они должны передаваться прокси-серверам. Тривиальным случаем МЭ является маршрутизатор, который может выступать в роли устройства для фильтрации пакетов. Все, что он может – только маршрутизировать пакеты. А прикладные шлюзы, наоборот, не могут быть сконфигурированы для маршрутизации трафика между внутренним и внешним интерфейсами МЭ, так как это может привести к обходу средств защиты. Все соединения между внешними и внутренними хостами должны проходить через прикладные шлюзы (прокси-сервера).

9.3.2.1. Маршрутизация источника

Маршрутизация источника – это механизм маршрутизации, посредством которого путь к машине-получателю пакета определяется отправителем, а не промежуточными

маршрутизаторами. Маршрутизация источника, в основном, используется для устранения проблем в сетях, но также может быть использована для атаки на хост. Если атакующий знает, что ваш хост доверяет какому-нибудь другому хосту, то маршрутизация источника может быть использована для создания впечатления, что пакеты атакующего приходят от доверенного хоста. Поэтому из-за такой угрозы безопасности маршрутизаторы с фильтрацией пакетов обычно конфигурируются так, чтобы отвергать пакеты с опцией маршрутизации источника. Поэтому сайт, желающий избежать проблем с маршрутизацией источника, обычно разрабатывает политику, в которой их маршрутизация запрещена.

9.3.2.2. Фальсификация IP-адреса

Фальсификация IP-адреса имеет место, когда атакующий маскирует свою машину под хост в сети объекта атаки (то есть пытается заставить цель атаки думать, что пакеты приходят от доверенной машины во внутренней сети). Политика в отношении маршрутизации пакетов должна быть четкой, чтобы можно было корректно построить обработку пакетов, если есть проблемы с безопасностью. Необходимо объединить аутентификацию на основе адреса отправителя с другими способами, чтобы защитить вашу сеть от атак подобного рода.

9.3.3. ТИПЫ МЕЖСЕТЕВЫХ ЭКРАНОВ

Существует несколько различных реализаций брандмауэров, которые могут быть созданы разными путями. Далее будет приведена краткая характеристика нескольких архитектур брандмауэров и их применимость к средам с низким, средним и высоким рискам.

9.3.3.1 Шлюзы с фильтрацией пакетов

Межсетевые экраны с фильтрацией пакетов используют маршрутизаторы с правилами фильтрации пакетов для предоставления или запрещения доступа на основе адреса отправителя, адреса получателя и порта. Они обеспечивают минимальную безопасность за низкую цену, и это может оказаться приемлемым для среды с низким риском. Они являются быстрыми, гибкими и прозрачными. Правила фильтрации часто нелегко администрировать, но имеется ряд средств для упрощения задачи создания и поддержания правил.

Шлюзы с фильтрацией имеют свои недостатки, включая следующие:

- адреса и порты отправителя и получателя, содержащиеся в заголовке IP-пакета, – единственная информация, доступная маршрутизатору при принятии решения: разрешать или запрещать доступ трафика во внутреннюю сеть;
- они не защищают от фальсификации IP- и DNS-адресов;
- атакующий получит доступ ко всем хостам во внутренней сети после того, как ему был предоставлен доступ МЭ;
- усиленная аутентификация пользователя не поддерживается некоторыми шлюзами с фильтрацией пакетов;
- практически отсутствуют средства протоколирования доступа к сети.

9.3.3.2. Прикладные шлюзы

Прикладной шлюз использует программы (называемые прокси-серверами), запускаемые на МЭ. Эти прокси-сервера принимают запросы извне, анализируют их и передают безопасные запросы внутренним хостам, которые предоставляют соответствующие сервисы. Прикладные шлюзы могут обеспечивать такие функции, как аутентификация пользователей и протоколирование их действий.

Прикладной шлюз считается самым безопасным типом МЭ. При этом он имеет ряд преимуществ:

– может быть сконфигурирован как единственный хост, видимый из внешней сети, что потребует осуществлять все внешние соединения через него;

– использование прокси-серверов для различных сервисов предотвращает прямой доступ к этим сервисам, защищая от атак небезопасные или плохо сконфигурированные внутренние хосты;

– с помощью прикладных шлюзов может быть реализована усиленная аутентификация;

– прокси-сервера могут обеспечивать детальное протоколирование на прикладном уровне.

Межсетевые экраны прикладного уровня должны конфигурироваться так, чтобы весь выходящий трафик казался исходящим от МЭ. Таким образом будет запрещен прямой доступ ко внутренним сетям. Все входящие запросы различных сетевых сервисов,

таких как Telnet, FTP, HTTP, RLOGIN, и т.д., независимо от того, какой внутренний хост запрашивается, должны проходить через соответствующий прокси-сервер на МЭ.

Прикладные шлюзы требуют прокси-сервера для каждого сервиса, такого как FTP, HTTP и т.д., поддерживаемого МЭ. Когда требуемый сервис не поддерживается прокси, у организации имеется три варианта действий:

- отказаться от использования этого сервиса, пока производитель брандмауэра не разработает для него безопасный прокси-сервер (многие новые сервисы имеют большое число уязвимых мест);
- разработать свой прокси – это достаточно сложная задача и должна решаться только техническими организациями, имеющими соответствующих специалистов;
- пропустить сервис через МЭ – большинство МЭ с прикладными шлюзами позволяет пропускать большинство сервисов с минимальной фильтрацией пакетов.

Низкий риск. Когда для входящих сервисов внешней сети нет прокси-сервера, но требуется пропускать его через МЭ, администратор МЭ должен использовать конфигурацию или «заплатку», которая позволит использовать требуемый сервис.

Средний-высокий. Все входящие сервисы внешней сети должны обрабатываться прокси-сервером на МЭ. Если требуется использование нового сервиса, то его использование должно быть запрещено до тех пор, пока производитель МЭ не разработает для него прокси-сервер и он не будет протестирован администратором МЭ. Только по специальному разрешению руководства можно разрабатывать свой прокси-сервер или закупать его у других производителей.

