ГОУВПО «Воронежский государственный
технический университет»

Т.И. Сергеева М.Ю. Сергеев

МЕТОДЫ И СРЕДСТВА ЗАЩИТЫ КОМПЬЮТЕРНОЙ ИНФОРМАЦИИ

Утверждено Редакционно-издательским советом
университета в качестве учебного пособия

Воронеж 2011

УДК 681.3
Сергеева Т.И. Методы и средства защиты компьютерной информации: учеб. пособие / Т.И. Сергеева, М.Ю. Сергеев. Воронеж: ГОУВПО «Воронежский государственный технический университет», 2011. 230 с.

Рассматриваются общая организация систем защиты компьютерной информации, методы и средства защиты информации, алгоритмы шифрования данных, способы аутентификации пользователей, защита информации средствами операционной системы и в сети.

Издание соответствует требованиям Государственного образовательного стандарта высшего профессионального образования по направлению 230100 «Информатика и вычислительная техника», специальности 230101 «Вычислительные машины, комплексы, системы и сети», дисциплине «Методы и средства защиты компьютерной информации».

Учебное пособие предназначено для студентов заочной и заочной сокращенной форм обучения.

Учебное пособие подготовлено в электронном виде в текстовом редакторе Microsoft Word 2003 и содержится в файле MCZKI_ZO.doc.

Табл. 17. Ил. 39. Библиогр.: 19 назв.

Научный редактор д-р техн. наук, проф. С.Л. Подвальный

Рецензенты: кафедра информационных и управляющих систем Воронежской государственной технологической академии (д-р техн. наук, проф. В.Ф. Лебедев);
д-р техн. наук, проф. В.Л. Бурковский

© Сергеева Т.И., Сергеев М.Ю., 2011
© Оформление. ГОУВПО «Воронежский государственный технический университет», 2011

ВВЕДЕНИЕ

Защита компьютерной информации является важной и актуальной задачей, обеспечивающей эффективную работу любой организации, предприятия, фирмы.

Актуальность и важность проблемы защиты информации обуславливают следующие причины:

резкое увеличение вычислительной мощности современных компьютеров при одновременном упрощении их эксплуатации;

резкое увеличение объемов информации, накапливаемой, хранимой и обрабатываемой с помощью компьютеров и других средств автоматизации;

сосредоточение в единых базах данных информации различного назначения и различной принадлежности;

высокие темпы роста парка персональных компьютеров, находящихся в эксплуатации в самых разных сферах деятельности;

резкое расширение круга пользователей, имеющих непосредственный доступ к вычислительным ресурсам и массивам данных;

бурное развитие программных средств, не удовлетворяющих даже минимальным требованиям безопасности;

повсеместное распространение сетевых технологий и объединение локальных сетей в глобальные;

развитие глобальной сети Интернет, практически не препятствующей нарушениям безопасности систем обработки информации во всем мире.

Целью данного пособия является изложение общих вопросов организации защиты компьютерной информации, рассмотрение основных методов и средств защиты информации в компьютерных системах.

В первой главе пособия рассмотрены основы защиты информации, приведены основные угрозы компьютерной информации, модели потенциальных нарушителей.

Во второй главе приведена классификация методов и средств защиты информации, стандарты информационной безопасности.

Третья и четвертая главы посвящены криптографическим методам и средствам защиты данных. Рассматриваются симметричные и асимметричные криптосистемы. В качестве примеров приведены алгоритм шифрования DES, отечественный стандарт шифрования, алгоритм шифрования RSA.

Пятая глава содержит описание моделей и алгоритмов защиты информации от несанкционированного доступа в операционных системах.

В шестой главе рассматриваются алгоритмы аутентификации пользователей.

Седьмая глава содержит обзор методов и средств защиты информации в сетях.

В восьмой главе рассматриваются вопросы защиты компьютерных систем от вредоносных программ, несанкционированного использования и копирования.

Пособие соответствует типовой программе по дисциплине «Методы и средства защиты компьютерной информации» и предназначено для студентов заочной полной и заочной сокращенной форм обучения.

1. ВВЕДЕНИЕ В ОСНОВЫ ЗАЩИТЫ ИНФОРМАЦИИ

1.1. Основные направления защиты информации

Быстро развивающиеся компьютерные информационные технологии привели к тому, что понятие «информация» используется для обозначения специального товара, который можно приобрести, продать, обменять и т.д. При этом стоимость информации часто превосходит в сотни и тысячи раз стоимость компьютерной системы, в которой она находится. Поэтому вполне естественно возникает необходимость в защите информации от несанкционированного доступа, умышлен-

ного изменения, кражи, уничтожения и других преступных действий.

Проблемы защиты информации привлекают все большее внимание как специалистов в области компьютерных систем и сетей, так и многочисленных пользователей современных компьютерных систем.

Одним из направлений защиты информации является защита информации путем ее шифрования с помощью методов и средств криптографического преобразования данных.

Для обозначения всей области тайной (секретной) связи используется термин «криптология», который происходит от греческих корней «cryptos" – тайный и "logos" – сообщение. Криптология довольно четко может быть разделена на два направления: криптографию и криптоанализ.

Задача криптографа – обеспечить конфиденциальность (секретность) и аутентичность (подлинность) передаваемых сообщений.

Задача криптоаналитика - «взломать» систему защиты, разработанную криптографами. Он пытается раскрыть зашифрованный текст или выдать поддельное сообщение за настоящее.

Появление новых информационных технологий и интенсивное развитие компьютерных сетей привлекают все большее внимание пользователей к глобальной сети Интернет. Подключение к Интернет дает большие преимущества в работе, однако при этом возникают серьезные проблемы с обеспечением информационной безопасности локальной или корпоративной сети.

В силу открытости своей идеологии Интернет предоставляет злоумышленникам много возможностей для вторжения во внутренние сети предприятий и организаций с целью хищения, искажения или разрушения важной и конфиденциальной информации. Решение задач по защите внутренних сетей от наиболее вероятных атак через Интернет может быть возложено на межсетевые экраны или брандмауэры или firewall.

Применяются и программные методы защиты, к которым относятся защищенные криптопротоколы SSL и SKIP.

Важным приложением, нуждающимся в эффективных средствах защиты, являются электронные платежные системы. В этих системах в качестве универсального платежного средства используются банковские пластиковые карты. Для обеспечения надежной работы электронная платежная система должна быть надежно защищена. С точки зрения информационной безопасности в системах электронных платежей существует ряд потенциально уязвимых мест:

пересылка платежных и других сообщений между банками;

между банком и банкоматом;

между банком и клиентами.

Для обеспечения защиты информации на отдельных узлах системы электронных платежей должны быть реализованы следующие механизмы защиты:

управление доступом на оконечных системах;

обеспечение целостности и конфиденциальности сообщений;

взаимная аутентификация абонентов;

гарантии доставки сообщения и т.д.

Качество решения указанных проблем существенно зависит от рационального выбора криптографических средств при реализации механизмов защиты.

1.2. Информация как предмет защиты

Информация применительно к задаче ее защиты - это сведения о лицах, предметах, фактах, событиях, явлениях или процессах независимо от формы их представления. В зависимости от формы представления информация может быть разделена на речевую, телекоммуникационную и документированную.

Информацию разделяют на открытую и ограниченного доступа. К информации **ограниченного доступа** относятся государственная тайна и конфиденциальная информация.

В соответствии с российским законодательством к **конфиденциальной информации** относится следующая информация:

служебная тайна (врачебная, адвокатская, тайна суда и следствия и т.д.);

коммерческая тайна;

персональные данные (сведения о фактах, событиях и обстоятельствах жизни гражданина, позволяющие идентифицировать его личность).

Информация является одним из объектов гражданских прав, в том числе и прав собственности, владения и пользования. **Собственник** информационных ресурсов, систем и технологий – это субъект с полномочиями владения, пользования и распоряжения указанными объектами. **Владельцем** информационных ресурсов, систем и технологий является субъект с полномочиями владения и пользования указанными объектами. Под **пользователем** информации будем понимать субъекта, обращающегося к информационной системе за получением необходимой ему информации и пользующегося ею.

Защищаемой информацией называют информацию, являющуюся предметом собственности и подлежащую защите в соответствии с требованиями правовых документов или требованиями, установленными собственником информации.

Однако защите подлежит не всякая информация, а только та, которая имеет цену. Ценной становится та информация, обладание которой позволит ее существующему и потенциальному владельцу получить какой-либо выигрыш: моральный, материальный, политический и т.д.

Ценность информации является критерием при принятии любого решения о ее защите и для выбора метода защиты. В

денежном выражении затраты на защиту информации не должны превышать возможные потери.

Принято следующее разделение информации по уровню важности:

1) *жизненно важная незаменимая информация*, наличие которой необходимо для функционирования организации;

2) *важная информация* - информация, которая может быть заменена или восстановлена, но процесс восстановления очень труден и связан с большими затратами;

3) *полезная информация* - информация, которую трудно восстановить, однако организация может эффективно функционировать и без нее;

4) *несущественная информация* - информация, которая больше не нужна организации.

Категория важности, как и ценность информации, обычно изменяется со временем и зависит от степени отношения к ней различных групп потребителей и потенциальных нарушителей.

Приведенное деление информации по уровню важности согласуется с принципом деления информации по уровням секретности.

Уровень секретности - это административная или законодательная мера, соответствующая мере ответственности лица за утечку или потерю конкретной секретной информации, регламентируемой специальным документом, с учетом государственных, военно-стратегических, коммерческих, служебных или частных интересов. Такой информацией может быть государственная, военная, коммерческая, служебная или личная тайна.

Защитой информации называют деятельность по предотвращению утечки защищаемой информации, несанкционированных и непреднамеренных воздействий на защищаемую информацию.

Под **утечкой** понимают неконтролируемое распространение защищаемой информации путем ее разглашения и не-

санкционированного доступа к ней. **Разглашение** – это доведение защищаемой информации до неконтролируемого количества получателей информации (например, публикация информации на открытом сайте в сети Интернет или в открытой печати).

Несанкционированный доступ – получение защищаемой информации заинтересованным субъектом с нарушением правил доступа к ней.

Несанкционированное воздействие на защищаемую информацию – воздействие с нарушением правил ее изменения (например, намеренное внедрение в защищаемые информационные ресурсы вредоносного программного кода или умышленная подмена электронного документа).

Непреднамеренное воздействие на защищаемую информацию – это воздействие на неё из-за ошибок пользователя, сбоя технических и программных средств, природных явлений иных нецеленаправленных воздействий (например, уничтожение документов в результате отказа накопителя на жестком магнитном диске компьютера).

Цель защиты информации – предотвращение ущерба собственнику, владельцу или пользователю информации.

Под **эффективностью защиты информации** понимают степень соответствия результатов защиты информации поставленной цели.

Объектом защиты может быть информация, ее носитель или информационный процесс, в отношении которых необходимо обеспечивать защиту в соответствии с поставленной целью.

Безопасность информации – состояние информации, технических средств и технологии ее обработки, характеризующееся свойствами конфиденциальности, целостности и доступности информации при ее обработке техническими средствами. Таким образом, основными характеристиками защищаемой информации являются целостность, конфиденциальность и доступность.

Целостность информации – свойство информации, технических средств и технологии ее обработки, характеризующееся способностью противостоять несанкционированному или непреднамеренному уничтожению и искажению информации. Целостность является частью более широкой характеристики информации – ее достоверности, включающей помимо целостности еще полноту и точность отображения предметной области.

Конфиденциальность информации – свойство информации, технических средств и технологии ее обработки, характеризующееся способностью информации сохраняться в тайне от субъектов, у которых нет полномочий на право ознакомления с ней.

Конфиденциальность является субъективной характеристикой информации, связанной с объективной необходимостью защиты законных интересов одних субъектов от других.

Доступность информации – свойство информации, технических средств и технологии ее обработки, характеризующееся способностью обеспечивать беспрепятственный доступ к информации субъектов, имеющих на это надлежащие полномочия. Отказом в обслуживании называют состояние информационной системы, при котором блокируется доступ к некоторому ее ресурсу.

Нарушение безопасности информации – утрата свойств, характеризующих безопасность информации при ее обработке техническими средствами.

Угроза безопасности информации – случайная или преднамеренная деятельность людей или физической явление, которые могут привести к нарушению безопасности информации.

Источник угрозы безопасности информации – любое физическое лицо, материальный объект или явление, создающие угрозу безопасности информации при ее обработке техническими средствами.

Утечка информации – утрата свойств конфиденциальности информации.

Несанкционированный доступ к информации (НСД) – доступ к информации при ее обработке техническими средствами без разрешения, используя возможности этих технических средств.

Искажение информации – любое преднамеренное или случайное изменение информации при ее обработке техническими средствами, изменяющее содержание этой информации.

Уничтожение информации – действие, в результате которого информация перестает физически существовать в технических средствах ее обработки.

Аппаратная закладка – электронное устройство, встраиваемое или подключаемое к элементам технических средств обработки информации в целях нарушения безопасности информации при ее обработке техническими средствами.

Политика безопасности – это набор документированных норм, правил и практических приемов, регулирующих управление, защиту и распределение информации ограниченного доступа.

1.3. Основные угрозы компьютерной безопасности

Под угрозой безопасности информации в компьютерной системе (КС) понимают событие или действие, которое может вызвать изменение функционирования КС, связанное с нарушением защищенности обрабатываемой в ней информации.

Выделяют следующие **основные угрозы информации** по предмету направленности:

угрозы нарушения конфиденциальности обрабатываемой информации - информация становится известной лицу, которое не должно иметь к ней доступ; разглашается конфиденциальная или секретная информация;

угрозы нарушения целостности обрабатываемой информации – изменение или искажение информации, приво-

дящее к нарушению ее качества или полному уничтожению; целостность информации может быть нарушена как злоумышленником, так и в результате объективных воздействий со стороны среды, окружающей систему; эта угроза особенно актуальна для компьютерных сетей и систем телекоммуникаций;

угрозы нарушения работоспособности системы (отказ в обслуживании) направлены на создание ситуаций, когда в результате преднамеренных действий снижается работоспособность вычислительной системы, либо ее ресурсы становятся недоступными.

По **источнику возникновения угрозы безопасности данных** разделяются на **воздействия** и **каналы утечки данных** (рис. 1).

Нарушение безопасности данных возможно как вследствие различных возмущающих **воздействий**, в результате которых происходит уничтожение (модификация) данных или создаются каналы утечки данных, так и вследствие использования нарушителем **каналов утечки** данных.

Воздействия, в результате которых может быть нарушена безопасность данных, включают в себя:

случайные воздействия природной среды (ураган, землетрясение, пожар, наводнение и т.п.);

целенаправленные воздействия нарушителя (шпионаж, разрушение компонентов информационно-вычислительной системы (ИВС), использование прямых каналов утечки данных);

внутренние возмущающие факторы (отказы аппаратуры, ошибки в математическом и программном обеспечении, недостаточная профессиональная и морально-психологическая подготовка персонала и т.д.).

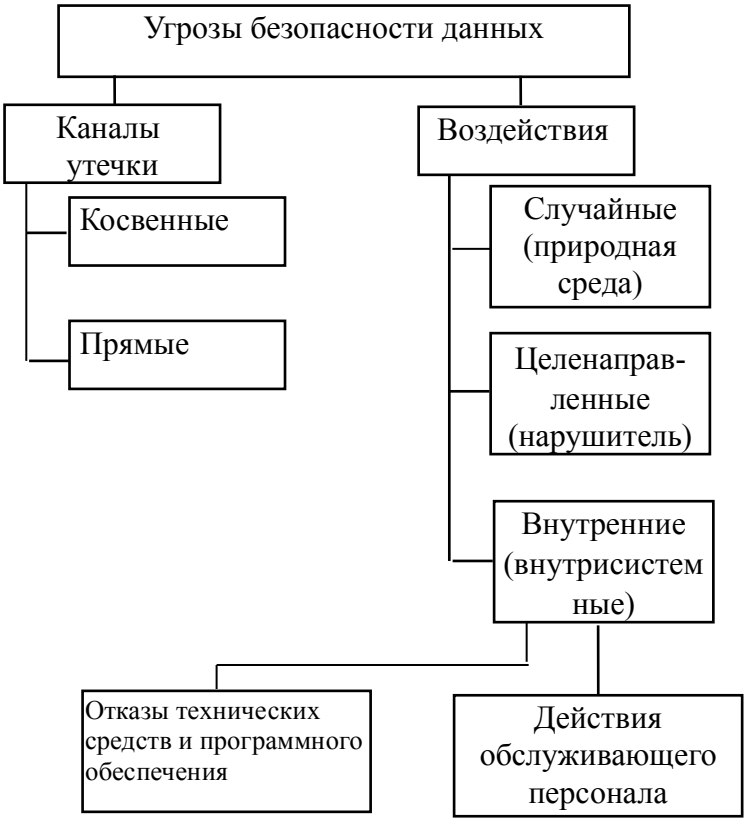


Рис. 1. Виды угроз безопасности данных

Под **каналом утечки** понимается потенциальная возможность несанкционированного доступа, которая обусловлена архитектурой, технологической схемой функционирования ИВС, а также существующей организацией работы с данными.

Все **каналы утечки данных** можно разделить на **косвенные** и **прямые**.

Косвенными называются такие каналы утечки, использование которых для несанкционированного доступа не требует непосредственного доступа к техническим

устройствам ИВС. Косвенные каналы утечки возникают, например, вследствие недостаточной изоляции помещений, просчетов в организации работы с данными и предоставляют нарушителю возможность применения подслушивающих устройств, дистанционного фотографирования, перехвата электромагнитных излучений, хищения носителей данных и производственных отходов (листингов машинных программ и др.).

Прямые каналы утечки данных требуют непосредственного доступа к техническим средствам ИВС и данным. Наличие прямых каналов утечки обусловлено недостатками технических и программных средств защиты, ОС, СУБД, математического и программного обеспечения, а также просчетами в организации технологического процесса работы с данными. Прямые каналы утечки данных позволяют нарушителю подключиться к аппаратуре ИВС, получить доступ к данным и выполнить действия по анализу, модификации и уничтожению данных.

При использовании прямых каналов утечки нарушитель может осуществить следующие действия:

получить несанкционированный доступ к секретной информации;

выдать себя за зарегистрированного пользователя, чтобы использовать его полномочия или снять с себя ответственность за НСД;

модифицировать программное обеспечение;

преднамеренно включить в программы специальные блоки для нарушения безопасности данных;

отказаться от факта формирования и выдачи данных;

утверждать о передаче данных какому-либо пользователю, хотя на самом деле данные не передавались;

отказаться от факта получения данных, которые на самом деле были получены;

незаконно расширить свои полномочия по доступу к информации и ее обработке;

незаконно изменить полномочия других пользователей;

скрыть факт наличия некоторых данных в других данных.

подключиться к линии связи между другими пользователями в качестве активного ретранслятора;

изучить права доступа пользователей (даже если сами данные остаются закрытыми);

преднамеренно изменить протокол обмена информацией с целью его нарушения или подрыва доверия к нему;

помешать обмену сообщениями между другими пользователями путем введения помех с целью нарушения аутентификации сообщений.

При подключении к магистральной линии связи нарушитель может осуществить следующие действия с передаваемыми данными:

1) раскрытие содержания передаваемых сообщений;

2) анализ трафика, позволяющий определить принадлежность отправителя и получателя данных к одной из групп пользователей сети, связанных общей задачей;

3) изменение потока сообщений, что может привести к нарушению режима работы какого-либо объекта, управляемого из удаленной ЭВМ;

4) неправомерный отказ в предоставлении услуг;

5) несанкционированное установление соединения.

1.4. Модель потенциального нарушителя

При разработке общей политики защиты ИС решающее значение имеет анализ модели потенциального нарушителя и его место в ИС.

Модель нарушителя определяет:

категории лиц, в числе которых может оказаться нарушитель;

возможные цели нарушителя и их градации по степени важности и опасности;

предположения о его квалификации;

оценка его технической вооруженности;

ограничения и предположения о характере его действий.

Можно выделить **четыре уровня возможностей реализации угрозы** и осуществления посягательства на информационные ресурсы ИС и, следовательно, **четыре уровня нарушителей информационной безопасности**. Классификация является иерархической, т.е. каждый следующий уровень включает в себя функциональные возможности предыдущего.

Первый уровень определяет самый низкий уровень возможностей. Нарушители осуществляют внешние угрозы (в основном, с помощью радиоэлектронных способов нарушения информационной безопасности) либо 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 компьютерной системы, 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, вплоть до включения в состав системы собственных технических средств с новыми функциями по обработке информации. К такому персоналу могут относиться разработчики информационных систем, системные программисты, администраторы баз данных, отдельные пользователи и соответствующие руководители подразделений.

Предполагается, что на своем уровне нарушитель является специалистом высшей квалификации, знает все об информационной системе и средствах ее защиты и может, при опреде-

ленных обстоятельствах, осуществить весь спектр посягательств на информационные ресурсы.

1.5. Способы мошенничества в информационных системах

Мошенничество (ст. 159 УК РФ) – хищение чужого имущества или приобретение права на чужое имущество путем обмана или злоупотребления доверием.

Компьютерные преступления (ст. 272, 273, 274 УК РФ) – неправомерный доступ к компьютерной информации; создание, использование и распространение вредоносных программ для ЭВМ; нарушение правил эксплуатации ЭВМ, системы ЭВМ или их сети, повлекшие уничтожение, блокирование, модификацию либо копирование информации, нарушение работы ЭВМ, системы ЭВМ или их сети.

В соответствии с этими понятиями под **способом мошенничества в ИС** понимается совокупность приемов и средств, обеспечивших несанкционированный доступ к информационным ресурсам и технологиям и позволивших их противоправно использовать.

Мошенничество в ИС является комплексным преступлением с тремя четко выраженными фазами (звеньями):

• подготовительной, когда осуществляется несанкционированный доступ к данным и машинной информации;

• основной, когда путем манипуляции данными и управляющими программами организуется движение информационных ресурсов;

• заключительной, когда маскируются следы мошенничества.

На **подготовительной фазе** реализуется один или комбинация из приемов:

1. Изъятие средств вычислительной техники (СВТ) путем хищения, разбоя, вымогательства (т.е. совершения обычных “некомпьютерных” преступлений). Объектами, как правило, являются системные блоки, содержащие в постоянной памяти

установочные данные обо всех клиентах, вкладчиках, кредиторах и т.д.

2. Перехват информации с использованием методов и аппаратуры аудио-, визуального и электромагнитного наблюдения (контроля). Объектами, как правило, являются каналы связи, телекоммуникационное оборудование, служебные помещения для проведения конфиденциальных переговоров, бумажные и магнитные носители (в том числе и технологические отходы).

3. Несанкционированный доступ к СВТ, который реализуется с использованием следующих основных приемов:

3.1. **“За дураком”** - проникновение как в производственные помещения (физическое), так и в электронные системы по следующей схеме:

• физическое проникновение - держа в руках предметы, связанные с работой на компьютерной технике (элементы маскировки), нужно ожидать кого-либо, имеющего санкционированный доступ, возле запертой двери, за которой находится предмет посягательства. Когда появляется законный пользователь, остается только войти внутрь вместе с ним или попросить его помочь занести якобы необходимые для работы на компьютере предметы;

• электронное проникновение - подключение компьютерного терминала к каналам связи с использованием шлейфа “шнурка” в тот момент времени, когда сотрудник кратковременно покидает свое рабочее место, оставляя терминал или персональный компьютер в активном режиме.

3.2. **“За хвост”** - преступник подключается к линии связи законного пользователя и дожидается сигнала, обозначающего конец работы, перехватывает его на себя, а потом, когда законный пользователь заканчивает активный режим, осуществляет доступ к системе. Подобными свойствами обладают телефонные аппараты с функцией удержания номера вызываемого абонентом.

3.3. **“Компьютерный абордаж”** - подбор вручную или с использованием автоматической программы кода доступа компьютерной системы с использованием обычного телефонного аппарата.

3.4. **“Неспешный выбор” (брешь, люк)** - изучение слабых мест в защите компьютерной системы, их исследований, выявление участков, имеющих ошибки или неудачную логику программного строения, разрыв программы и дополнительное введение команд.

3.5. **“Маскарад”** - проникновение в компьютерную систему, выдавая себя за законного пользователя с использованием его кодов и других идентифицирующих шифров.

3.6. **Мистификация** - создание условий, когда пользователь подключается к чьей-либо системе, будучи абсолютно уверенным в том, что он работает с нужным ему абонентом. Формируя правдоподобные ответы на запросы пользователя и поддерживая его заблуждения некоторое время, обычно добываются коды доступа или отклик на пароль.

3.7. **“Аварийный”** - создание условий для возникновения сбоев или других отклонений в работе ЭВМ, когда в компьютерном центре включается особая программа, позволяются в аварийном режиме получать доступ к наиболее ценным данным, как правило, в этом режиме “отключаются” все имеющиеся в системе средства защиты информации.

3.8. **“Салями”** - оригинальная электронная версия методов изъятия “лишних” денежных средств в свою пользу. При использовании этого метода злоумышленник так же, как и в предыдущем случае, “дописывает” прикладное программное обеспечение специальным модулем, который манипулирует с информацией, перебрасывая на подставной счет результаты округления при проведении законных транзакций. Расчет построен на том, что отчисляемые суммы столь малы, что их потери практически незаметны, а незаконное накопление средств проводится за счет суммы совершения большого количества операций (ст. 272, 273,158 УК РФ).

На **основной фазе** основными приемами манипуляции данными и управляющими программами, приводящими к движению информационных ресурсов, являются:

• подмена данных - изменение или введение новых данных, как правило, при вводе-выводе информации для приписывания адресным данным “чужой” истории.

• “троянский конь (матрешка, червь, бомба)” - тайное введение в программное обеспечение специальных программ, как правило, отчисляющих. Все манипуляции с данными производятся и контролируются этой программой в определенный заданный момент времени и при стечении благоприятных для преступника обстоятельств.

• “асинхронная атака” - используя асинхронную природу операционной системы, преступник может заставить работать при ложных условиях из-за чего управление обработкой частично или полностью нарушается. Эта ситуация используется для внесения изменений в операционную систему, причем вне ее эти изменения не будут заметны.

• “моделирование” - построение модели поведения ИС в различных условиях с целью оптимизации способа манипуляции данными и организации движения информационных ресурсов.

Заключительная стадия – это сокрытие следов несанкционированного доступа.

Дробление денежных сумм - злоумышленник делит полученные в результате несанкционированных манипуляций с банковской информацией денежные средства на неравные долевые части с зачислением на корреспондентские счета сторонних банков, в которых можно было бы впоследствии снять переведенные суммы наличными.

Переброска денежных средств - злоумышленник организует перевод полученных денежных сумм по счетам различных клиентов банка - и в результате - затрудняет возможность определения истинного пути происхождения

средств. Далее, когда “концы” потеряны, эти суммы можно использовать по своему усмотрению.

“Бухинг” (организация электронного блокирования) - банковская компьютерная система блокируется одновременной “атакой” несанкционированного доступа большим количеством злоумышленников (сообщников) со своих персональных компьютеров из различных регионов. Они организуют прикрытие одной основной незаконной транзакции огромным количеством фиктивных платежных поручений, которые затрудняют определение истинных путей утечки денежных средств.

Наиболее распространенные мотивы совершения компьютерных преступлений:

- корыстные побуждения – 66%;
- политические мотивы или государственные интересы – 17%;
- исследовательский интерес – 7%;
- хулиганские побуждения и озорство – 5%;
- обида и желание отомстить – 5%.

Наиболее распространенные цели совершения компьютерных преступлений:

- хищение денежных средств – 52%;
- разрушение и уничтожение средств компьютерной техники – 16%;
- подмена исходных данных – 12%;
- хищение информации и программ – 10%;
- хищение услуг – 10%.

2. КЛАССИФИКАЦИЯ МЕТОДОВ И СРЕДСТВ ЗАЩИТЫ ИНФОРМАЦИИ

Отметим, что **метод защиты данных** – это совокупность приемов и операций, реализующих функции защиты данных (например: методы аутентификации пользователей, методы шифрования и т.д.).

Средство защиты – это инструменты, устройства, программы, законы и другие нормативные акты (например программные, аппаратные или программно-аппаратные средства), реализующее защиту данных определенным методом (устройства шифрации/дешифрации, программы анализа пароля, датчики охранной сигнализации, законы об авторских правах).

Механизм защиты – это совокупность средств защиты, функционирующих совместно для выполнения определенной задачи по защите данных (пример: криптографические протоколы, механизмы защиты операционных систем, баз данных и т.д.).

На первом этапе развития концепций обеспечения безопасности данных преимущество отдавалось программным методам и средствам защиты. По мере формирования системного подхода к проблеме обеспечения безопасности данных, возникла необходимость комплексного применения различных методов защиты и созданных на их основе средств и механизмов защиты.

2.1. Методы защиты информации

Основные методы защиты данных следующие:

управление;
препятствия;
маскировка;
регламентация;
побуждение;
принуждение.

Управление представляет собой регулирование использования всех ресурсов системы в рамках установленного технологического цикла обработки и передачи данных, где в качестве ресурсов рассматриваются технические средства, ОС, программы, БД, элементы данных и т.п. Управление защитой данных реализует процесс целенаправленного

воздействия подсистемы управления СОБД на средства и механизмы защиты данных и компоненты ИВС с целью обеспечения безопасности данных.

Препятствия физически преграждают нарушителю путь к защищаемым данным.

Маскировка представляет собой метод защиты данных путем их криптографического закрытия.

Регламентация заключается в разработке и реализации в процессе функционирования ИВС комплексов мероприятий, создающих такие условия технологического цикла обработки данных, при которых минимизируется риск НСД к данным. Регламентация охватывает как структурное построение ИВС, так и технологию обработки данных, организацию работы пользователей и персонала сети.

Побуждение состоит в создании такой обстановки и условий, при которых правила обращения с защищенными данными регулируются моральными и нравственными нормами.

Принуждение включает угрозу материальной, административной и уголовной ответственности за нарушение правил обращения с защищенными данными.

2.2. Классификация средств защиты информации

На основе перечисленных методов создаются **средства защиты информации** (рис. 2.).

Формальными называются такие средства защиты, которые выполняют свои функции по заранее установленным процедурам без вмешательства человека.

К формальным средствам защиты относятся технические и программные средства.

К **техническим** средствам защиты относятся все устройства, которые предназначены для защиты данных. В свою очередь, технические средства защиты можно разделить на физические и аппаратные.

Рис. 2. Классификация средств защиты информации

Физическими называются средства защиты, которые создают физические препятствия на пути к защищаемым данным и не входят в состав аппаратуры ИВС, а **аппаратными** - средства защиты данных, непосредственно входящие в состав аппаратуры ИВС.

Программными называются средства защиты данных, функционирующие в составе программного обеспечения ИВС.

Отдельную группу формальных средств составляют **криптографические** средства, которые реализуются

в виде программных, аппаратных и программно-аппаратных средств защиты.

2.3. Организационные средства защиты информации

К методам и средствам **организационной защиты** информации относятся организационно-технические и организационно-правовые мероприятия, проводимые в процессе создания и эксплуатации компьютерных систем (КС) с целью защиты данных. Эти мероприятия должны проводиться:

при строительстве или ремонте помещений, в которых будет размещаться КС;

проектировании системы, монтаже и наладке ее технических и программных средств;

испытаниях и проверке работоспособности КС.

Методы и средства организационной защиты должны обеспечить:

полное или частичное перекрытие значительной части каналов утечки информации (например, хищения или копирования носителей информации);

объединение всех используемых в КС средств в целостный механизм защиты информации.

Методы и средства организационной защиты информации включают в себя:

ограничение физического доступа к объектам КС и реализация режимных мер;

ограничение возможности перехвата побочных электромагнитных излучений и наводок (ПЭМИН);

разграничение доступа к информационным ресурсам и процессам КС (установка правил разграничения доступа, шифрование информации при ее хранении и передаче, обнаружение и уничтожение аппаратных и программных закладок);

резервное копирование наиболее важных с точки зрения утраты массивов документов;

профилактику заражения компьютерными вирусами.

Перечислим основные виды мероприятий, которые должны проводиться на различных этапах жизненного цикла КС.

На этапе создания КС должны осуществляться:

при разработке общего проекта КС и ее отдельных структурных элементов – анализ возможных угроз и методов их нейтрализации;

при строительстве и переоборудовании помещений – приобретение сертифицированного оборудования, выбор лицензированных организаций;

при разработке математического, программного, информационного лингвистического обеспечения – использование сертифицированных программных и инструментальных средств;

при монтаже и наладке оборудования – контроль за работой технического персонала;

при испытаниях и приемке в эксплуатацию – включение в состав аттестационных комиссий сертифицированных специалистов.

В процессе эксплуатации КС должны осуществляться:

организация пропускного режима;

определение технологии автоматизированной обработки документов;

организация работы обслуживающего персонала;

распределение реквизитов разграничения доступа пользователей к элементам КС (паролей, ключей, карт и т.п.);

организация ведения протоколов работы КС;

контроль выполнения требований служебных инструкций и т.п.

Мероприятия общего характера:

подбор и подготовка кадров;

организация плановых и предупреждающих проверок средств защиты информации;

планирование мероприятий по защите информации;

обучение персонала, участие в семинарах, конференциях и выставках по проблемам защиты безопасности информации и т.п.

2.4. Законодательные средства защиты информации

Законодательные и нормативные документы в области информационной безопасности на правовом уровне должны регулировать доступ к информации со стороны потребителей. В российском законодательстве позже, чем в законодательстве других развитых стран, появились необходимые правовые акты, хотя далеко не все.

Можно выделить **четыре уровня правового обеспечения информационной безопасности**.

Первый уровень образуют международные договоры, к которым присоединилась Российская Федерация и федеральные законы России.

Это следующие документы:

международные всемирные конвенции об охране промышленной собственности, охране интеллектуальной собственности, авторском праве;

Конституция РФ (ст. 23 определяет право граждан на тайну переписки, телефонных, телеграфных и иных сообщений);

Гражданский кодекс РФ (в ст. 139 устанавливается право на возмещение убытков от утечки информации с помощью незаконных методов; при этом рассматривается информация, относящаяся к служебной и коммерческой тайне);

Уголовный кодекс РФ (ст. 272 - устанавливает ответственность за неправомерный доступ ук компьютерной безопасности; ст. 273 – за создание, использование и распространение вредоносных программ для ЭВМ; ст. 274 – за нарушение правил эксплуатации ЭВМ, систем и сетей);

Федеральный закон «Об информации, информатизации и защите информации» от 20.02.95 № 24-ФЗ (ст. 10 устанавлива-

ет разнесение информационных ресурсов по категориям доступа: открытая информация, государственная тайна, конфиденциальная информация, ст. 21 определяет порядок защиты информации);

Федеральный закон «О государственной тайне» от 21.07.93 № 5485-1 (ст. 5 устанавливает перечень сведений, составляющих государственную тайну; ст. 8 –степень секретности сведений и грифы секретности их носителей: «особой важности», «совершенно секретно», и «секретно»; ст. 20 – органы по защите государственной тайны, межведомственную комиссию по защите государственной тайны для координации деятельности этих органов; ст. 28 – порядок сертификации средств защиты информации, относящейся к государственной тайне);

Федеральные законы «О лицензировании отдельных видов деятельности» от 08.08.2001 № 128-ФЗ, «О связи» от 16.02.95 № 15-ФЗ, «Об электронной цифровой подписи» от 10.01.02 № 1-ФЗ, «Об авторском праве и смежных правах» от 09.07.93 № 5361-1, «О правовой охране программ для электронных вычислительных машин и баз данных» от 23.09.92 № 3523-1 (ст. 4 определяет условие признания авторского права – знак @ с указанием правообладателя и первого года выпуска продукта в свет; ст. 18 – защиту прав на программы для ЭВМ и базы данных путем выплаты компенсации в размере от 5000 до 50 000 минимальных размеров оплаты труда при нарушении этих прав с целью извлечения прибыли или путем возмещения причиненных убытков, в сумму которых включаются полученные нарушителем доходы).