9.3.3.3. Гибридные или сложные шлюзы

Гибридные шлюзы объединяют в себе два описанных выше типа МЭ и реализуют их последовательно, а не параллельно. Если они соединены последовательно, то общая безопасность увеличивается, с другой стороны, если их использовать параллельно, то общая безопасность системы будет равна наименее безопасному из используемых методов. В средах со средним и высоким риском гибридные шлюзы могут оказаться идеальной реализацией.

В табл. 9.2 приводятся рейтинги и риски безопасности различных типов МЭ.

9.2. Рейтинги и риски безопасности межсетевых экранов

Архитектура МЭ	Среда с BP	Среда со CP	Среда с HP
Фильтрация пакетов	0	1	4
Прикладные шлюзы	3	4	2
Гибридные шлюзы	4	3	2

Примечание: 4 – рекомендованный вариант; 3 – эффективный вариант; 2 – допустимый вариант; 1 – минимальная безопасность; 0 – неприемлемо.

Контрольные вопросы

1. Поясните цели политики безопасности в ИВС.
2. Перечислите пояса защиты доступа в ИВС. Какие методы и средства защиты в них используются?
3. Предложите методы паролирования, которые обеспечивают надежное установление подлинности.
4. Какие группы операций установления подлинности Вам известны?
5. Приведите классификацию характеристик установления подлинности.
6. В чем проявляется многоуровневая защита ИВС?

10. ЗАЩИТА ОТ ИНФОРМАЦИОННЫХ ИНФЕКЦИЙ. ВИРУСОЛОГИЯ

Информационные инфекции специфически ориентированы и обладают определенными чертами: противоправны (незаконны), способны самовосстанавливаться и размножаться, а также имеют определенный инкубационный период – замедленное время начала действия.

Информационные инфекции имеют злонамеренный характер: их действия могут иметь деструктивный результат (уничтожения набора данных), реже физическое уничтожение (резкое включение и выключение дисковода), сдерживающее действие (переполнение канала ввода-вывода, памяти), или просто видоизменяющее влияние на работу программ.

Самовосстановление и размножение приводит к заражению других программ и распространению по линиям связи.

Замедленное действие проявляется в том, что работа программы начинается на оп-

ределенных условиях: дата, час, продолжительность, наступление события и т.д.

В зависимости от механизма действия информационные инфекции делятся на:

Вирус представляет собой программу, которая обладает способностью размножаться и самовостанавливаться.

Логические бомбы представляют собой программы или их части, резидентно находящиеся в ИВС и запускаемые всякий раз, когда выполняются определенные условия.

Троянский конь – это программы, полученные путем явного изменения или добавления команд в программы пользователя и способные вмешиваться в процесс обработки информации.

Червь представляет собой паразитный процесс, который потребляет (истощает) ресурсы системы и способен перемещаться в ИВС или сети и самовоспроизводить копии.

10.1. КЛАССИФИКАЦИЯ КОМПЬЮТЕРНЫХ ВИРУСОВ

Вирусы можно разделить на классы по следующим основным признакам:

- среда обитания;
- операционная система (ОС);
- особенности алгоритма работы;
- деструктивные возможности.

В зависимости от *среды обитания* вирусы можно разделить на:

– файловые;
– загрузочные;
– макровирусы;
– сетевые.

Файловые вирусы либо различными способами внедряются в выполняемые файлы, либо создают файлы-двойники (вирусы-компаньоны), либо используют особенности организации файловой системы (link-вирусы).

Загрузочные вирусы записывают себя либо в загрузочный сектор диска (boot-сектор), либо в сектор, содержащий системный загрузчик винчестера (Master Boot Record), либо меняют указатель на активный boot-сектор.

Макровирусы заражают файлы-документы и электронные таблицы нескольких популярных редакторов.

Сетевые вирусы используют для своего распространения протоколы или команды компьютерных сетей и электронной почты.

Существует большое количество сочетаний, например файлово-загрузочные вирусы, заражающие как файлы, так и загрузочные сектора дисков. Такие вирусы, имеют довольно сложный алгоритм работы, часто применяют оригинальные методы проникновения в систему, используют «стелс–» и полиморфик-технологии. Другой пример такого сочетания – сетевой макровирус, который не только заражает редактируемые документы, но и рассылает свои копии по электронной почте.

Заражаемая *операционная система* является вторым уровнем деления вирусов на классы. Каждый файловый или сетевой вирус заражает файлы какой-либо одной или нескольких ОС – DOS, Windows 95/98/Me/NT/2000/XP, OS/2, UNIX и т. д. Макровирусы заражают файлы форматов Word, Excel, других приложений Microsoft Office. Загрузочные вирусы ориентированы на конкретные форматы расположения системных данных в загрузочных секторах дисков.

Среди *особенностей алгоритма работы* вирусов выделяются следующие:

- резидентность;
- использование «стелс»-алгоритмов;
- самошифрование и полиморфичность;
- использование нестандартных приемов.

Резидентный вирус при инфицировании компьютера оставляет в оперативной памяти свою резидентную часть, которая затем перехватывает обращения ОС к объектам заражения и внедряются в них. Резидентные вирусы находятся в памяти и являются активными вплоть до выключения компьютера или перезагрузки ОС. Нерезидентные вирусы не заражают память компьютера и сохраняют активность ограниченное время. Некоторые вирусы оставляют в оперативной памяти небольшие резидентные программы, которые не распространяют вирус. Такие вирусы считаются нерезидентными.

Резидентными можно считать макровирусы, поскольку они также присутствуют в памяти компьютера в течение всего времени работы зараженного редактора. При этом роль ОС берет на себя редактор, а понятие «перезагрузка операционной системы» трактуется как выход из редактора.