Второй уровень правового обеспечения информационной безопасности составляют подзаконные акты, к которым относятся указы Президента РФ и постановления Правительства РФ, а также письма Высшего Арбитражного Суда РФ и постановления пленумов Верховного Суда РФ. Примерами таких актов могут являться Указ Президента РФ «Об утверждении перечня сведений конфиденциального характера» от

06.03.97 № 188 или Постановление Правительства РФ «О перечне сведений, которые не могут составлять коммерческую тайну» от 05.12.91 № 35.

Третий уровень правового обеспечения информационной безопасности составляют государственные стандарты (ГОСТы) в области защиты информации, руководящие документы, нормы, методики и классификаторы, разработанные соответствующими государственными органами.

В качестве примеров можно привести следующие документы:

ГОСТ Р 50922-96 «Защита информации. Основные термины и определения», ГОСТ Р 50739-95 «Средства вычислительной техники. Защита от несанкционированного доступа к информации. Общие технические требования», ГОСТ 28147-89 «Системы обработки информации. Защита криптографическая. Алгоритмы криптографического преобразования» и др.;

руководящие документы Государственной технической комиссии при Президенте Российской Федерации (Гостехкомиссии России) «Концепция защиты средств вычислительной техники и автоматизированных систем от несанкционированного доступа к информации», «Автоматизированные системы. Защита от несанкционированного доступа к информации. Классификация автоматизированных систем и требования по защите информации» и др.

Четвертый уровень правового обеспечения информационной безопасности образуют локальные нормативные акты, положения, инструкции, методические рекомендации и другие документы по комплексной защите информации в КС конкретной организации.

К таким нормативным документам относятся:

приказ об утверждении перечня сведений, составляющих коммерческую тайну предприятия;

трудовые и гражданско-правовые договоры (подряда, поручения, комиссии и т.п.), в которые включены пункты об обя-

занности возмещения ущерба за разглашение сведений, составляющих коммерческую тайну предприятия, и др.

2.5. Физические средства защиты данных

К **физическим средствам** защиты информации относят механические, электронно-механические, электромеханические, оптические, акустические, лазерные, радио и радиационные и другие устройства, системы и сооружения, предназначенные для создания физических препятствий на пути к защищаемой информации и способные выполнять самостоятельно или в комплексе с другими средствами функции защиты информации. Физические средства представляют собой первый рубеж защиты информации и элементов вычислительных систем, и поэтому обеспечение физической целостности таких систем и их устройств является необходимым условием защищенности информации.

Физические средства защиты выполняют следующие основные задачи:

1) охрана территории и зданий;
2) охрана внутренних помещений;
3) охрана оборудования и наблюдение за ним;
4) контроль доступа в защищаемые зоны;
5) нейтрализация излучений и наводок;
6) создание препятствий визуальному наблюдению и подслушиванию;
7) противопожарная защита;
8) блокировка действий нарушителя и т.п.

В соответствии с этими задачами можно предположить следующую **классификацию основных существующих средств физической защиты.**

1. Механические преграды
   1.1. Заборы, ограды, решетки, ставни, экраны
   1.2. Специальное остекленение
   1.3. Сейфы, шкафы

1.4. Механические замки

1.5. Сложные замки с кодовым набором, с управлением от микропроцессора, радиоуправляемые и т.д.

2. Датчики различного типа

2.1. Сверхвысокочастотные (СВЧ), ультразвуковые (УЗ) и инфракрасные (ИК) системы

2.2. Лазерные и оптические системы

2.3. Акустические системы

2.4. Телевизионные системы

2.5. Кабельные системы

2.6. Системы защиты окон и дверей

3. Устройства идентификации

4. Устройства идентификации по физическим признакам

5. Устройства пространственного зашумления, сетевые помехоподавляющие фильтры.

6. Системы пожаротушения и датчики огня, дыма

Механические преграды предназначены для препятствия механическому проникновению, а также для защиты от наблюдения и подслушивания.

Регулирование доступа на территорию и в помещения может осуществляться и с помощью специальных замков, в том числе замков с управлением от микропроцессоров и ЭВМ и с содержанием микропроцессоров.

Датчики различного типа необходимы для предотвращения проникновения нарушителей на охраняемые объекты.

СВЧ, УЗ и ИК системы

Предназначены для обнаружения движущихся объектов, определения их размеров, скорости и направления перемещения. Принцип их действия основан на изменении частоты отраженного от движущегося объекта сигнала.

ИК системы бывают активными и пассивными. Активные системы содержат источник излучения и его приемник. Функционирование пассивных систем основано на фиксации теплового излучения ИК-датчиками.

УЗ и ИК системы применяются, главным образом, внутри помещений. СВЧ системы могут применяться как внутри помещений, так и для охраны зданий и территории.

Лазерные и оптические системы, работающие в видимой части спектра, реагируют на пересечение нарушителями светового луча и применяются, в основном, внутри помещений.

Телевизионные системы применяются для наблюдения как за территорией охраняемого объекта, так и за обстановкой внутри помещений.

Кабельные системы используются для охраны небольших объектов, обычно временно находящихся на территории, а также оборудования внутри помещений. Они состоят из заглубленного кабеля, окружающего защищаемый объект и излучающего радиоволны. Приемник излучения реагирует на изменение поля, создаваемого нарушителем.

Системы защиты окон и дверей предназначены для препятствия механическому проникновению, а также для защиты от наблюдения и подслушивания.

Устройства идентификации необходимы для идентификации пользователей:

- ***пластиковые карты с магнитной полосой***, на которой помимо ключевой информации могут размещаться и дополнительные реквизиты пользователя (его фамилия, имя, отчество, фотография, название организации и ее подразделения и т.п.); подобные карты наиболее дешевы, но и наименее защищены от копирования и подделки;

- ***карты со штрихкодом***, покрытым непрозрачным составом, считывание информации с которых происходит в инфракрасных лучах; эти карты также относительно дешевы, но уязвимы для подделки;

- ***смарт-карты***, носителем ключевой информации в которых является специальная бескорпусная микросхема, включающая в себя только память для хранения ключевой информации (простые смарт-карты) или микропроцессор (ин-

теллектуальные карты), позволяющий реализовывать достаточно сложные процедуры идентификации и аутентификации;

- ***элементы Touch Memory*** (аналогичные изделия других производителей именуются iButton), включающие в себя энергонезависимую память в виде постоянного запоминающего устройства с уникальным для каждого изделия серийным номером и (в более дорогих вариантах) оперативного запоминающего устройства для хранения идентифицирующей пользователя информации, а также встроенный элемент питания со сроком службы до 10 лет (элемент Touch Memory напоминает миниатюрную батарейку диаметром 16 мм и толщиной 3...6 мм, он имеет один сигнальный контакт и один контакт заземления, а для контакта элемента с Устройством чтения достаточно простого касания);

- ***маркеры eToken (USB-брелки)***, представляющие собой подключаемое к USB-порту компьютера устройство, которое включает в себя аналогичную смарт-карте микросхему с процессором защищенной от несанкционированного доступа памятью (в отличие от пластиковых карт не требуется установка устройства их чтения с кабелем для подключения этого устройства к компьютеру).

Устройства идентификации по физическим признакам основаны на анализе следующих биометрических характеристик человека:

- узор радужной оболочки и сетчатки глаз;
- отпечатки пальцев;
- геометрическая форма руки;
- форма и размеры лица;
- особенности голоса;
- биомеханические характеристики рукописной подписи;
- биомеханические характеристики «клавиатурного почерка».

При регистрации пользователь должен продемонстрировать один или несколько раз свои характерные биометрические признаки. Эти признаки (известные как подлинные)

регистрируются системой как контрольный «образ» законного пользователя. Этот образ пользователя хранится в электронной форме и используется для проверки идентичности каждого, кто выдает себя за соответствующего законного пользователя.

Системы идентификации по узору радужной оболочки и сетчатки глаз могут быть разделены на два класса:

- использующие рисунок радужной оболочки глаза;
- использующие рисунок кровеносных сосудов сетчатки глаза.

Поскольку вероятность повторения данных параметров равна 10^{-78}, эти системы являются наиболее надежными среди всех биометрических систем. Такие средства применяются, например, в США в зонах военных и оборонных объектов.

Системы идентификации по отпечаткам пальцев являются самыми распространенными. Одна из основных причин широкого распространения таких систем заключается в наличии больших банков данных по отпечаткам пальцев. Основными пользователями таких систем во всем мире являются полиция, различные государственные и некоторые банковские организации.

Системы идентификации по геометрической форме руки используют сканеры формы руки, обычно устанавливаемые на стенах. Следует отметить, что подавляющее большинство пользователей предпочитают системы именно этого типа, а не описанные выше.

Системы идентификации по лицу и голосу являются наиболее доступными из-за их дешевизны, поскольку большинство современных компьютеров имеют видео- и аудиосредства. Системы данного класса широко применяются при удаленной идентификации в телекоммуникационных сетях.

Системы идентификации по динамике рукописной подписи учитывают интенсивность каждого усилия подписывающегося, частотные характеристики написания каждого элемента подписи и начертания подписи в целом.

Системы идентификации по биомеханическим характеристикам «клавиатурного почерка» основываются на том, что моменты нажатия и отпускания клавиш при наборе текста на клавиатуре существенно различаются у разных пользователей. Этот динамический ритм набора («клавиатурный почерк») позволяет построить достаточно надежные средства идентификации.

Для защиты от перехвата электромагнитного излучения применяются *экранирование* и *зашумляющие генераторы излучений*.

Устройства пространственного зашумления, сетевые помехоподавляющие фильтры.

Данные устройства реализуют методы и средства защиты информации от утечки по каналам перехвата побочных электромагнитных излучений и наводок (ПЭМИН).

Основной задачей является уменьшение соотношения сигнал/шум в этих каналах до предела, при котором восстановление информации становится принципиально невозможным.

Возможными методами решения этой задачи могут быть:

1) снижение уровня излучения сигналов в аппаратных средствах КС;

2) увеличение мощности помех в соответствующих этим сигналам частотных диапазонах.

Для применения первого метода необходим выбор системно-технических и конструкторских решений при создании технических средств КС в защищенном исполнении, также рациональный выбор места размещения этих средств относительно мест возможного перехвата ПЭМИН (для соблюдения условия максимального затухания информационного сигнала). Требования к средствам вычислительной техники в защищенном исполнении определяются в специальных ГОСТах.

Реализация второго метода возможна путем применения активных средств защиты в виде генераторов сигналоподобных помех или шума.

Отметим перспективные методы и средства защиты информации в КС от утечки по каналам ПЭМИН:

выбор элементной базы технических средств КС с возможно более малым уровнем информационных сигналов;

замена в информационных каналах КС электрических цепей волоконно-оптическими линиями;

локальное экранирование узлов технических средств, являющихся первичными источниками информационных сигналов;

включение в состав информационных каналов КС устройств предварительного шифрования обрабатываемой информации.

Примеры генераторов шума.

Стационарный генератор шума ГНОМ-3(рис. 3) используется для защиты помещений и объектов с электронно-вычислительной техникой от утечки конфиденциальной информации за счет побочных электромагнитных излучений компьютеров и другой оргтехники.

Рис. 3. Стационарный генератор шума ГНОМ-3

Универсальный шумогенератор Гром-ЗИ-4 (рис.4) используется для создания шумовой помехи по радиоканалу, телефонной линии и электросети для блокировки несанкционированно установленных устройств, передающих информацию; предотвращает съем информации с персональных компьютеров и локальных вычислительных сетей на базе ПК по побочным электромагнитным излучениям.

Рис. 4. Универсальный генератор шума Гром-ЗИ-4

Стационарный генератор шума ГШ-1000 (рис. 5) используется для маскирования побочных электромагнитных излучений и наводок работающих ПЭВМ.

Рис. 5. Универсальный генератор шума ГШ-1000

Генератор шума ГШК-1000 используется для защиты ПЭВМ от перехвата обрабатываемой информации за счет по-

бочных электромагнитных излучений и наводок; выполнен в виде платы, вставляемой в свободный слот PCI материнской платы ПЭВМ, включается вместе с ПЭВМ; антенна шумогенератора выводится через отверстие на задней панели компьютера и закрепляется на каркасе.

Импульсный подавитель КОБРА (рис. 6) предназначен для электрического подавления (уничтожения) в отключенных проводных коммуникациях устройств несанкционированного съема информации путём подачи в линию импульсного высоковольтного разряда.

Рис. 6. Импульсный подавитель КОБРА

Портативный акустический шумогенератор PNG-200 (рис. 7) может быть использован для оперативной защиты пространства помещения при проведении переговоров.

Рис. 7. Портативный акустический шумогенератор PNG-200

Генератор шума по сети 220 В SEL SP-41/C (рис. 8) является техническим средством активной защиты информации

объектов 1 категории, создающим маскирующий сигнал в цепях электропитания; предназначен для защиты информации, обрабатываемой средствами оргтехники, от утечки по сети электропитания, а также для подавления устройств несанкционированного съема информации, использующих в качестве канала передачи цепи электропитания 220 В.

Рис. 8. Генератор шума по сети 220 В SEL SP-41/C

Генератор шума по сети электропитания и линиям заземления "Соната-РС1" (рис. 9) предназначен для активной защиты объектов ЭВТ (объектов информатизации) от утечки информации в форме информативных электрических сигналов, возникающих в сети электропитания, системе заземления, инженерных коммуникациях и т.п.

Рис. 9. Генератор шума по сети Соната-РС1

Фильтр сетевой помехоподавляющий ФСП-1Ф-7А, ФСП-3Ф-10А (рис. 10) предназначен для защиты радиоэлектронных устройств и средств вычислительной техники от утечки информации по цепям электропитания с напряжением 220 В. Фильтр применяется для обеспечения электромагнитной развязки по цепям электропитания радиоэлектронных устройств и электросети промышленных объектов и офисных помещений.

Рис. 10. Фильтр сетевой помехоподавляющий ФСП-1Ф-7А

Отметим, что при использовании технических средств КС для обработки информации ограниченного доступа необходимо проведение специальных проверок, целью которых является обнаружение и устранение внедренных специальных электронных устройств подслушивания, перехвата информации или вывода технических средств из строя (аппаратных закладок).

При проведении таких проверок может потребоваться практически полная их разборка, что иногда может привести к возникновению неисправностей в работе технических средств и дополнительным затратам на их устранение.

Рассмотрим средства обнаружения электронных подслушивающих (радиозакладных) устройств, простейшими из которых являются нелинейные локаторы.

Нелинейные локаторы с помощью специального передатчика в сверхвысокочастотном диапазоне радиоволн облучают окружающее пространство и регистрируют вторичный, переизлученный сигнал, поступающий от различных полупроводниковых элементов, находящихся как во включенном, так и в выключенном состоянии.

Нелинейные локаторы могут не выявить радиозакладное устройство, если оно вмонтировано в электронное устройство (системный блок компьютера, телевизор, телефонный аппарат и т.п.), так как сигнал отклика от подслушивающего устройства будет замаскирован откликом от электронной аппаратуры. В этом случае потребуется применение более сложных устройств контроля постороннего радиоизлучения – индикаторов электромагнитного излучения, сканирующих приемников, компьютерных анализаторов.

2.6. Аппаратные и программные средства защиты информации

К аппаратным средствам защиты информации относятся электронные и электронно-механические устройства, включаемые в состав технических средств КС и выполняющие (самостоятельно или в едином комплексе с программными средствами) некоторые функции обеспечения информационной безопасности.

К **основным аппаратным средствам** защиты информации относятся:

устройства для ввода идентифицирующей пользователя информации (магнитных и пластиковых карт, отпечатков пальцев и т.п.);

устройства для шифрования информации;

устройства для воспрепятствования несанкционированному включению рабочих станций и серверов (электронные замки и блокираторы).

Примеры вспомогательных аппаратных средств защиты информации:

устройства уничтожения информации на магнитных носителях;

устройства сигнализации о попытках несанкционированных действий пользователей КС и др.

К основным **программным средствам защиты информации** относятся:

программы идентификации и аутентификации пользователей КС;

программы разграничения доступа пользователей к ресурсам КС;

программы шифрования информации;

программы защиты информационных ресурсов (системного и прикладного программного обеспечения, баз данных, компьютерных средств обучения и т.п.) от несанкционированного изменения, использования и копирования.

Под **идентификацией**, применительно к обеспечению информационной безопасности КС, понимают **однозначное распознавание уникального имени субъекта КС**.

Аутентификация означает **подтверждение того, что предъявленное имя соответствует данному субъекту** (подтверждение подлинности субъекта).

Примеры **вспомогательных программных средств защиты информации**:

программы уничтожения остаточной информации (в блоках оперативной памяти, временных файлах и т.п.);

программы аудита (ведения регистрационных журналов) событий, связанных с безопасностью КС, для обеспечения возможности восстановления и доказательства факта происшествия этих событий;

программы имитации работы с нарушителем (отвлечение его на получение якобы конфиденциальной информации);

программы тестового контроля защищенности КС и др.

К **преимуществам программных средств защиты информации** относятся:

простота тиражирования;

гибкость (возможность настройки на различные условия применения, учитывающие специфику угроз информационной безопасности конкретных КС);

простота применения – одни программные средства, например шифрования, работают в «прозрачном» (незаметном для пользователя) режиме, а другие не требуют от пользователя никаких новых (по сравнению с другими программами) навыков;

практически неограниченные возможности их развития путем внесения изменений для учета новых угроз безопасности информации.

К **недостаткам программных средств защиты информации** относятся:

снижение эффективности КС за счет потребления ее ресурсов, требуемых для функционирования программ защиты;

более низкая производительность (по сравнению с аппаратными средствами защиты, выполняющими аналогичные функции, например, функции шифрования);

пристыкованность многих программных защиты, а не встроенность, что создает для нарушителя принципиальную возможность их обхода;

возможность злоумышленного изменения программных средств защиты в процессе эксплуатации КС.

Можно привести следующие программно-аппаратные средства защиты информации от несанкционированного доступа:

Cisco Security (производитель Cisco Systems);

RuToken (компания «Актив»);

СРД «Щит» (ООО фирма «АНКАД»);

КСЗИ «Блокада-НР» (ЗАО «НПП Информационные технологии в бизнесе»);

СЗИ Secret Net 5.0 (компания «Информзащита»);

КСЗИ «Панцирь-К» (ЗАО «НПП «Информационные технологии в бизнесе»»);

ПАК СЗИ НСД Аккорд-NT/2000 (ОКБ САПР) и др.

Примеры наиболее распространенных программно-аппаратных средств защиты информации от несанкционированного доступа (НСД), а также их функциональные возможности приведены в таблице.

Программно-аппаратные средства защиты

Средство защиты информации от НСД	RuToken	«Щит»	ПАК СЗИ НСД Аккорд NT/ 2000
Производитель	Компания «Актив»	ООО Фирма «Анкад»	ОКБ САПР
Вид средства	Программно-аппарат-ное средство (ПАС)	ПАС	ПАС

Продолжение таблицы

Активное или пассивное средство (наличие или отсутствие собственного процессора)	Активное	Активное	Активное средство на базе собственного процессора, независимого от процессора ПК
Наличие средств централизованного управления и/или аудита	В составе программного обеспечения партнеров	Да	Подсистема Аккорд-РАУ: распределенный аудит и управление
Реализация механизмов защиты на уровне ОС	Поддержка MS CryptoAPI (криптографический интерфейс приложений OCWindows)	Мандатный и дискреционный принципы разгран-я доступа к программам, разграничение и контроль доступа к USB-устройствам, аудит событий	ПО Аккорд-NT обеспечивает разграничение доступа, создание изолированной программной среды для каждого пользователя, управ-е потоками инф-и, ведение протокола и т.д.

Продолжение таблицы

Реализация механизмов защиты микропроцессором устройства	Аппаратная реализация ГОСТ 28147-89, аппаратная аутентификация по PIN-коду	Идентификация и аутентификация пользователей, разграничение доступа к устройствам, блокировка по RESET, управление внешними устройствами	Контроллер АМДЗ обеспечивает доверенную загрузку ОС (блокировка загрузки с отчуждаемых носителей), идентификация / аутентификация пользователей, контроль целостности аппаратных средств и ПО
Контроль доступа к локальным устройствам и сетевым ресурсам	На уровне MS PKI	Есть разграничения доступа к жестким дискам, НГМД, и сетевым картам	Контроль подключения устройств по USB-интерфейсу, контроль печати, доступа к файлам, каталогам, дискам, расположенным на локальных дисках и сетевых серверах.

Продолжение таблицы

Наличие и тип сертификата ФСТЭК / ФСБ / Минобороны / ГОСТ 15408	Сертификат ФСТЭК, сертификат ФСБ в работе	Сертификат ФСТЭК	Сертификаты ФСТЭК (Гостехкомиссии) России, ФАПСИ (ФСБ России), Минобороны
Возможность шифрования данных: 1. Шифрование для собственных нужд. 2. Шифрование произвольных данных пользователя. 3. Шифрование обеспечивается аппаратно / программно	1. Да 2. Да 3. Да	Проходное шифрование («на лету») произвольных данных пользователя на жестких, USB и Flash–дисках с помощью аппаратных модулей	1. Есть во всех контролерах. 2. Есть в контроллере АМД3 Аккорд-5.5: шифрование по ГОСТ 28147-89; выч-е хэш-функций по ГОСТ Р 34.11-94; выч-е и проверка ЭЦП по ГОСТР34.10-94 и по ГОСТ РР34.10-2001; выч-е защитных кодов аутентификации. 3. Реализация аппаратная.

Продолжение таблицы

Наличие средств очистки памяти (на локальных дисках, на внешних носителях, на сетевых дисках)	rtAdmin	Есть на локальных дисках	Есть, посредством ПО Аккорд-NT – многопроходовое затирание данных при удалении файлов и при освобождении оперативной памяти
Возможность контроля целостности произвольных файлов, ветвей реестра Windows	Нет	Есть	Есть и на аппаратном, и на программной уровне. Кроме того, на программном уровне осуществляется при запуске контроль целостности файлов.
Поддерживаемые ОС	Семейство ОС Windows	Windows NT, 2000, XP	Windows 9x/ NT/ 2000/2003/ XP/VISTA UNIX, LINUX и др.

Окончание таблицы

Возможные средства авторизации	RuToken сам представляет собой средство авторизации	Идентификаторы Touch Memory	ТМ-идентификатор, смарт-карта, устр-во считывания отпечатка пальца, ПСКЗИ ШИПКА
Возможность использования в терминальных решениях	Да	Да	Да
Поддерживаемые методы разграничения доступа	Разграничение доступа по ролям («Гость», «Пользователь», «Администратор»)	Мандатный, дискреционный	Мандатный, дискреционный
Класс защищенности	Не указывается	3-й класс защищенности для СВТ, 2-й уровень контроля НДВ.	3-й класс защищенности по СВТ, 2-й уровень контроля по НДВ, может использоваться в АС, требующих класса защищ-и до 1Б включ-о.

2.7. Требования к комплексным системам защиты информации

Поскольку потенциальные угрозы безопасности информации весьма разнообразны и многообразны, эффективная защита информации возможна только путем создания комплексной системы защиты информации (КСЗИ).

Комплексная система защиты информации – совокупность методов и средств, объединенных единым целевым назначением и обеспечивающих необходимую эффективность защиты информации в КС.

Основные требования к комплексной системе защиты информации*:*

разработка системы защиты на основе положений и требований существующих законов, стандартов и нормативно-методических документов по обеспечению информационной безопасности;

использование комплекса организационных мер, физических и программно-аппаратных средств для защиты КС;

надежность, производительность, конфигурируемость;

экономическая целесообразность (стоимость КСЗИ включается в стоимость КС и поэтому стоимость средств защиты не должна быть выше возможного ущерба от потери информации);

защита информации должна осуществляться на всех этапах жизненного цикла обработки информации в КС (в том числе при проведении ремонтных и регламентных работ);

возможность совершенствования;

обеспечение не только пассивной, но и активной защиты (например, обеспечение разграничения доступа к конфиденциальной информации с отвлечением нарушителя на ложную информацию);

взаимодействие с незащищенными КС должно осуществляться по установленным для этого правилам разграничения доступа;

учет и расследование случаев нарушения безопасности информации в КС;

возможность оценки эффективности применения системы защиты.

2.8. Стандарты безопасности КС

Впервые основные требования к системе защиты информации были сформулированы в документе Министерства обороны США «Trusted Computer System Evaluation Criteria» («Критерии оценки безопасности компьютерных систем», или более известной (по цвету обложки) под названием «Оранжевая книга») в 1985 г. В этом документе предложены три основные категории требований.

1. Политика:

наличие явной и хорошо определенной политики обеспечения безопасности;

использование маркировки объектов КС для управления доступом к ним.

2. Подотчетность:

индивидуальная идентификация субъектов КС;

сохранение и защита информации аудита.

3. Гарантии:

включение в состав КС программно-аппаратных средств защиты для гарантированного выполнения требований 1 и 2 категории;

постоянная защищенность средств обеспечения безопасности информации в КС от их преодоления и (или) несанкционированного изменения.

В «Оранжевой книге» были введены семь классов защищенности КС. Минимальная защита – класс D1. Верхний класс – класс A1.

Требования «Оранжевой книги» явились первой попыткой создать единый стандарт безопасности КС, рассчитанный

на проектировщиков, разработчиков (программистов), пользователей подобных систем и специалистов по их сертификации.

Отличительной чертой этого стандарта является ориентация на государственные (в первую очередь военные) организации и существующие операционные системы.

В 1992 году Гостехкомиссия России опубликовала первый комплект руководящих документов по защите средств вычислительной техники (СВТ) и автоматизированных систем (АС) от несанкционированного доступа.

СВТ используются в качестве элементов АС и непосредственно не решают прикладных задач. В качестве примере СВТ, используемых как элемент АС, можно привести плату расширения BIOS с соответствующим аппаратным и программным интерфейсом для аутентификации пользователей АС или программу шифрования информации на жестком диске.

В руководящих документах Гостехкомиссии России определены **семь классов защищенности СВТ от несанкционированного доступа** к обрабатываемой (сохраняемой, передаваемой) с помощью этих средств информации (наиболее защищенным является первый класс).

АС рассматривается как комплекс СВТ и имеет дополнительные характеристики: полномочия пользователей, модель нарушителя, технология обработки и передачи информации. Типичным примером АС является многопользовательская и многозадачная операционная система.

В руководящих документах Гостехкомиссии России определены **девять классов защищенности АС от несанкционированного доступа**, объединенных в три группы:

однопользовательские АС с информацией, размещенной на носителях одного уровня конфиденциальности (класс 3Б, 3А);

многопользовательские АС с одинаковыми полномочиями пользователей и информацией на носителях разного уровня конфиденциальности (классы 2Б, 2А);

многопользовательские АС с разными полномочиями пользователей и информацией разного уровня конфиденциальности (в порядке возрастания защищенности от класса 1Д до класса 1А).

Под **несанкционированным доступом** к информации в руководящих документах Гостехкомиссии России понимается **доступ к информации, нарушающий установленные правила разграничения доступа** и **использующий штатные возможности СВТ и АС**.

Руководящие документы Гостехкомиссии России, подобно «Оранжевой книге», ориентированы прежде всего на применение в КС силовых структур Российской Федерации.

Дальнейшее развитие стандартов в области информационной безопасности КС завершилось на сегодняшний день принятием «Общих критериев безопасности информационных технологий» (Common Criteria for Information Technology Security Evaluation). За этим документом исторически закрепилось более короткое название – «Общие критерии», или ОК. В создании этого документа приняли участие правительственные организации следующих стран: Канада, США, Великобритания, Германия, Нидерланды и Франция. ОК имеют несколько версий, например, версия 2.1 принята в августе 1999 года.

«Проект ОК» с самого начала носил не только технический, но и экономико-политический характер. Его цель состояла, в частности, в том, чтобы упростить, удешевить и ускорить выход сертифицированных изделий информационных технологий (ИТ) на мировой рынок. Для этого в мае 2000 года уполномоченные правительственные организации шести стран-основателей «Проекта ОК», а также Австралии и Новой Зеландии, Греции, Италии, Испании, Норвегии, Финляндии и Швеции подписали соглашение «О признании сертификатов по Общим критериям в области безопасности информационных технологий» (позднее к нему присоединились Австрия и Израиль).

«Общие критерии…» адресованы трем группам специалистов (пользователям, разработчикам и экспертам по классификации КС) и представляют собой новый межгосударственный уровень в стандартизации безопасности информационных технологий.

Участие в соглашении предполагает соблюдение двух независимых условий: признание сертификатов, выданных соответствующими органами других стран-участниц, а также возможность осуществления подобной сертификации.

По данным на конец 2002 года, правом выдачи сертификатов, признаваемых участниками соглашения, обладали Австралия, Новая Зеландия, Великобритания, Германия, Канада, США и Франция.

К началу 2003 года сертификаты по «Общим критериям» получили около семидесяти разнообразных изделий ИТ ведущих производителей:

операционные системы,

системы управления базами данных,

межсетевые экраны, коммуникационные средства и т.п.

В 1999 году была организована работа по подготовке российского стандарта и Руководящего документа (РД) Гостехкомиссии России на основе «Общих критериев». Она велась в тесном контакте с зарубежными коллегами и успешно завершена в 2002 году.

В Российской Федерации «Общие критерии…» изданы в качестве ГОСТа:

ГОСТ Р ИСО/МЭК 15408-2001 «Методы и средства обеспечения безопасности»,

ГОСТ Р ИСО/МЭК 15408-2002 «Критерии оценки безопасности информационных технологий», ввод в действие с 1 января 2004 года).

В дальнейшем Россия присоединилась к соглашению «О признании сертификатов».

В «Общих критериях…» предложена система функциональных требований к защищенным КС и критерии их независимого ранжирования.

Иначе говоря, в этих стандартах не устанавливается линейная шкала уровней безопасности КС, характерная для «Оранжевой книги». Это объясняется тем, что для одних КС наиболее важным требованием является идентификация и аутентификация пользователей, а для других – реализация конкретной политики разграничения доступа к ресурсам или обеспечение доступности информации.

Основные понятия и идеи «Общих критериев». Основным свойством, которым должны обладать действительно общие критерии оценки безопасности информационных технологий, является универсальность. Следовательно, они не должны содержать априорных предположений об объекте оценки.

В ОК данное условие выполнено: под объектом оценки (ОО) понимается аппаратно-программный продукт или информационная система с соответствующей документацией.

Система – это специфическое воплощение информационных технологий с конкретным назначением и условиями эксплуатации.

Продукт, согласно ОК, есть совокупность средств ИТ, предоставляющая определенные функциональные возможности и предназначенная для непосредственного использования или включения в различные системы.

В качестве собирательного термина для систем и продуктов применяют словосочетание «изделие ИТ». Оно может быть как уже существующим, так и проектируемым.

Объект оценки рассматривается в определенном контексте – среде безопасности, в которую включаются все, что имеет отношение к его безопасности, а именно:

законодательная среда – законы и нормативные акты, затрагивающие ОО;

административная среда – положения политик и программ безопасности, учитывающие особенности ОО;

процедурная среда – физическая среда ОО и меры физической защиты, персонал и его свойства (знания, опыт и т.п.), принятые эксплуатационные и иные процедуры;

программно-техническая среда – предназначение объекта оценки и предполагаемые области его применения, активы (ресурсы), которые требуют защиты средствами ОО.

Дальнейший этап технологического цикла подготовки к оценке, согласно «Общим критериям», - описание следующих аспектов среды ОО:

предположения безопасности. Они выделяют объект оценки из общего контекста, задают границы рассмотрения. Истинность этих предположений принимается без доказательств, а из множества возможных отбирается только то, что заведомо необходимо для обеспечения безопасности ОО;

угрозы безопасности ОО, наличие которых в рассматриваемой среде установлено или предполагается. Они характеризуются несколькими параметрами: источник, метод воздействия, опасные с точки зрения злонамеренного использования уязвимости, ресурсы (активы), потенциально подверженные повреждению. При анализе рисков принимаются во внимание вероятность активизации угрозы и ее успешного осуществления, а также размер возможного ущерба. По результатам анализа из множества допустимых угроз отбираются только те, ущерб от которых нуждается в уменьшении;

положения политики безопасности, предназначенные для применения к объекту оценки. Для системы ИТ такие положения могут быть описаны точно, для продукта – в общих чертах.

На основании предположений, при учете угроз и положений политики безопасности формулируются цели безопасности для объекта оценки, направленные на обеспечение противостояния угрозам и выполнение политики безопасности.

В зависимости от непосредственного отношения к ОО или к среде они подразделяются на две группы.

Часть целей для среды может достигаться нетехническими (процедурными) мерами.

Все остальные (для объекта и среды) носят программно-технический характер. Для их достижения к объекту и среде предъявляются требования безопасности.

«Общие критерии» в главной своей части как раз и являются каталогом (библиотекой) требований безопасности. Спектр стандартизованных требований чрезвычайно широк – это необходимое условие универсальности ОК.

Высокий уровень детализации делает их конкретными, допускающими однозначную проверку.

Наличие параметров обуславливает гибкость требований, а дополнительную возможность ее достижения привносит использование нестандартных (не входящих в каталог ОК) требований.

В России разработаны следующие государственные стандарты (ГОСТы) в области защиты информации.

ГОСТ Р ИСО/МЭК 15408 - 1 – 2002. Информационная технология. Методы и средства обеспечения безопасности. Критерии оценки безопасности информационных технологий. Часть 1. Введение и общая модель.

ГОСТ Р ИСО/МЭК 15408 - 2 – 2002. Информационная технология. Методы и средства обеспечения безопасности. Критерии оценки безопасности информационных технологий. Часть 2. Функциональные требования безопасности.

ГОСТ Р ИСО/МЭК 15408 - 2 – 2002. Информационная технология. Методы и средства обеспечения безопасности. Критерии оценки безопасности информационных технологий. Часть 3. Требования доверия к безопасности.

Гостехкомиссия России. Руководящий документ. Концепция защиты СВТ и АС от НСД к информации.