Использование «*стелс*»-*алгоритмов* позволяет вирусам полностью или частично скрыть себя в системе. Наиболее распространенным «стелс»-алгоритмом является перехват запросов ОС на чтение-запись зараженных объектов и затем «стелс»-вирусы либо

временно лечат их, либо подставляют вместо себя незараженные участки информации. В случае макровирусов наиболее популярный способ – запрет вызовов меню просмотра макросов.

Самошифрование и *полиморфичность* используются практически всеми типами вирусов для того, чтобы максимально усложнить процедуру обнаружения вируса. Полиморфик – вирусы (polymorphic) достаточно трудно поддаются обнаружению; они не имеют сигнатур, т.е. не содержат ни одного постоянного участка кода. В большинстве случаев два образца одного и того же полиморфик – вируса не будут иметь ни одного совпадения. Это достигается шифрованием основного тела вируса и модификациями программы – расшифровщика.

Различные *нестандартные приемы* часто используются в вирусах для того, чтобы как можно глубже спрятать себя в ядре ОС (как это делает вирус «ЗАРАЗА»), защитить от обнаружения свою резидентную копию (вирусы TPVO, Trout2), затруднить лечение от вируса (например, помешают свою копию в Flash-BIOS) и т.д.

По *деструктивным возможностям* вирусы можно разделить на:

- безвредные, т.е. никак не влияющие на работу компьютера (кроме уменьшения свободной памяти на диске при своем распространении);
- неопасные, влияние которых ограничивается уменьшением свободной памяти на диске и графическими, звуковыми и прочими эффектами;
- опасные вирусы, которые могут привести к серьезным сбоям в работе компьютера;
- очень опасные – в алгоритм их работы заведомо заложены деструктивные процедуры (вызывающие потерю программ, уничтожение данных, или способствующие быстрому износу движущихся частей механизмов).

Прочие вредные программы. К вредным программам помимо вирусов относятся также «троянские кони», «логические бомбы», intended-вирусы, конструкторы вирусов и полиморфик–генераторы.

«*Троянский конь*» (*логические бомбы*) – это программа, наносящая какие-либо разрушительные действия в зависимости от определенных условий или при каждом запуске, уничтожая информацию на дисках, «приводящая» систему к зависанию и т.п. Большинство известных инфекций такого рода подделываются под какие-либо полезные программы, новые версии популярных утилит или дополнения к ним. Очень часто они рассылаются по BBS-станциям или электронным конференциям. По сравнению с вирусами «троянские кони» не получают широкого распространения по достаточно простым причинам: они либо уничтожают себя вместе с остальными данными на диске, либо демаскируют свое присутствие и уничтожаются пострадавшим пользователем.

К «троянским коням» также можно отнести «дропперы» вирусов – зараженные файлы, код которых подправлен таким образом, что известные версии антивирусов не определяют вируса в файле. Например, файл шифруется каким-либо специальным образом или упаковывается редко используемым архиватором, что не позволяет антивирусу «увидеть» заражение.

Следует отметить также «злые шутки» (hoax). К ним относятся программы, которые не причиняют компьютеру какого-либо прямого вреда, однако выводят сообщения о том, что такой вред уже причинен, либо будет причинен при каких-либо условиях, либо предупреждают пользователя о несуществующей опасности. К «злым шуткам» относятся, например, программы, которые «пугают» пользователя сообщениями о форматировании диска, определяют вирусы в незараженных файлах, выводят странные вирусоподобные сообщения и т.д.

Intended-вирусы. К таким вирусам относятся программы, которые на первый взгляд являются стопроцентными вирусами, но не способны размножаться по причине ошибок. Например, вирус, который при заражении «забывает» поместить в начало файлов команду передачи управления на код вируса, либо записывает в нее неверный адрес своего кода, либо неправильно устанавливает адрес перехватываемого прерывания (что в подавляющем большинстве случаев завешивает компьютер) и т.д.

К категории intended-вирусов также относятся вирусы, которые по приведенным выше причинам размножаются только один раз из «авторской» копии. Заразив какой-либо файл, они теряют способность к дальнейшему размножению.

Конструкторы вирусов – это утилита, предназначенная для изготовления новых компьютерных вирусов. Известны конструкторы вирусов для DOS, Windows и макровирусов. Они позволяют генерировать исходные тексты вирусов (ASM-файлы), объектные модули и/или непосредственно зараженные файлы.

Некоторые конструкторы снабжены стандартным оконным интерфейсом, позволяющим при помощи системы меню выбрать тип вируса, поражаемые объекты (COM и/или EXE), наличие или отсутствие самошифровки, противодействие отладчику, внут-

ренние текстовые строки, выбрать эффекты, сопровождающие работу вируса, и т. п. Прочие конструкторы не имеют интерфейса и считывают информацию о типе вируса из конфигурационного файла.

Полиморфные генераторы, как и конструкторы вирусов, не являются вирусами в прямом смысле этого слова, поскольку в их алгоритм не закладываются функции размножения, т.е. открытия, закрытия и записи в файлы, чтения и записи секторов и т.д. Главной функцией подобного рода программ является шифрование тела вируса и генерация соответствующего расшифровщика.

Резидентные вирусы. Под термином «резидентность» (DOS'овский термин TSR – Terminate and Stay Resident) понимается способность вирусов оставлять свои копии в операционной системе, перехватывать некоторые события (например, обращения к файлам или дискам) и вызывать при этом процедуры заражения обнаруженных объектов (файлов и секторов). Таким образом, резидентные вирусы активны не только в момент работы зараженной программы, но и после того как программа закончила свою работу. Резидентные копии таких вирусов остаются жизнеспособными вплоть до очередной перезагрузки, даже если на диске уничтожены все зараженные файлы.

Полиморфик-вирусами являются те, обнаружение которых невозможно (или крайне затруднительно) осуществить при помощи так называемых вирусных масок – участков постоянного кода, специфичных для конкретного вируса. Достигается это двумя основными способами – шифрованием основного кода вируса с непостоянным ключом и случайным набором команд расшифровщика или изменением самого выполняемого кода вируса.