Гостехкомиссия России. Руководящий документ. Защита от несанкционированного доступа к информации. Термины и определения.

Гостехкомиссия России. Руководящий документ. Автоматизированные системы. Защита от несанкционированного доступа к информации. Классификация автоматизированных систем и требования по защите информации.

Гостехкомиссия России. Руководящий документ. Временное положение по организации разработки, изготовления и эксплуатации программных и технических средств защиты информации от НСД в автоматизированных системах и средствах вычислительной техники.

Гостехкомиссия России. Руководящий документ. Средства вычислительной техники. Защита от несанкционированного доступа к информации. Показатели защищенности от несанкционированного доступа к информации.

Гостехкомиссия России. Руководящий документ. Средства вычислительной техники. Межсетевые экраны. Защита от несанкционированного доступа к информации. Показатели защищенности от несанкционированного доступа к информации.

Гостехкомиссия России. Руководящий документ. Безопасность информационных технологий. Критерии оценки безопасности информационных технологий.

Гостехкомиссия России. Руководство по разработке профилей защиты и заданий по безопасности (проект).

Гостехкомиссия России. Руководящий документ. Руководство по регистрации профилей защиты (проект).

3. КРИПТОГРАФИЧЕСКИЕ МЕТОДЫ И СРЕДСТВА ЗАЩИТЫ ДАННЫХ

3.1. Общие определения

Криптография (от греческих слов kryptos - тайный и grapho - пишу) – наука о методах изменения сообщений с целью сделать их непонятными для непосвященных. Метод

криптографии можно определить как некоторое множество отображений одного пространства (пространства возможных сообщений) в другое пространство (пространство возможных криптограмм). Каждое конкретное отображение из этого множества соответствует шифрованию при помощи конкретного ключа.

Исходное сообщение (открытый текст) - сообщение, текст которого необходимо сделать непонятным для посторонних.

Шифрование данных - процесс преобразования открытых данных в зашифрованные данные (шифротекст, криптограмму) при помощи шифра.

Шифр - совокупность обратимых преобразований множества возможных открытых данных во множество возможных шифротекстов, осуществляемых по определенным правилам с применением ключей.

Ключ - конкретное секретное состояние некоторого параметра (параметров), обеспечивающее выбор одного преобразования из совокупности возможных для используемого метода шифрования.

Криптоанализ - наука о раскрытии исходного текста зашифрованного сообщения без доступа к ключу. Успешный анализ может раскрыть исходный текст или ключ.

В 1949 году была опубликована статья Клода Шеннона «Теория связи в секретных системах», которая определила научную базу криптографии и криптоанализа. С этого времени стали говорить о новой науке – криптологии.

Криптология (от греческого kryptos - тайный и logos - сообщение) - наука о преобразовании информации для обеспечения ее секретности.

Криптографические методы защиты информации могут применяться для защиты информации, обрабатываемой ЭВМ и для закрытия информации, передаваемой по линиям связи.

Поэтому криптографические методы могут использоваться как внутри отдельных устройств или звеньев системы, так и на различных участках линий связи.

Известны различные подходы к классификации методов криптографического преобразования информации. По виду воздействия на исходную информацию методы криптографического преобразования информации могут быть разделены на четыре группы (рис. 11).

Рис. 11. Классификация методов криптографического преобразования информации

1. **Шифрование**. Заключается в проведении обратимых математических, логических, комбинаторных и других преобразований исходной информации, в результате которых зашифрованная информация представляет собой хаотический набор букв, цифр, других символов и двоичных кодов. Для шифрования информации используют алгоритм преобразования и ключ. Как правило, алгоритм для определенного метода шифрования является неизменным. Ключ шифрования может изменяться. Существует следующая классификация методов шифрования:

- замена (подстановка);
- перестановка;
- аналитическое преобразование;
- гаммирование;
- комбинированное преобразование.

2. **Стеганография**. Методы стеганографии позволяют скрыть не только смысл хранящейся или передаваемой информации, но и сам факт хранения или передачи закрытой информации. В компьютерных сетях практическое использование стеганографии только начинается. В основе всех методов стеганографии лежит маскирование закрытой информации среди открытых файлов.

Существует несколько методов скрытой передачи информации. Например:

- представление графической и звуковой информации в числовом виде. Так в графических объектах наименьший элемент изображения может кодироваться одним байтом;

- помещение битов скрытого файла в младшие разряды определенных байтов изображения в соответствии с алгоритмом криптографического преобразования.

Очень сложно выявить скрытую информацию с помощью специальных программ.

Наилучшим образом для внедрения скрытой информации подходят изображения местности, снимки со спутников, самолетов и т.п.

С помощью средств стеганографии могут маскироваться текст, изображение, речь, цифровая подпись, зашифрованное сообщение. Комплексное использование стеганографии и шифрования многократно повышает сложность решения задачи обнаружения и раскрытия секретной информации.

3. **Кодирование**. При кодировании информации происходит замена смысловых конструкций исходной информации (слов, предложений) кодами. В качестве кодов могут использоваться сочетания букв, цифр, букв и цифр. При кодировании и обратном преобразовании используют специальные таблицы или словари. Кодирование информации целесообразно применять в системах с ограниченным набором смысловых конструкций. Такой вид криптографического преобразования применим, например, в командных линиях АС. Недостатками кодирования конфиденциальной информации является необхо-

димость хранения и распространения кодировочных таблиц, которые необходимо часто менять, чтобы избежать раскрытия кодов статистическими методами обработки перехваченных сообщений.

4. **Сжатие**. Сжатие информации может быть отнесено к методам криптографического преобразования информации с определенными оговорками. Целью сжатия является сокращение объема информации. В то же время сжатая информация не может быть прочитана или использована без обратного преобразования. Учитывая доступность средств сжатия и обратного преобразования, эти методы нельзя рассматривать как надежные средства криптографического преобразования информации. Поэтому сжатые файлы конфиденциальной информации подвергаются последующему шифрованию. Для сокращения времени целесообразно совмещать процесс сжатия и шифрования информации.

В настоящее время разработано большое количество различных методов шифрования, созданы теоретические и практические основы их применения. Подавляющее число этих методов может быть успешно использовано для закрытия информации в АС.

Процесс криптографического закрытия данных может осуществляться как программно, так и аппаратно. Однако аппаратная реализация обладает рядом преимуществ, главным из которых является высокая производительность.

3.2. Общие сведения о криптографических системах

Обобщенная схема криптографической системы, обеспечивающей шифрование передаваемой информации, показана на рис. 12.

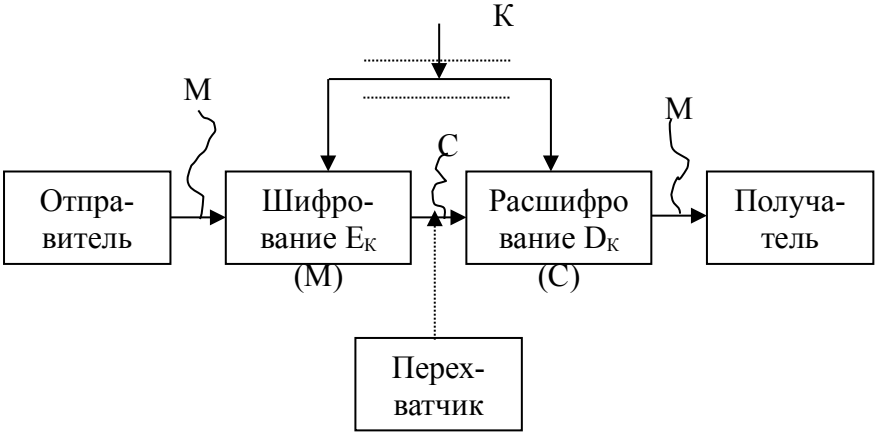


Рис. 12. Обобщенная схема криптосистемы

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

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

$$D_K (C) = E_K^{-1}(E_K (M)) = M$$

Преобразование E_K выбирается из семейства криптографических преобразований, называемых криптоалгоритмами. Параметр, с помощью которого выбирается отдельное используемое преобразование, называется криптографическим ключом К. Криптосистема имеет разные варианты реализации: набор инструкций, аппаратные средства, комплекс

программ компьютера, которые позволяют зашифровать открытый текст и расшифровать шифртекст различными способами, один из которых выбирается с помощью конкретного ключа К.

Таким образом, криптографическая система – это однопараметрическое семейство $(E_K)_{K\in\overline{K}}$ обратимых преобразований $E_K : \overline{M} \to \overline{C}$ из пространства $\overline{M}$ сообщений открытого текста в пространство $\overline{C}$ шифрованных текстов. Параметр К (ключ) выбирается из конечного множества $\overline{K}$, называемого пространством ключей.

Преобразование шифрования может быть симметричным или асимметричным относительно преобразования расшифрования. Это важное свойство функции преобразования определяет два класса криптосистем: симметричные (одноключевые) криптосистемы; асимметричные (двухключевые) криптосистемы (с открытым ключом).

Схема симметричной криптосистемы с одним секретным ключом была показана на рис. 12. В ней используются одинаковые секретные ключи в блоке шифрования и блоке расшифрования.

Обобщенная схема асимметричной криптосистемы с двумя разными ключами K_1 и K_2 показана на рис. 13. В этой криптосистеме один ключей является открытым а другой - секретным.

В симметричной криптосистеме секретный ключ надо передавать отправителю и получателю по защищенному каналу распространения ключей, например, такому, как курьерская служба. На рис. 14 этот канал показан «экранированной» линией. Существуют и другие способы распределения секретных ключей. В асимметричной криптосистеме передают по незащищенному каналу только открытый ключ, а секретный ключ сохраняют на месте его генерации.

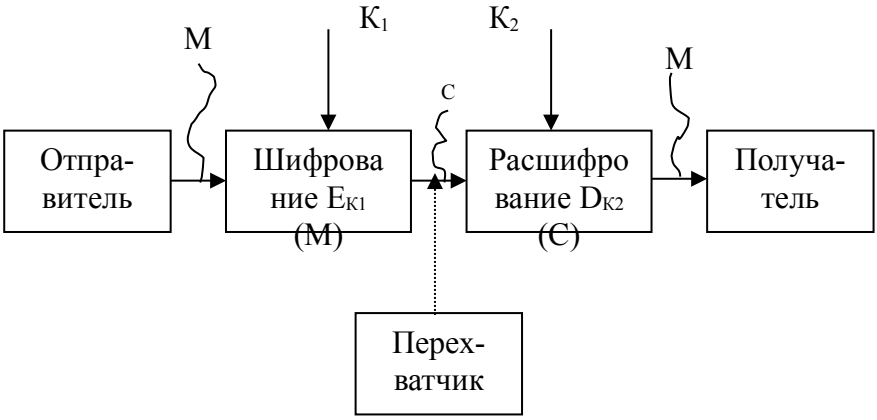


Рис.13. Обобщенная схема асимметричной криптосистемы с открытым ключом

На рис. 14 показан поток информации в криптосистеме в случае активных действий перехватчика. Активный перехватчик не только считывает все шифротексты, передаваемые по каналу, но может также пытаться изменять их по своему усмотрению.

Любая попытка со стороны перехватчика расшифровать шифротекст C для получения открытого текста M или зашифровать свой собственный текст M^1 для получения правдоподобного шифротекста C^1, не имея подлинного ключа, называется криптоаналитической атакой.

Если предпринятые криптоаналитические атаки не достигают поставленной цели и криптоаналитик не может, не имея подлинного ключа, вывести M из C или C^1 из M^1, то полагают, что такая криптосистема является криптостойкой.

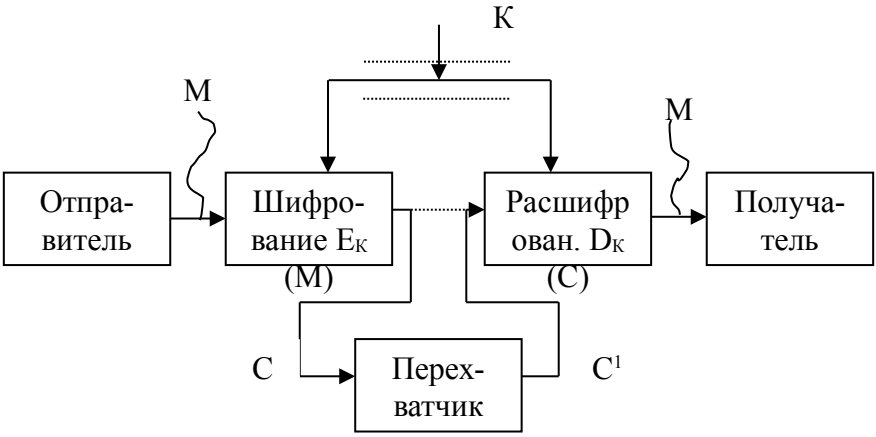


Рис. 14. Поток информации в криптосистеме при активном перехвате сообщений

Фундаментальное правило построения криптосистем, впервые сформулированное голландцем А. Керкхоффом еще в XIX веке, заключается в том, что стойкость шифра (криптосистемы) должна определяться только секретностью ключа. Иными словами, правило Керкхоффа состоит в том, что весь алгоритм шифрования, кроме значения секретного ключа, известен криптоаналитику противника. Это обусловлено тем, что криптосистема, реализующая семейство криптографических преобразований, обычно рассматривается как открытая система. Такой подход отражает очень важный принцип технологии защиты информации: защищенность системы не должна зависеть от секретности чего-либо такого, что невозможно быстро изменить в случае утечки секретной информации. Обычно криптосистема представляет собой совокупность аппаратных и программных средств, которую можно изменить только при значительных затратах времени и средств, тогда как ключ является легко изменяемым объектом. Именно

поэтому стойкость криптосистемы определяется только секретностью ключа.

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

Таким образом**, основные требования**, предъявляемые к **методам защитного преобразования информации** следующие:

1) применяемый метод должен быть достаточно устойчивым к попыткам раскрыть исходный текст, имея только зашифрованный текст;

2) объем ключа не должен затруднять его запоминание и пересылку;

3) алгоритм преобразования информации и ключ, используемые для шифрования и дешифрования, не должны быть очень сложными: затраты на защитные преобразования должны быть приемлемы при заданном уровне сохранности информации;

4) ошибки в шифровании не должны вызывать потерю информации. Из-за появления ошибок передачи шифрованного сообщения по каналам связи не должна исключаться возможность надежной расшифровки текста на приемном конце;

5) длина зашифрованного текста не должна превышать длину исходного текста;

6) необходимые временные и стоимостные ресурсы на шифрование и дешифрование информации определяются требуемой степенью защиты информации.

3.3. Методы шифрования

Методы замены. Шифрование методом замены (подстановки) основано на алгебраической операции, называемой подстановкой.

Подстановкой называется взаимно однозначное 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}.

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

Далее в примерах, где необходимо, будет использовано кодирование букв русского алфавита, приведенное в табл. 3.1. Знак "_" означает пробел.

Таблица 3.1

Кодирование букв русского алфавита

Буквы	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 _
Коды	01020304050607080910111213141516171819202122232425262728 2930313233

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

Пример 1. Открытый текст: "ШИФРОВАНИЕ_ЗАМЕНОЙ".

Подстановка задана табл. 3.2.

Таблица 3.2

Алфавиты исходного и шифротекста

Алфавит исходного текста	А Б В Г Д ...
Алфавит шифротекста	_ Я Ю Э Ь

Шифротекст: "ИШМРТЮ_УШЫАЩ_ФЫУТЧ".

Основным недостатком рассмотренного метода является то, что статистические свойства открытого текста (частоты повторения букв) сохраняются в шифротексте.

Общая формула моноалфавитной замены имеет следующий вид:

$$Yi=k1*Xi+k2(mod\ N)$$

где yi - i-й символ алфавита;

k1 и k2 - константы;

Xi - i-й символ открытого текста (номер буквы в алфавите);

N - длина используемого алфавита.

Шифр, задаваемый формулой:

$$yi=xi+ki(mod\ N),$$

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

Пример 2. Открытый текст: "ЗАМЕНА".

Ключ: "КЛЮЧ" (табл. 3.3).

Таблица 3.3

Шифрование с помощью ключа «Ключ»

З А М Е Н А
К Л Ю Ч К Л

y1=8+11(mod 33)=19 -> Т

y2=1+12(mod 33)=13 -> М

y3=13+31(mod 33)=11-> К

y4=6+24(mod 33)=30 -> Э

y5=14+11(mod 33)=25 -> Ш

y6=1+12(mod 33)=13 -> М.

Шифротекст: "ТМКЭШМ".

Шифры Бофора используют формулы:

$$yi=ki-xi(mod\ n)\quad и$$

$$yi=xi-ki(mod\ n).$$

Гомофоническая замена одному символу открытого текста ставит в соответствие несколько символов шифро-

текста. Этот метод применяется для искажения статистических свойств шифротекста.

Пример 3. Открытый текст: "ЗАМЕНА". Подстановка задана в табл. 3.4.

Таблица 3.4

Алфавиты открытого и шифротекста при гомофонической замене

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

Шифротекст: "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 шифротекста. Пусть используется k алфавитов. Тогда открытый текст:

$X=X_1X_2...X_k\ X_{k+1}...X_{2k}\ X_{2k+1}...$

заменяется шифртекстом:

$Y=F1(X1)F2(X2)...Fk(Xk)\ F1(Xk+1)...Fk(X2k)\ F1(X2k+1)...$

где Fi(Xj) означает символ шифртекста алфавита i для символа открытого текста Xj.

Пример 4. Открытый текст: "ЗАМЕНА", k=3.
Подстановка задана таблицей из примера 3.
Шифртекст: "76 31 61 97 84 48".