Уровни полиморфизма. Существует деление полиморфик-вирусов на уровни в зависимости от сложности кода, который встречается в расшифровщиках этих вирусов. Такое деление впервые предложил доктор Алан Соломон, через некоторое время Весселин Бончев расширил его.

Уровень 1. Вирусы, которые имеют некоторый набор расшифровщиков с постоянным кодом и при заражении выбирают один из них. Такие вирусы являются полуполиморфиками и носят также название олигоморфик (oligomorphic).

Уровень 2. Расшифровщик вируса содержит одну или несколько постоянных инструкций, основная же его часть непостоянна.

Уровень 3. Расшифровщик содержит неиспользуемые инструкции мусор типа NOP, CLI, STI и т.д.

Уровень 4. В расшифровщике используются взаимозаменяемые инструкции и изменение порядка следования (перемешивание) инструкций. Алгоритм расшифровки при этом не изменяется.

Уровень 5. Используются все перечисленные выше приемы, алгоритм расшифровки непостоянен, возможно повторное шифрование кода вируса и даже частичное шифрование самого кода расшифровщика.

Уровень 6. Permutating-вирусы. Изменению подлежит основной код вируса – делится на блоки, которые при заражении переставляются в произвольном порядке. Вирус при этом остается работоспособным. Подобные вирусы могут быть не зашифрованы.

Приведенное деление не свободно от недостатков, поскольку производится по единственному критерию – возможности обнаруживать вирус по коду расшифровщика при помощи стандартного приема вирусных масок.

Если произвести деление на уровни с точки зрения антивирусов, использующих системы автоматического расшифрования кода вируса (эмуляторы), то деление на уровни будет зависеть от сложности эмуляции кода вируса. Возможно, более объективным является деление, в котором помимо критерия вирусных масок участвуют и другие параметры:

1. Степень сложности полиморфик-кода (процент от всех инструкций процессора, которые могут встретиться в коде расшифровщика).
2. Использование антиэмуляторных приемов.
3. Постоянство алгоритма расшифровщика.
4. Постоянство длины расшифровщика.

10.2. ПРОФИЛАКТИКА И ЛЕЧЕНИЕ ИНФОРМАЦИОННЫХ ИНФЕКЦИЙ. ПРОГРАММЫ ОБНАРУЖЕНИЯ И ЗАЩИТЫ ОТ ВИРУСОВ

Для обнаружения, удаления и защиты от компьютерных вирусов разработано несколько видов специальных программ, которые позволяют обнаруживать и уничтожать вирусы. Такие программы называются антивирусными.

Различают следующие виды антивирусных программ (рис. 10.1):

Рис. 10.1. Виды антивирусных программ

- программы-детекторы;
- программы-доктора или фаги;
- программы-мониторы (ревизоры);
- программы фильтры;
- программы-вакцины или иммунизаторы.

Программы-детекторы (*сканеры*) осуществляют поиск характерной для конкретного вируса последовательности байтов (сигнатуры вируса) в оперативной памяти и в файлах и при обнаружении выдают соответствующее сообщение. Недостатком таких антивирусных программ является то, что они могут находить только те вирусы, которые известны разработчикам.

Во многих сканерах используются также алгоритмы эвристического сканирования, т.е. анализ последовательности команд в проверяемом объекте, набор некоторой статистики и принятие решения («возможно, заражен» или «не заражен») для каждого проверяемого объекта.

К достоинствам сканеров относится их универсальность, к недостаткам – размеры антивирусных баз, которые сканерам приходится «таскать за собой», и относительно небольшая скорость поиска вирусов.

Программы-доктора или фаги, а также программы-вакцины не только находят зараженные вирусами файлы, но и «лечат» их, т.е. удаляют из файла тело программы вируса, возвращая файлы в исходное состояние. В начале своей работы фаги ищут вирусы в оперативной памяти, уничтожая их, и только затем переходят к «лечению» файлов. Среди фагов выделяют полифаги, т.е. программы-доктора, предназначенные для поиска и уничтожения большого количества вирусов. Наиболее известными полифагами являются программы Aidstest, Scan, Norton AntiViris и Doctor Web.

Программы-ревизоры (*CRC-сканеры*) относятся к самым надежным средствам защиты от вирусов. Принцип работы CRC-сканеров основан на подсчете CRC-сумм (контрольных сумм) для присутствующих на диске файлов/системных секторов. Эти CRC-суммы затем сохраняются в базе данных антивируса, как, впрочем, и некоторая другая информация: длины файлов, даты их последней модификации и т.д. При последующем запуске CRC-сканеры сверяют данные, содержащиеся в базе данных, с реально подсчитанными значениями. Если информация о файле, записанная в базе данных, не совпадает с реальными значениями, то CRC-сканеры сигнализируют о том, что файл был изменен или заражен вирусом.

CRC-сканеры, использующие «антистелс»-алгоритмы, являются довольно сильным оружием против вирусов: практически 100 % вирусов оказываются обнаруженными почти сразу после их появления на компьютере. Однако у этого типа антивирусов есть врожденный недостаток, который заметно снижает их эффективность. Этот недостаток состоит в том, что CRC-сканеры не способны поймать вирус в момент его появления в системе, а делают это лишь через некоторое время, уже после того, как вирус разошелся по компьютеру. CRC-сканеры не могут детектировать вирус в новых файлах, поскольку в их базах данных отсутствует информация об этих файлах. Более того, периодически появляются вирусы, которые используют эту «слабость» CRC-сканеров, заражают только вновь создаваемые файлы и остаются невидимыми для CRC-сканеров. К числу программ-ревизоров относится, например, известная в России программа ADinf фирмы «Диалог-наука».