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:

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

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

3) 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;

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

Пример 5. Открытый текст: "ШИФР_ПЛЭЙФЕРА". Матрица алфавита представлена в табл. 3.5.

Таблица 3.5

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

Шифртекст: "РДЫИ,-СТ-И.ХЧС"

При рассмотрении этих видов шифров становится очевидным, что чем больше длина ключа (например, в шифре Виженера), 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 продолжает-

ся с помощью открытого текста или криптограммы, смещенной на длину первичного ключа.

Пример 6. Открытый текст: "ШИФРОВАНИЕ_ЗАМЕНОЙ".

Первичный ключ: "КЛЮЧ"

Схема шифрования с автоключом при использовании открытого текста представлена в табл. 3.6.

Таблица 3.6

Шифрование с автоключом при использовании открытого текста

Ш	И	Ф	Р	О	В	А	Н	И	Е	_	З	А	М	Е	Н	О	Й
К	Л	Ю	Ч	Ш	И	Ф	Р	О	В	А	Н	И	Е	_	З	А	М
36	21	52	41	40	12	22	31	24	09	34	22	10	19	39	22	16	23
В	Ф	Т	З	Ж	Л	Х	Ю	Ч	И	А	Х	Й	Т	Е	Х	П	Ц

Схема шифрования с автоключом при использовании криптограммы представлена в табл. 3.7.

Таблица 3.7

Шифрования с автоключом при использовании криптограммы

Ш	И	Ф	Р	О	В	А	Н	И	Е	_	З	А	М	Е	Н	О	Й
К	Л	Ю	Ч	В	Ф	Т	З	С	Ч	У	Х	Ъ	Э	У	Э	Ы	Й
36	21	52	41	18	24	20	22	27	30	53	30	24	43	26	44	39	20
В	Ф	Т	З	С	Ч	У	Х	Ъ	Э	У	Э	Ы	Й	Щ	К	Й	У

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

Пример 7. Открытый текст: "ШИФРОВАНИЕ_ПЕРЕСТАНОВКОИ". Ключ (правило перестановки): группы из 8 букв с порядковыми номерами 1.2.....8 переставить в порядок 3-8-1-5-2-7-6-4.

Шифротекст: "ФНШОИАВР_СИЕЕЕРПННТВАОКО".

Можно использовать и усложненную перестановку. Для этого открытый текст записывается в матрицу по опреде-

ленному ключу k1. Шифртекст образуется при считывании из этой матрицы по ключу k2.

Пример 8. Открытый текст:
"ШИФРОВАНИЕ_ПЕРЕСТАНОВКОЙ".

Матрица из четырех столбцов:
Ключи: k1 3-4-2-5-1-6; k2 4-2-3-1.
Исходная матрица

1	Ш	и	Ф	Р
2	О	в	А	Н
3	И	е		П
4	Е	р	Е	С
5	Т	а	Н	О
6	В	к	О	Й

Запись по строкам в соответствии с ключом k1.

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
	1	2	3	4

Чтение по столбцам в соответствии с ключом k2.

Шифротекст: "ПСНОРЙЕРВАИК_ЕАНФОИЕОТШВ".

Методы аналитических преобразований. Шифрование методами аналитических преобразований основано на понятии односторонней функции. Функция y=f(x) является односторонней, если она за сравнительно небольшое число операций преобразует элемент открытого текста x в элемент шифротекста y для всех значений x из области определения, а обратная операция (вычисление $x=F^{-1}(y)$ при известном шифротексте) является вычислительно трудоемкой.

В качестве односторонней функции можно использовать следующие преобразования:

1) умножение матриц;

2) решение задачи об укладке ранца;

3) вычисление значения полинома по модулю;

4) экспоненциальные преобразования и другие.

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 9. Открытый текст: "ПРИКАЗ" ("16 17 09 11 01 08").

Матрица C: $C = \begin{vmatrix} 1 & 3 & 2 \\ 2 & 1 & 5 \\ 3 & 2 & 1 \end{vmatrix}$

$$\begin{vmatrix} 1 & 3 & 2 \\ 2 & 1 & 5 \\ 3 & 2 & 1 \end{vmatrix} \times \begin{vmatrix} 16 \\ 17 \\ 09 \end{vmatrix} = | 85\ 94\ 91 |$$

$$\begin{vmatrix} 1 & 3 & 2 \\ 2 & 1 & 5 \\ 3 & 2 & 1 \end{vmatrix} \times \begin{vmatrix} 11 \\ 01 \\ 08 \end{vmatrix} = | 30\ 63\ 43 |$$

Шифротекст: "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 $s_i=|x_1,x_2,...,x_n|$, $x_k \in \{0,1\}$.

Шифротекст получается как скалярное произведение $C \cdot s_i$.

Пример 10. Открытый текст: "ПРИКАЗ" ("16 17 09 11 01 08").

Вектор C={1,3,5,7,11}.

Запишем код каждой буквы открытого текста в двоичном виде, используя пять разрядов.

П	Р	И	К	А	З
10000	10001	01001	01011	00001	01000

Произведем соответствующие операции:

$y1=1 \cdot 1=1$

$y2=1 \cdot 1+1 \cdot 11=12$

$y3=1 \cdot 3+1 \cdot 11=14$

$y4=1 \cdot 3+1 \cdot 7+1 \cdot 11=21$

$y5=1 \cdot 11=11$

$y6=1 \cdot 3=3$.

Шифротекст: "01 12 14 21 11 03".

Метод полиномов основан на преобразовании

$$y_i = x_i^n+a_1 \cdot x_i^{n-1}+...+a_n \cdot x_i \,(\mathrm{mod}\ p).$$

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 11. Открытый текст: "ПРИКАЗ" ("16 17 09 11 01 08").

Полином: $y_i=x_i^3+2 \cdot x_i^2+3 \cdot x_i+4(\mathrm{mod}\ 991)$.

$y1=16^3+2 \cdot 16^2+3 \cdot 16+4(\mathrm{mod}\ 991)=696$

$y2=17^3+2 \cdot 17^2+3 \cdot 16+4(\mathrm{mod}\ 991)=591$

$y3=9^3+2 \cdot 9^2+3 \cdot 9+4(\mathrm{mod}\ 991)=922$

$y4=11^3+2 \cdot 11^2+3 \cdot 11+4(\mathrm{mod}\ 991)=619$

$y5=1^3+2 \cdot 1^2+3 \cdot 1+4(\mathrm{mod}\ 991)=10$

$y6=8^3+2 \cdot 8^2+3 \cdot 8+4(\mathrm{mod}\ 991)=668$.

Шифротекст: "696 591 922 619 010 668".

Экспоненциальный шифр использует преобразование вида

$$y_i = a^{(x_i)} \pmod p,$$

где x_i - целое, $1 \le x_i \le p-1$;

p - большое простое число;

a - целое, $1 \le a \le p$.

Пример 12. Открытый текст: "ПРИКАЗ" ("16 17 09 11 01 08"); a=3; p=991.

$y_1=3^{16} \pmod{991}=43046721 \pmod{991}=654$

$y_2=3^{17} \pmod{991}=129140163 \pmod{991}=971$

$y_3=3^9 \pmod{991} = 19683 \pmod{991} =854$

$y_4=3^{11} \pmod{991}=177147 \pmod{991}=749$

$y_5=3^1 \pmod{991}=3$

$y_6=3^8 \pmod{991}=6561 \pmod{991}=615.$

Шифротекст: "654 971 854 749 003 615".

Методы гаммирования. Особым случаем метода аналитических преобразований является метод, основанный на преобразовании

$$y_i = x_i \mathbin{++} h_i$$

где y_i - i-й символ шифротекста;

x_i - i-й символ открытого текста;

h_i - i-й символ гаммы;

++ - выполняемая операция (наложение гаммы).

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.

Пример 13. Открытый текст: "ПРИКАЗ" ("16 17 09 11 01 08").

Гамма: "ГАММА" ("04 01 13 13 01").

Операция: сложение по mod 33.

y_1= 16+4(mod 33)=20
y_2= 17+1(mod 33)=18
y_3= 9+13(mod 33)=22
y_4= 11+13(mod 33)=24
y_5= 1+1(mod 33)=2
y_6= 8+4(mod 33)=12.

Шифротекст: "УСХЧБЛ" ("20 18 22 24 02 12").

Пример 14. Открытый текст: "ПРИКАЗ" ("16 17 09 11 01 08").

Первые значения датчика: "21794567".

Операция: сложение по mod 2.

Запишем код каждой буквы открытого текста в двоичном виде, используя пять разрядов, а каждую цифру гаммы - используя четыре разряда:

10000 10001 01001 01011 00001 01000
++
00100 00101 11100 10100 01010 11001

10100 10100 10101 11111 01011 10001.

Шифротекст: "УУФЮКР".

Комбинированные методы. Наиболее часто применяются такие комбинации, как подстановка и гамма, перестановка и гамма, подстановка и перестановка, гамма и гамма.

Примером может служить шифр Френдберга, который комбинирует многоалфавитную подстановку с генератором псевдослучайных чисел, суть алгоритма поясняется следующей схемой:

1) установление начального состояния генератора псевдослучайных чисел;

2) установление начального списка подстановки;
3) все символы открытого текста зашифрованы?
4) если да - конец работы, если нет -продолжить;
5) осуществление замены;
6) генерация случайного числа;
7) перестановка местами знаков в списке замены;
8) переход на шаг 4.

Особенность данного алгоритма состоит в том, что при большом объеме шифротекста частотные характеристики символов шифротекста близки к равномерному распределению независимо от содержания открытого текста.

Пример 15. Открытый текст: "АБРАКАДАБРА".
Используем следующую замену:

А Б Д К Р
X V N R S

Последовательность чисел, вырабатываемая датчиком: 31412543125.

1. y_1=X. После перестановки символов исходного алфавита получаем таблицу(h1=3).

Д Б А К Р
X V N R S

2. y_2=V. Таблица замены после перестановки (h2=1) принимает вид:

Б Д А К Р
X V N R S

Осуществляя дальнейшие преобразования в соответствии с алгоритмом Френдберга, получаем шифротекст: "XVSNSXXSSSN".

Одной из разновидностей метода гаммирования является наиболее часто применяемый метод многократного наложения гамм.

При составлении комбинированных шифров необходимо проявлять осторожность, так как неправильный выбор составлявших шифров может привести к исходному открытому тексту. Простейшим примером служит наложение одной гаммы дважды.

4. СОВРЕМЕННЫЕ СИММЕТРИЧНЫЕ И АСИММЕТРИЧНЫЕ КРИПТОСИСТЕМЫ

По мнению К. Шеннона, в практических шифрах необходимо использовать два общих принципа: рассеивание и перемешивание.

Рассеивание представляет собой распространение влияния одного знака открытого текста на много знаков шифротекста, что позволяет скрыть статистические свойства открытого текста.

Перемешивание предполагает использование таких шифрующих преобразований, которые усложняют восстановление взаимосвязи статистических свойств открытого и шифрованного текстов. Однако шифр должен не только затруднять раскрытие, но и обеспечивать легкость зашифрования и расшифрования при известном пользователю секретном ключе.

Распространенным способом достижения эффектов рассеивания и перемешивания является использование составного шифра, т.е. такого шифра, который может быть реализован в виде некоторой последовательности простых шифров, каждый из которых вносит свой вклад в суммарное рассеивание и перемешивание.

В составных шифрах в качестве простых шифров чаще всего используются простые перестановки и подстановки. При **перестановке** просто перемешивают символы открытого текста, причем конкретный вид перемешивания определяется секретным ключом. При **подстановке** каждый символ открытого текста заменяют другим символом из того же алфавита, а кон-

кретный вид подстановки также определяется секретным ключом. Следует заметить, что в современном блочном шифре блоки открытого текста и шифротекста представляют собой двоичные последовательности обычно длиной 64 бита. В принципе каждый блок может принимать 2^{64} значений. Поэтому подстановки выполняются в очень большом алфавите, содержащем до $2^{64} \sim 10^{19}$ "символов".

При многократном чередовании простых перестановок и подстановок, управляемых достаточно длинным секретным ключом, можно получить очень стойкий шифр с хорошим рассеиванием и перемешиванием. Рассмотренные ниже криптоалгоритмы DES и отечественный стандарт шифрования данных построены в полном соответствии с указанной методологией.

4.1. Стандарт шифрования данных (DES)

Одним из наиболее распространенных криптографических стандартов на шифрование данных, применяемых в США, является DES (Data Encrtyption Standart). Стандарт шифрования DES был опубликован в 1977 году. 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 дешифрация данных, защищенных с помощью DES, не могла быть выполнена статистическими (например, с помощью частотного словаря) или математическими ("прокручивание" 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 ЭВМ (исключая некоторые данные,

методы защиты которых определяются специальными актами). Его применяют многие негосударственные 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. Заметим, что, одной стороны, публикация алгоритма нежелательна, поскольку может привести к дешифрации закрытой информации. Но, с другой стороны, это не столь существенно (если стандарт сильный) по сравнению со слабыми методами защиты данных, используемыми государственными институтами. Иначе говоря, потери от публикации алгоритма криптографической защиты намного меньше, чем потеря от применения методов защиты с низкой криптостойкостью. Разумеется, стандартный алгоритм шифрования данных должен обладать такими характеристиками, чтобы его опубликование не сказалось на криптостойкости.

К настоящему времени DES является наиболее распространенным алгоритмом, используемым в системах защиты коммерческой информации. Более того, реализация алгоритма DES в таких системах становится признаком хорошего тона. Основные достоинства алгоритма DES:

используется только один ключ длиной 56 бит;

зашифровав сообщение с помощью одного пакета программ, для расшифровки можно использовать любой другой пакет программ, соответствующий стандарту DES;

относительная простота алгоритма обеспечивает высокую скорость обработки;

достаточно высокая стойкость алгоритма.

Алгоритм DES также использует комбинацию подстановок и перестановок. DES осуществляет шифрование 64-битовых блоков данных с помощью 64-битового ключа, в котором значащими являются 56 бит (остальные 8 бит - проверочные биты для контроля на четность). Дешифрование в DES являет-

ся операцией, обратной шифрованию, и выполняется путем повторения операций шифрования в обратной последовательности. Обобщенная схема процесса шифрования в алгоритме DES показана на рис. 15.

Следует сразу отметить, что все приводимые таблицы являются стандартными и должны включаться в реализацию алгоритма DES в неизменном виде.

Все перестановки и коды в таблицах подобраны разработчиками таким образом, чтобы максимально затруднить процесс расшифровки путем подбора ключа. При описании алгоритма DES применены следующие обозначения:

L и R - последовательности битов (левая (left) и правая (right));

LR - конкатенация последовательностей L и R, т.е. такая последовательность битов, длина которой равна сумме длин L R; в последовательности LR биты последовательности R следуют за битами последовательности L;

+ - операция побитового сложения по модулю 2.

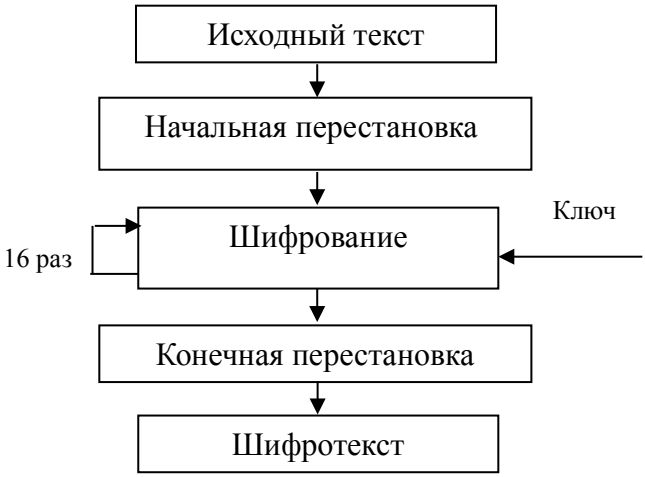


Рис.15. Обобщенная схема шифрования в алгоритме DES

Подробнее процесс шифрования данных поясняет рис. 16.

1. Первоначально каждые 64 бита входной последовательности перестанавливаются в соответствии с табл. 4.1. Таким образом, бит 58 входной последовательности становится битом 1; бит 59 - битом 2 и т.д.

Таблица 4.1

Матрица начальной перестановки

58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

2. Полученная последовательность битов T_0 разделяется на две последовательности: L_0 - левые или старшие биты, R_0- правые или младшие биты, каждая из которых содержит 32 бита.

Затем выполняется итеративный процесс шифрования, состоящий из 16 шагов (циклов). Пусть Ti - результат i-й итерации:

$Ti = L_iR_i$,

где $Li = t_1\ t_2\ ...\ t_{32}$ (первые 32 бита); $Rj = t_{33}1_{34}...\ t_{64}$ (последние 32 бита). Тогда результат i-й итерации описывается следующими формулами:

$Li = R_{i-1}$, $i = 1,2, ..., 16$;

$Ri = L_{i-1}+f(R_{i-1},Ki)$, $i= 1,2, ..., 16$.