Антивирусные мониторы – это резидентные программы, перехватывающие вирусоопасные ситуации и сообщающие об этом пользователю. К вирусоопасным относятся вызовы на открытие для записи в выполняемые файлы запись в загрузочные секторы дисков или MBR винчестера, попытки программ остаться резидентно и т.д., т.е. вызовы, которые характерны для вирусов в моменты их размножения.

К достоинствам мониторов относится их способность обнаруживать и блокировать вирус на самой ранней стадии его размножения, что, кстати, бывает очень полезно в случаях, когда давно известный вирус постоянно «выползает неизвестно откуда». К недостаткам относятся существование путей обхода защиты монитора и большое количество ложных срабатываний.

Вакцины или *иммунизаторы* – это резидентные программы, предоставляющие заражение файлов. Вакцины применяют, если отсутствуют программы-доктора, «лечащие» этот вирус. Иммунизаторы делятся на два типа: иммунизаторы, сообщающие о заражении, и иммунизаторы, блокирующие заражение каким-либо типом вируса. Первые обычно записываются в конец файлов (по принципу файлового вируса и при запуске файла каждый раз проверяют его на изменение. Недостаток у таких иммунизаторов всего один, но он летален: абсолютная неспособность сообщить о заражении «стелс»-вирусом. По-

этому такие иммунизаторы, как и мониторы, практически не используются в настоящее время.

Второй тип иммунизации защищает систему от поражения вирусом какого-то определенного вида. Файлы на дисках модифицируются таким образом, что вирус принимает их за уже зараженные. Для защиты от резидентного вируса в память компьютера заносится программа, имитирующая копию вируса, при запуске вирус натыкается на нее и считает, что система уже заражена.

Качество антивирусной программы определяется по следующим позициям, приведенным в порядке убывания их важности:

1. Надежность и удобство работы – отсутствие зависаний антивируса и прочих технических проблем, требующих от пользователя специальной подготовки.

2. Качество обнаружения вирусов всех распространенных типов, сканирование внутри файлов документов/таблиц (MS Word, Excel, Office), упакованных и архивированных файлов. Отсутствие «ложных срабатываний». Возможность лечения зараженных объектов (для сканеров – периодичность появления новых версий, т.е. скорость настройки сканера на новые вирусы).

3. Существование версий антивируса под все популярные платформы (DOS, Windows 95/98/NT/Me/2000/XP, Novell NetWare, OS/2, Alpha, UNIX, Linux и т. д.), присутствие не только режима «сканирование по запросу», но и «налету».

Контрольные вопросы

1. Дайте классификацию информационным инфекциям.
2. Укажите мероприятия по защите информации от вирусов.
3. Чем определяется качество антивирусной программы?

Заключение

Цель мероприятий в области информационной безопасности – защитить интересы субъектов информационных отношений. Интересы эти многообразны, но все они концентрируются вокруг трех основных аспектов: доступность, целостность, конфиденциальность.

Первый шаг при построении системы ИБ организации – ранжирование и детализация этих аспектов.

Важность проблематики ИБ объясняется двумя основными причинами:

- ценностью накопленных информационных ресурсов;
- критической зависимостью от информационных технологий.

Разрушение важной информации, кража конфиденциальных данных, перерыв в работе вследствие отказа – все это выливается в крупные материальные потери, наносит ущерб репутации организации.

Современные информационные системы сложны и, значит, опасны уже сами по себе, даже без учета активности злоумышленников. Постоянно обнаруживаются новые уязвимые места в программном обеспечении. Приходится принимать во внимание чрезвычайно широкий спектр аппаратного и программного обеспечения, многочисленные связи между компонентами.

Успех в области информационной безопасности может принести только комплексный подход, сочетающий меры четырех уровней: законодательного, административного, процедурного, программно-технического.

Законодательный уровень является важнейшим для обеспечения информационной безопасности. Необходимо всячески подчеркивать важность проблемы ИБ; сконцентрировать ресурсы на важнейших направлениях исследований; скоординировать образовательную деятельность; создать и поддерживать негативное отношение к нарушителям ИБ. На законодательном уровне особое внимание заслуживают правовые акты и стандарты.

Российские правовые акты в большинстве своем имеют ограничительную направленность. К тому же в законах не предусмотрена ответственность государственных органов за нарушения ИБ. Реальность такова, что в России в деле обеспечения ИБ на помощь государства рассчитывать не приходится. На этом фоне поучительным является законодательство США в области ИБ, которое гораздо обширнее и многограннее российского.

Среди стандартов выделяются «Оранжевая книга», рекомендации X.800 и «Критерии оценки безопасности информационных технологий».

Международный стандарт ISO 15408, известный как «Общие критерии», реализует современный подход, в нем зафиксирован чрезвычайно широкий спектр сервисов безопасности. Его принятие в качестве национального стандарта важно не только из абстрактных соображений интеграции в мировое сообщество; но и существенно расширит спектр доступных сертифицированных решений.

Главная задача 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.

Разработка политики и программы безопасности начинается с анализа рисков, первым этапом которого, в свою очередь, является ознакомление с наиболее распространенными угрозами.

Главные угрозы – внутренняя сложность ИС, непреднамеренные ошибки штатных пользователей, операторов, системных администраторов и других лиц, обслуживающих информационные системы. Реальную опасность представляют пожары и другие аварии поддерживающей инфраструктуры.

Для подавляющего большинства организаций достаточно общего знакомства с рисками; ориентация на типовые, апробированные решения позволит обеспечить базовый уровень безопасности при минимальных интеллектуальных и разумных материальных затратах.

Безопасность невозможно добавить к системе, ее нужно закладывать с самого начала и поддерживать до конца.

Меры процедурного уровня ориентированы на людей (а не на технические средства) и подразделяются на следующие виды: управление персоналом, физическая защита, поддержание работоспособности, реагирование на нарушения режима безопасности, планирование восстановительных работ.

На этом уровне применимы важные принципы безопасности: непрерывность защиты в пространстве и времени, разделение обязанностей, минимизация привилегий.

Информационная безопасность во многом зависит от аккуратного ведения текущей работы, которая включает: поддержку пользователей, поддержку программного обеспечения, конфигурационное управление, резервное копирование, управление носителями, документирование, регламентные работы.

В то же время необходимо готовиться к событиям неординарным, т.е. к нарушениям ИБ. Заранее продуманная реакция на нарушения режима безопасности преследует три главные цели: локализация инцидента и уменьшение наносимого вреда, выявление нарушителя, предупреждение повторных нарушений.

Программно-технические меры, т.е. меры, направленные на контроль компьютерных сущностей – оборудования, программ и/или данных, образуют последний и самый важный рубеж информационной безопасности.

На этом рубеже становятся очевидными не только позитивные, но и негативные последствия быстрого прогресса информационных технологий. Во-первых, дополнительные возможности появляются не только у специалистов по ИБ, но и у злоумышленников. Во-вторых, информационные системы все время модернизируются, перестраиваются, к ним добавляются недостаточно проверенные компоненты, что затрудняет соблюдение режима безопасности.

Меры безопасности целесообразно разделить на следующие виды: превентивные, препятствующие нарушениям ИБ; меры обнаружения нарушений; локализующие, сужающие зону воздействия нарушений; меры по выявлению нарушителя; меры восстановления режима безопасности.

С практической точки зрения важными также 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.

Центральным для программно-технического уровня является понятие сервиса безопасности. В число таких сервисов входят: идентификация и аутентификация, управление доступом, протоколирование и аудит, шифрование, контроль целостности, экранирование, анализ защищенности, обеспечение отказоустойчивости, обеспечение безопасного восстановления, управление.

Эти сервисы должны функционировать в открытой сетевой среде с разнородными компонентами, т.е. быть устойчивыми к соответствующим угрозам, а их применение должно быть удобным для пользователей и администраторов.

Миссия обеспечения информационной безопасности трудна, во многих случаях невыполнима, но всегда благородна.

Список Литературы

1. Завгородний, В.И. Комплексная защита информации в компьютерных системах : учебное пособие для вузов / В.И. Завгородний. – М. : Логос, 2001. – 264 с.
2. Зегжда, Д.П. Основы безопасности информационных систем / Д.П. Зегжда, А.М. Ивашко. – М. : Горячая Линия – Телеком, 2000. – 452 с.
3. Малюк, А.А. Введение в защиту информации в автоматизированных системах / А.А. Малюк, С.В. Пазизин, Н.С. Погожин. – М. : Горячая Линия – Телеком, 2001. – 148 с.
4. Теоретические основы компьютерной безопасности : учебное пособие для вузов / П.Н. Девянин, О.О. Михальский, Д.И. Правиков, А.Ю. Щербаков. – М. : Радио и связь, 2000. – 192 с.
5. UNIX: Руководство системного администратора / Эви Немет, Гарт Снайдер, Скотт Сибасс, Трент Р. Хейн ; пер. с англ. С.М. Тимачева ; под ред. М.В. Коломыцева. – 3-е изд. – Киев : BHV, 1998. – 832 с.
6. Грушо, А.А. Теоретические основы защиты информации / А.А. Грушо, Е.Е. Тимонина. – М. : Яхтсмен, 1996. – 192 с.
7. Жельников, В.Г. Криптография от папируса до компьютера / В.Г. Жельников. – М. : Dore Print, 1999. – 214 с.
8. Баричев, С.Г. Основы современной криптографии / С.Г. Баричев, В.В. Гончаров, Р.Е. Серов. – М. : Горячая линия – Телеком, 2001. – 121 с.
9. ГОСТ 28147–89 «Системы обработки информации. Защита криптографическая. Алгоритм криптографического преобразования информации».
10. ГОСТ Р 51583 «Защита информации. Порядок создания автоматизированных систем в защищенном исполнении».
11. Руководящий документ. Концепция защиты средств вычислительной техники и автоматизированных систем от несанкционированного доступа к информации // Сборник руководящих документов по защите информации от несанкционированного доступа. – М. : ГТК РФ, 1998. – С. 15–22.
12. Руководящий документ. Средства вычислительной техники. Защита от несанкционированного доступа к информации. Показатели защищенности от несанкционированного доступа к информации // Сборник руководящих документов по защите информации от несанкционированного доступа. – М. : ГТК РФ, 1998. – С. 53–72.
13. Руководящий документ. Защита от несанкционированного доступа к информации. Термины и определения // Сборник руководящих документов по защите информации от несанкционированного доступа. – М. : ГТК РФ, 1998. – С. 5–13.
14. Руководящий документ. Временное положение по организации разработки, изготовления и эксплуатации программных и технических средств защиты информации от несанкционированного доступа в автоматизированных системах и средствах вычислительной техники // Сборник руководящих документов по защите информации от несанкционированного доступа. – М. : ГТК РФ, 1998. – С. 73–92.
15. Руководящий документ. Положение по аттестации объектов информатизации по требованиям безопасности информации. – М. : ГТК РФ, 1994.
16. Руководящий документ. Автоматизированные системы. Защита от несанкционированного доступа к информации. Классификация автоматизированных систем и требования к защите информации // Сборник руководящих документов по защите информации от несанкционированного доступа. – М. : ГТК РФ, 1998. – С. 23–52.
17. Руководящий документ. Средства вычислительной техники. Межсетевые экраны. Защита от несанкционированного доступа к информации. Показатели защищенности от несанкционированного к информации // Сборник руководящих документов по защите информации от несанкционированного доступа. – М. : ГТК РФ, 1998. – С. 93–106.
18. Руководящий документ. Защита от несанкционированного доступа к информации. Ч. 1. Программное обеспечение СЗИ. Классификация по уровню контроля отсутствия недекларированных возможностей. – М. : ГТК РФ, 1999.
19. Руководящий документ. Специальные требования и рекомендации по технической защите конфиденциальной информации. – М. : ГТК РФ, 2001.
20. Герасименко, В.А. Защита информации в автоматизированных системах обработки данных / В.А. Герасименко. – М. : Энергоатомиздат, 1994. – Кн. 1, 2. – 278 с., 217 с.
21. Дайсон, П. Операционная система UNIX: Настольный справочник / пер. С. Орлов; под ред. В. Вагина. – М. : ЛОРИ, 1997. – 395 с.
22. Дейтел, Г. Введение в операционные системы: в 2 т. / пер с англ. – М. : Мир, 1987. – Т. 2. – 359 с.