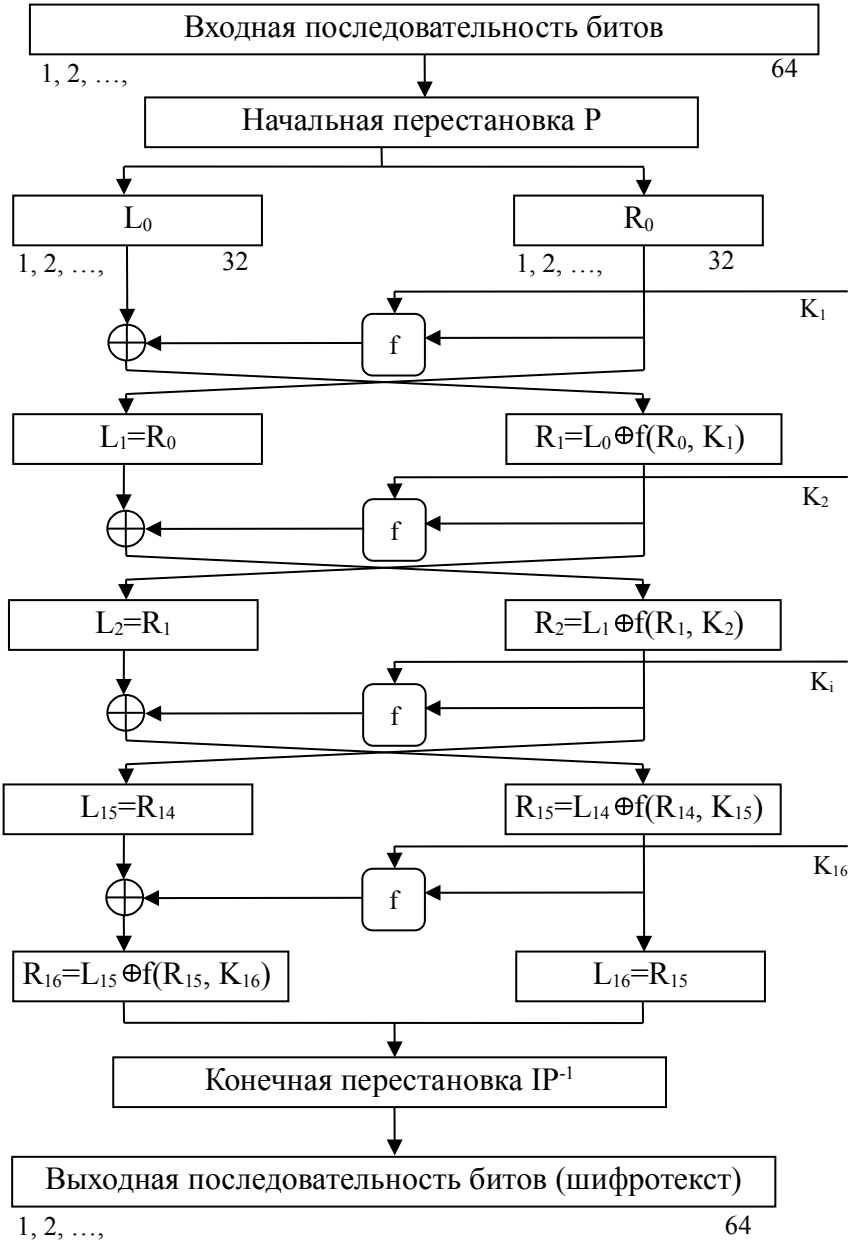

Входная последовательность битов
1, 2, …,
64
Начальная перестановка P
L0
R0
1, 2, …,
32
1, 2, …,
32
K1
f
L1=R0
R1=L0⊕f(R0, K1)
K2
f
L2=R1
R2=L1⊕f(R1, K2)
Ki
f
L15=R14
R15=L14⊕f(R14, K15)
K16
f
R16=L15⊕f(R15, K16)
L16=R15
Конечная перестановка IP-1
Выходная последовательность битов (шифротекст)
1, 2, …,
64

Рис. 16. Структура алгоритма DES

Функция f называется функцией шифрования. Ее аргументами являются последовательность R_{i-1}, получаемая на предыдущем шаге итерации, и 48-битовый ключ K_i, который является результатом преобразования 64-битового ключа шифра К. (Подробнее функция шифрования f и алгоритм получения ключа K_i, описаны ниже.)

На последнем шаге итерации получают последовательности R_{16} и L_{16} (без перестановки местами), которые конкатенируются в 64-битовую последовательность R_{16} L_{16}.

3. По окончании шифрования осуществляется восстановление позиций битов с помощью матрицы обратной перестановки IP^{-1} (табл.4.2).

Таблица 4.2

Матрица обратной перестановки

40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25

Процесс расшифровки данных является инверсным по отношению к процессу шифрования. Все действия должны быть выполнены в обратном порядке. Это означает, что расшифровываемые данные сначала переставляются в соответствии с матрицей обратной перестановки, а затем над последовательностью битов $R_{16}L_{16}$ выполняются те же действия, что и в процессе шифрования, но в обратном порядке.

Итеративный процесс расшифровки может быть описан следующими формулами:

$R_{i-1} = L_i, \quad i = 1,2,\ldots,16;$

$L_{i-1} = R_i + f(L_i, K_i)$, $i = 1, 2, \ldots,$ 16.

Таким образом, для процесса расшифровки переставленным входным блоком $R_{16}L_{16}$ на первой итерации используется ключ K_{16}, на второй итерации – K_{15} и т.д. На 16-й итерации используется ключ K_1. На последнем шаге итерации будут получены последовательности L_0 и R_0, которые конкатенируются в 64-битовую последовательность L_0R_0. Затем в этой последовательности 64 бита переставляются в соответствии с матрицей IP. Результат такого преобразования - исходная последовательность битов (расшифрованное 64-битовое значение).

Рассмотрим, что скрывается под преобразованием, обозначенным буквой f. Схема вычисления функции шифрования $f(R_{i-1}, K_i)$ показана на рис. 17.

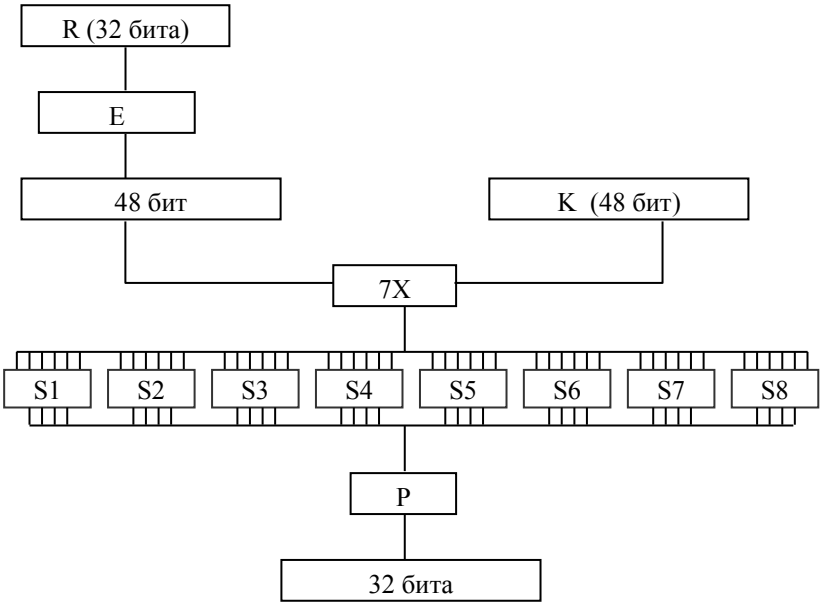


Рис.17. Схема вычисления функции шифрования f

Для вычисления значения функции f используются:

функция E (расширение 32 бит до 48);

функции S1, S2, ..., S8 (преобразование 6-битового числа в 4-битовое);

функция P (перестановка битов в 32-битовой последовательности).

Рассмотрим определения этих функций.

Аргументами функции шифрования f являются R_{i-1} (32 бита) и K_i (48 бит).

Результат функции E (R_{i-1}) есть 48-битовое число. Функция расширения E, выполняющая расширение 32 бит до 48 (принимает блок из 32 бит и порождает блок из 48 бит), определяется табл. 4.3.

Таблица 4.3

Функция расширения E

32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1

Как видно из табл. 4.3 часть бит (например, 1, 32 и т.д.) повторяются несколько раз; за счет этого 32 бита расширяются до 48 бит.

Полученный результат (обозначим его $E(R_{i-1})$) складывается по модулю 2 (операция XOR) с текущим значением ключа K_i (который состоит из 48 бит) и затем разбивается на восемь 6-битовых блоков Bi, B_2, ..., B_8:

$$E(R_{i-1})+Ki = B_1B_2 \ldots B_8.$$

Далее каждый из этих блоков используется как номер элемента в функциях-матрицах S_1, S_2, ..., S_8, содержащих 4-битовые значения.

Следует отметить, что выбор элемента в матрице S_j осуществляется достаточно оригинальным образом.

Пусть на вход матрицы S_j, поступает 6-битовый блок Bj = b_1 b_2 b_3 b_4 b_5 b_6, тогда двух битовое число b_1b_6 указывает номер строки матрицы, а четырех битовое число $Ь_2$ $Ь_3$ b_4 b_5 - номер столбца.

Например, если на вход матрицы S_1 поступает 6-битовый блок B_1= $Ь_1$ b_2 b_3 b_4 b_5 b_6 = = 100110, то 2-битовое число b_1 b_6 = $10_{(2)} = 2_{10}$ указывает строку с номером 2 матрицы S_1, а 4-битовое число b_2 b_3'b_4 b_5=$0011_{(2)}$=$3_{(10)}$ указывает столбец с номером 3 матрицы S_1.

Это означает, что в матрице S_1 блок B_1 = 100110 выбирает элемент на пересечении строки с номером 2 и столбца с номером 3, т.е. элемент $8_{(10)}$ = =$1000_{(2)}$. Совокупность 6-битовых блоков B_1, B_2,..., B_8 обеспечивает выбор четырех битового элемента в каждой из матриц S_1, S_2, ..., S_8.

В результате получаем $S_1(B_1)$ $S_2(B_2)$ $S_3(B_3)$ ... $S_8(B_8)$, т.е. 32-битовый блок (поскольку матрицы S_j содержат 4-битовые элементы). Этот 32-битовый блок преобразуется с помощью функции перестановки битов P (табл.4.4).

Таблица 4.4

Функция перестановки P

16	7	20	21
29	12	28	17
1	15	23	26
5	18	31	10
2	8	24	14
32	27	3	9
19	13	30	6
22	11	4	25

Таким образом, функция шифрования
$f(R_{i\_1},K_i) = P(S_1(B_1).....S_8(B_8))$.

На каждой итерации (рис. 16) используется новое значение ключа K_j (длиной 48 бит). Новое значение ключа K_j вычисляется из начального ключа К (рис.18).

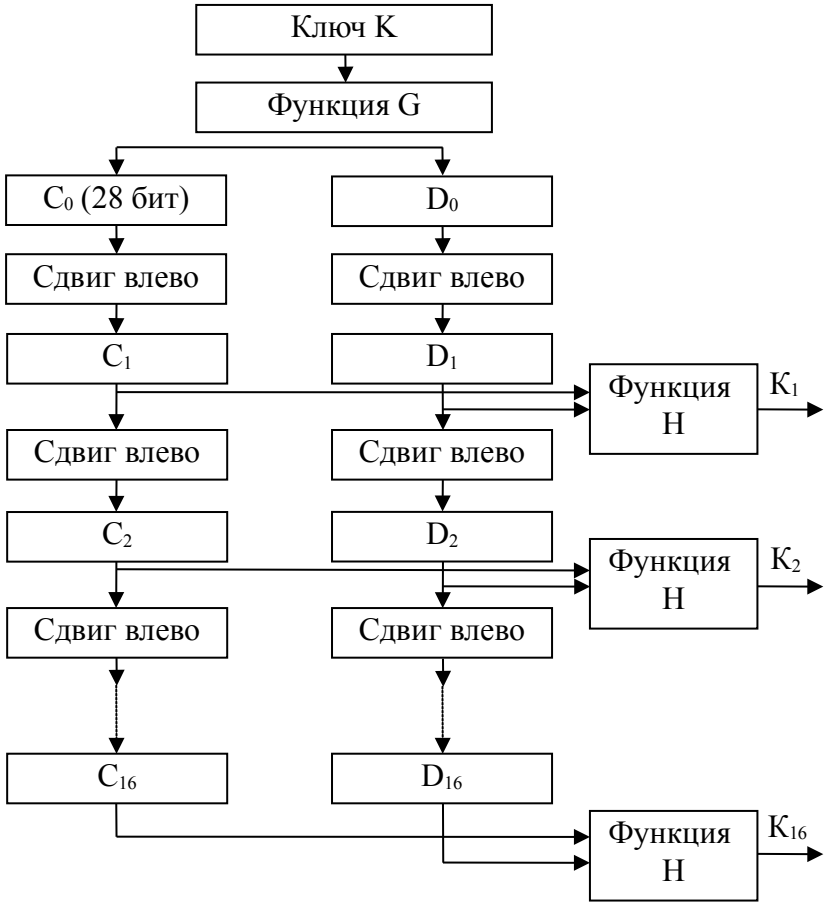


Рис. 18. Схема алгоритма вычисления ключей K_i

Ключ К представляет собой 64-битовый блок с 8 битами контроля по четности, расположенными в позициях 8, 16, 24, 32, 40, 48, 56, 64. Для удаления контрольных битов и подготовки ключа к работе используется функция G первоначальной подготовки ключа (табл. 4.5).

Таблица 4.5

Функция G первоначальной подготовки ключа

57	49	41	33	25	17	9
1	58	50	42	34	26	18
10	2	59	51	43	35	27
19	11	3	60	52	44	36
63	55	47	39	31	23	15
7	62	54	46	38	30	22
14	6	61	53	45	37	29
21	13	5	28	20	12	4

Табл. 4.5 разделена на две части. Результат преобразования G(K) разбивается на две половины C_0 и D_0 по 28 бит каждая. Первые четыре строки матрицы G определяют, как выбираются биты последовательности C_0 (первым битом C_0 будет бит 57 ключа шифра, затем бит 49 и т.д., а последними битами - биты 44 и 36 ключа).

Следующие четыре строки матрицы G определяют, как выбираются биты последовательности D_0 (т.е. последовательность D_0 будет состоять из битов 63, 55, 47. .12, 4 ключа шифра).

Как видно из табл. 4.5, для генерации последовательностей Co и Do не используются биты 8, 16, 24, 32, 40, 48, 56 и 64 ключа шифра. Эти биты не влияют на шифрование и могут служить для других целей (например, для контроля по четности). Таким образом, в действительности ключ шифра является 56-битовым.

После определения C_0 и D_0 рекурсивно определяются C_i и D_i, i = 1, 2, …, 16. Для этого применяются операции циклического сдвига влево на один или два бита в зависимости от номера шага итерации, как показано в табл. 4.6.

Операции сдвига выполняются для последовательностей C_i и D_i независимо. Например, последовательность C_3 получается посредством циклического сдвига влево на две позиции последовательности C_2, а последовательность D_3 - посредством сдвига влево на две позиции последовательности D_2, C_{16} и D_{16} получаются из C_{15} и D_{15} посредством сдвига влево на одну позицию.

Таблица 4.6

Таблица сдвигов S_i для вычисления ключа

Номер итерации	Количество Si сдвигов влево, бит	Номер итерации	Количество s, сдвигов влево, бит
1	1	9	1
2	1	10	2
3	2	11	2
4	2	12	2
5	2	13	2
6	2	14	2
7	2	15	2
8	2	16	1

Ключ K_j, определяемый на каждом шаге итерации, есть результат выбора конкретных битов из 56-битовой последовательности C_j , D_j и их перестановки. Другими словами, ключ K_j =H(C_j D_j), где функция H определяется матрицей, завершающей обработку ключа (табл. 4.7).

Таблица 4.7

Функция H завершающей обработки ключа

14	17	11	24	1	5
3	28	15	6	21	10
23	19	12	4	26	8
16	7	27	20	13	2
41	52	31	37	47	55
30	40	51	45	33	48
44	49	39	56	34	53
46	42	50	36	29	32

Как следует из табл.4.7, первым битом ключа K_j будет 14-й бит последовательности C_j D_j, вторым - 17-й бит, 47-м битом ключа K_j будет 29-й бит C_jD_j а 48-м битом - 32-й бит C_jD_j.

4.2. Основные режимы работы алгоритма DES

Алгоритм DES можно использовать как для шифрования, так и для аутентификации данных. Он позволяет непосредственно преобразовывать 64-битовый входной открытый текст в 64-битовый выходной шифрованный текст. Однако данные редко ограничиваются 64 разрядами.

Чтобы воспользоваться алгоритмом DES для решения разнообразных криптографических задач, разработаны четыре рабочих режима использования алгоритма:

электронная кодовая книга ECB (Electronic Code Book);
сцепление блоков шифра CBC (Cipher Block Chaining);
обратная связь по шифротексту CFB (Cipher Feed Back);
обратная связь по выходу OFB (Output Feed Back).

Режим "Электронная кодовая книга" (ECB).

Сообщение разбивают на 64-битовые отрезки (блоки) по 8 байтов. Каждый из этих блоков шифруют независимо с использованием одного и того же ключа шифрования (рис. 19).

Основное достоинство - простота реализации. Недостаток - относительно слабая устойчивость против квалифицированных криптоаналитиков. Из-за фиксированного характера шифрования при ограниченной длине блока 64 бита возможно проведение криптоанализа "со словарем". Блок такого размера может повториться в сообщении вследствие большой избыточности в тексте на естественном языке.