23. Стандарты и рекомендации в области информационной безопасности // JetInfo. – 1996. – № 1–3. – 32 с.

24. Физическая защита информационных систем // JetInfo. – 1997. – № 1. – 28 с.

25. Доступность как элемент информационной безопасности // JetInfo. – 1997. – № 1. – 28 с.

26. Программно-технологическая безопасность информационных систем // JetInfo. – 1997. – № 6–7. – 28 с.

27. Общие критерии оценки безопасности информационных технологий и перспективы их использования // JetInfo.– 1998. – № 1. – С. 12–17.

28. Концептуальные вопросы оценки безопасности информационных технологий // JetInfo. – 1998. – № 5–6. – С. 13–21.

29. Анализ рисков и управление рисками // JetInfo. – 1999. – № 1. – 28 с.

30. Современная трактовка сервисов безопасности // JetInfo. – 1999. – № 5. – С. 14–24.

31. Аудит безопасности информационных систем // JetInfo. – 1999. – № 9. – 24 с.

32. Гостехкомиссия России – точка зрения на техническую защиту информации // JetInfo. – 1999. – № 11. – С. 2–12.

33. Информационная безопасность. Ситуация в мире и России // JetInfo. – 2000. – № 8. – 16 с.

34. Общее описание процедуры аттестации автоматизированных систем по требованиям информационной безопасности // JetInfo. – 2000. – № 11. – 20 с.

35. Обзор криптотехнологий // JetInfo. – 2001. – № 3. – 16 с.

36. Новые приоритеты в информационной безопасности США // JetInfo. – 2001. – № 10. – 12 с.

37. Актуальные вопросы выявления сетевых атак // JetInfo. – 2002. – № 3. – 28 с.

38. Компания «Инфосистемы Джет» // JetInfo. – 2002. – № 5. – 16 с.

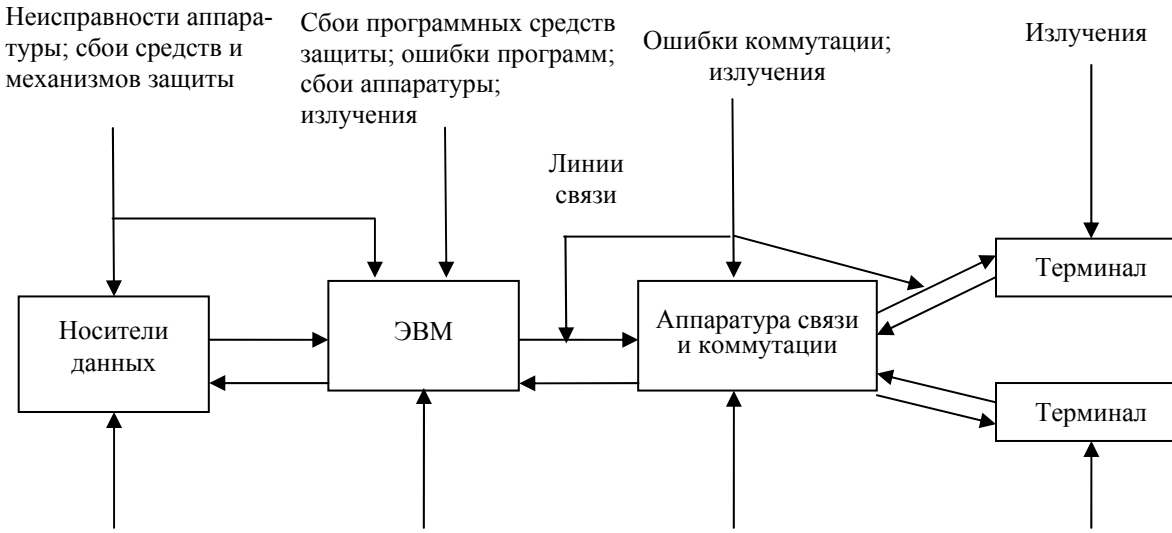
39. Анатомия информационной безопасности США // JetInfo. – 2002. – № 6. – 40 с.

40. Анализ защищенности корпоративных автоматизированных систем // JetInfo.– 2002. – № 7. – 28 с.

41. Технологии и инструментарий для управления рисками // JetInfo. – 2003. – № 2. – 32 с.

42. Профили защиты на основе «Общих критериев». Аналитический обзор // JetInfo.– 2003. – № 3. – 32 с.

43. Информация журналов JetInfo в электронном виде (формат pdf) на сайте www. jetinfo.ru.


Неисправности аппаратуры; сбои средств и механизмов защиты
Сбои программных средств защиты; ошибки программ; сбои аппаратуры; излучения
Ошибки коммутации; излучения
Излучения
Линии связи
Носители данных
ЭВМ
Аппаратура связи и коммутации
Терминал
Терминал
Хищение; несанкционированное копирование и считывание носителей данных
Ошибки операторов; раскрытие механизмов защиты; отключение механизмов защиты; несанкционированный доступ к данным
Незаконное подключение; раскрытие содержания данных; анализ, изменение прерывание потока данных; инициация ложного соединения
Маскировка под другого пользователя; ошибки в работе; раскрытие механизмов защиты; незаконное использование данных; копирование данны

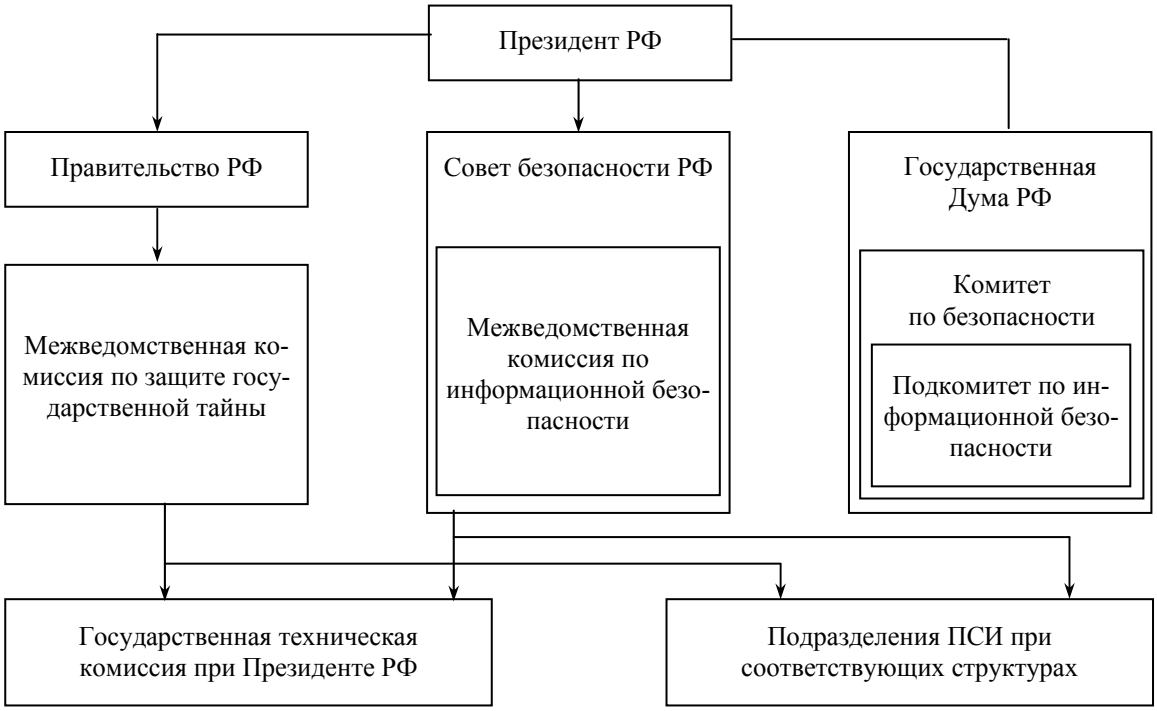


Рис. 3.1. Структура государственных органов обеспечения безопасности

Рис. 2.1. Классификация угроз по степени преднамеренности проявления

Рис. 2.2. Каналы утечки технических средств информационных систем

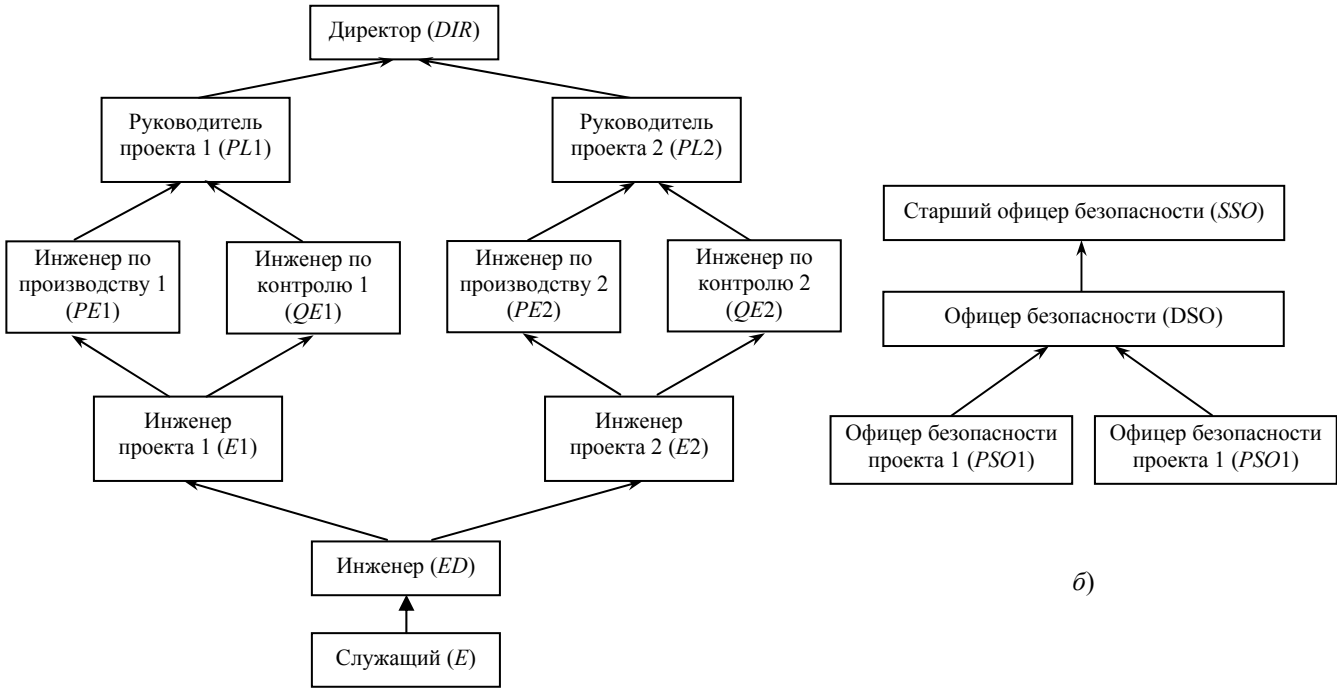


Рис. 5.9. Пример иерархии ролей (*а*) и иерархии административных ролей (*б*)

Рис. 8.2. Взаимодействие участников процесса создания и распространения ПО