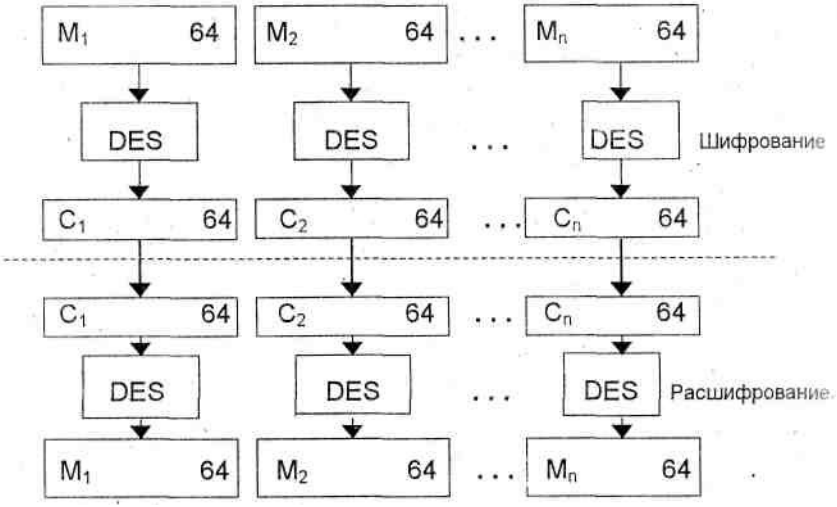


Рис. 19. Схема алгоритма DES в режиме электронной кодовой книги

Это приводит к тому, что идентичные блоки открытого текста в сообщении будут представлены идентичными блоками шифротекста, что дает криптоаналитику некоторую информацию о содержании сообщения.

Режим "Сцепление блоков шифра" (CBC).

В этом режиме исходный файл M разбивается на 64-битовые блоки: $M = M_1M_2...M_n$. Первый блок M_1 складывается

по модулю 2 с 64-битовым начальным вектором IV, который меняется ежедневно и держится в секрете (рис. 20). Полученная сумма затем шифруется с использованием ключа DES, известного и отправителю, и получателю информации. Полученный 64-битовый шифр C_1 складывается по модулю 2 со вторым блоком текста, результат шифруется и получается второй 64-битовый шифр C_2, и т.д. Процедура повторяется до тех пор, пока не будут обработаны все блоки текста.

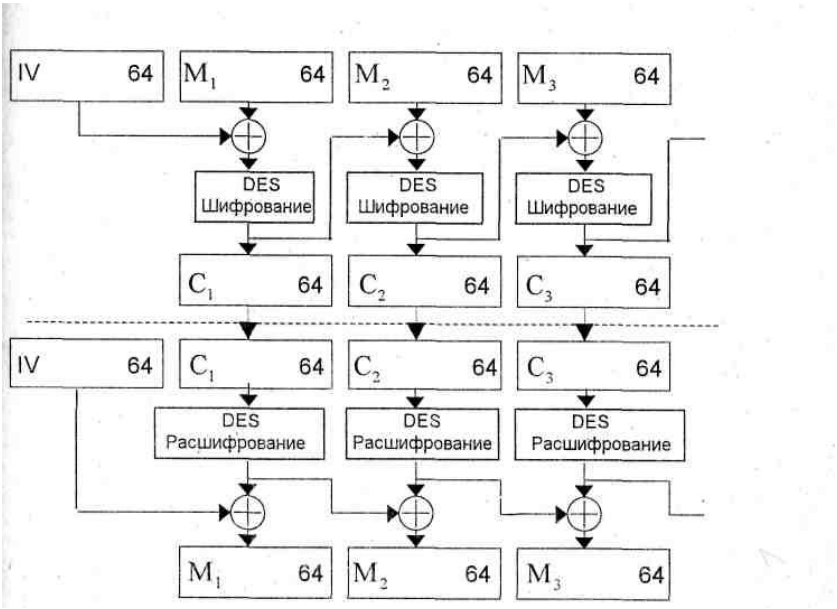


Рис. 20. Схема алгоритма DES в режиме сцепления блоков шифра

Таким образом, для всех i = 1...n (n - число блоков) результат шифрования Ci_j определяется следующим образом: C_i =DES (M_i + C_{i-1}), где C_0 = IV - начальное значение шифра, равное начальному вектору (вектору инициализации).

Очевидно, что последний 64-битовый блок шифротекста является функцией секретного ключа, начального вектора и каждого бита открытого текста независимо от его длины.

Этот последний блок шифротекста называют кодом аутентификации сообщения (КАС).

Код КАС может быть легко проверен получателем, владеющим секретным ключом и начальным вектором, путем повторения процедуры, выполненной отправителем. Посторонний, однако, не может осуществить генерацию КАС, который воспринялся бы получателем как подлинный, чтобы добавить его к ложному сообщению, либо отделить КАС от истинного сообщения для использования его с измененным или ложным сообщением.

Достоинство данного режима в том, что он не позволяет накапливаться ошибкам при передаче.

Блок M_i является функцией только C_{i-1} и C_i. Поэтому ошибка при передаче приведет к потере только двух блоков исходного текста.

Режим "Обратная связь по шифру" (CFB).

В этом режиме размер блока может отличаться от 64 бит (рис. 21). Файл, подлежащий шифрованию (расшифрованию), считывается последовательными блоками длиной к битов (к =1 ...64).

Схема шифрования реализуется следующим образом.

1. Исходное сообщение (файл) разбивается на блоки длиной к битов каждый (остаток дописывается нулями или пробелами).

2. Задается входной блок (называется 64-битовый регистр сдвига), который содержит 64-битовый вектор инициализации.

3. Входной блок подвергается DES шифрованию.

4. Полученный после шифрования блок разделяется на k старших бит и 64-k бит. K старших бит складываются по модулю 2 с k битами открытого текста. Блок, полученный после сложения, является k битовым блоком шифротекста.

5. k битовый блок шифротекста участвует в обновлении регистра сдвига. Из входного блока удаляются k старших

битов, остальные биты сдвигаются влево и на освободившееся место записывается k битовый блок шифротекста.

6. Процесс шифрования заканчивается, когда будут зашифрованы все блоки открытого текста.

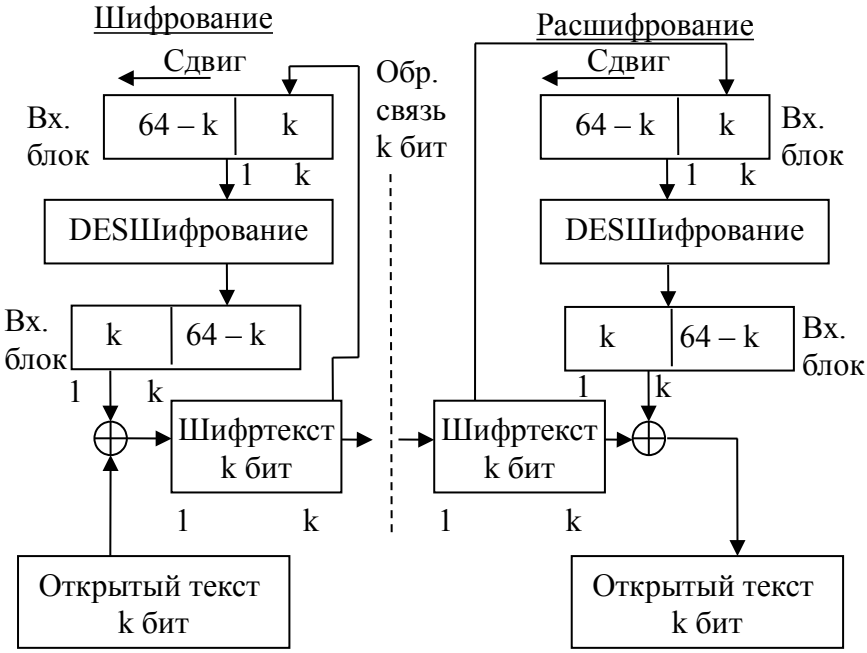


Рис. 21. Схема алгоритма DES в режиме обратной связи по шифротексту

Таким образом, получение i =1...n блоков шифротекста осуществляется по формуле:

$Ci = Mi + Pi-1$,

где Pi-1 обозначает к старших битов предыдущего зашифрованного блока.

Восстановление зашифрованных данных также выполняется относительно просто: Pi-1 и Ci вычисляются аналогичным образом и

$Mi = Ci + Pi-1$.

Режим "Обратная связь по выходу" (OFB).

Этот режим тоже использует переменный размер блока и сдвиговый регистр, инициализируемый так же, как в режиме CFB, а именно - входной блок вначале содержит вектор инициализации IV, выровненный по правому краю (рис. 22).

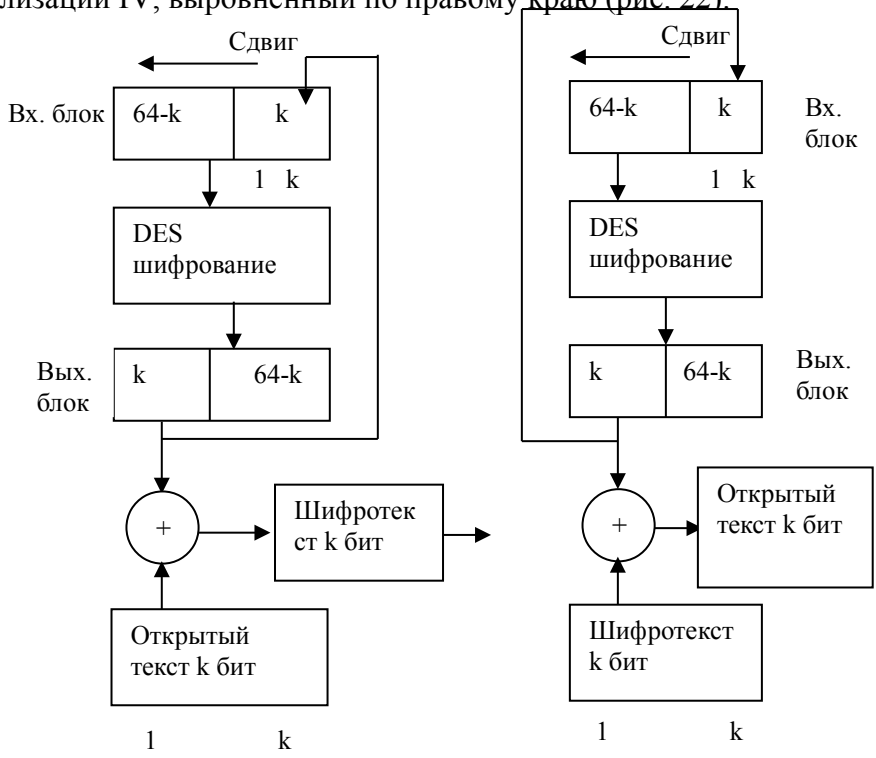


Рис. 22. Схема алгоритма DES в режиме обратной связи по выходу

При этом для каждого сеанса шифрования данных необходимо использовать новое начальное состояние регистра, которое должно пересылаться по каналу открытым текстом.

Для всех i = 1... n $C_i = M_i + P_i$,

где P_i - старшие к битов операции DES (C_{i-1}).

Шифрование Расшифрование

Отличие от режима обратной связи по шифротексту состоит в методе обновления сдвигового регистра.

Это осуществляется путем отбрасывания старших к битов и дописывания справа P_i.

Области применения алгоритма DES.

Каждому из рассмотренных режимов (ECB, CBC, CFB, OFB) свойственны свои достоинства и недостатки, что обуславливает области их применения.

Режим ECB хорошо подходит для шифрования ключей. Режим CFB, как правило, предназначается для шифрования отдельных символов, а режим OFB нередко применяется для шифрования в спутниковых системах связи.

Режимы CBC и CFB пригодны для аутентификации данных. Эти режимы позволяют использовать алгоритм DES для:

интерактивного шифрования при обмене данными между терминалом и главной ЭВМ;

шифрования криптографического ключа в практике автоматизированного распространения ключей;

шифрования файлов, почтовых отправлений, данных спутников и других практических задач.

Первоначально стандарт DES предназначался для шифрования и расшифрования данных ЭВМ. Однако его применение было обобщено и на аутентификацию.

В системах автоматической обработки данных человек не в состоянии просмотреть данные, чтобы установить, не внесены ли в них какие-либо изменения. При огромных объемах данных, проходящих в современных системах обработки, просмотр занял бы слишком много времени. К тому же избыточность данных может оказаться недостаточной для обнаружения ошибок. Даже в тех случаях, когда просмотр человеком возможен, данные могут быть изменены таким образом, что обнаружить эти изменения человеку очень трудно. Например, "do" может быть заменено на "do not", "$1900" - на "$9100".

Без дополнительной информации человек при просмотре может легко принять измененные данные за подлинные. Такие опасности могут существовать даже при использовании шифрования данных. Поэтому желательно иметь автоматическое средство обнаружения преднамеренных и непреднамеренных изменений данных.

Обыкновенные коды, обнаруживающие ошибки, непригодны, так как если алгоритм образования кода известен, противник может выработать правильный код после внесения изменений в данные. Однако с помощью алгоритма DES можно образовать криптографическую контрольную сумму, которая может защитить как от случайных, так и преднамеренных, но несанкционированных изменений данных.

Этот процесс описывает стандарт для аутентификации данных ЭВМ (FIPS 113). Суть стандарта состоит в том, что данные зашифровываются в режиме обратной связи по шифротексту (режим CFB) или в режиме сцепления блоков шифра (режим CBC), в результате чего получается последний блок шифра, представляющий собой функцию всех разрядов открытого текста. После этого сообщение, которое содержит открытый текст, может быть передано вместе с зашифрованным последним блоком шифротекста, служащего в качестве криптографической контрольной суммы.

Одни и те же данные можно защитить, пользуясь как шифрованием, так и аутентификацией. Данные защищаются от ознакомления шифрованием, а изменения обнаруживаются посредством аутентификации. Алгоритм аутентификации можно применить как к открытому, так и к зашифрованному тексту. При финансовых операциях, когда в большинстве случаев реализуются и шифрование, и аутентификация, последняя применяется и к открытому тексту.

Шифрование и аутентификацию используют для защиты данных, хранящихся в ЭВМ. Во многих ЭВМ пароли зашифровывают необратимым образом и хранят в памяти машины. Когда пользователь обращается к ЭВМ и вводит пароль,

последний зашифровывается и сравнивается с хранящимся значением. Если обе зашифрованные величины одинаковы, пользователь получает доступ к машине, в противном случае следует отказ.

Нередко зашифрованный пароль вырабатывают с помощью алгоритма DES, причем ключ полагается равным паролю, а открытый текст - коду идентификации пользователя.

С помощью алгоритма DES можно также зашифровать файлы ЭВМ для их хранения.

Одним из наиболее важных применений алгоритма DES является защита сообщений электронных систем платежей (ЭСП) при операциях с широкой клиентурой и между банками.

Алгоритм DES реализуется в банковских автоматах, терминалах в торговых точках, автоматизированных рабочих местах и главных ЭВМ. Диапазон защищаемых им данных весьма широк - от оплат $50 до переводов на многие миллионы долларов. Гибкость основного алгоритма DES позволяет использовать его в самых разнообразных электронных системах платежей.

Повышение криптостойкости алгоритма DES.

Число возможных ключей шифрования в криптосистеме DES равно $2^{56} = 128^8 = 72\ 057\ 594\ 037\ 927\ 936$. Если же при выборе ключа шифрования ограничиться лишь печатными символами (например, взятыми из пароля пользователя), то число возможных ключей уменьшится до $96^8 = 7\ 213\ 895\ 789\ 838\ 336$.

Для повышения криптостойкости алгоритма DES, вызванной недостаточными на сегодняшний день длиной ключа шифрования и числом раундов, используются различные модификации этой криптосистемы. Среди них наиболее известны 3- DES и DESX.

В 3-DES к одному и тому же блоку открытого текста M функция шифрования применяется трижды с тремя разными ключами (k1, k2, k3), что обеспечивает увеличение длины ключа окончательного шифрования и числа раундов в три раза:

$C = E_{k3}(D_{k2}(E_{k1}(M)))$.

Расшифрование выполняется следующим образом:

$M = D_{k1}(E_{k2}(D_{k3}(C)))$.

На втором шаге тройного DES используется не функция шифрования, а функция расшифрования, поскольку при k1=k2=k3 результат шифрования по алгоритму 3-DES совпадает с шифрованием по алгоритму DES на ключе k1. Использование двойного DES более уязвимо для криптоанализа.

Недостатком алгоритма 3-DES является снижение производительности шифрования в три раза по сравнению с алгоритмом DES. Этого недостатка лишен алгоритм DESX:

$C = k2 + E_k (k1 + M)$,

где k – ключ DES-шифрования длиной 56 бит; k1 и k2 – дополнительные ключи шифрования длиной 64 бита каждый; + - операция сложения по модулю 2.

Общая длина ключа шифрования, используемого в алгоритме DESX, составляет, таким образом, 184 бита. Расшифрование шифротекста по алгоритму DESX производится следующим образом:

$M = D_k(C + k_2) + k_1$.

Многие операционные системы семейства Unix включают в свой состав системную программу des, реализующую шифрование (расшифрование) по алгоритму DES информации со стандартного устройства ввода с передачей результата на стандартное устройство вывода.

В операционных системах Windows, как в открытых, начиная с версии Windows 95 OSR2, так и защищенных доступ к шифрованию по алгоритму DES и другим возможен с помощью функций криптографического интерфейса CryptoAPI.

4.3. Криптографическая система ГОСТ 28147-89

Используемая в Российской Федерации криптосистема определена в стандарте ГОСТ 28147-89 «Системы обработки информации. Защита криптографическая. Алгоритм криптографического преобразования данных» (в 1989 г. с этого алгоритма был снят гриф секретности, хотя он был разработан значительно раньше). Стандарт обязателен для организаций, предприятий и учреждений, применяющих криптографическую защиту данных, хранимых и передаваемых в сетях ЭВМ, в отдельных вычислительных комплексах и ЭВМ.

Этот алгоритм криптографического преобразования данных предназначен для аппаратной и программной реализации, удовлетворяет криптографическим требованиям и не накладывает ограничений на степень секретности защищаемой информации.

В алгоритме ГОСТ 28147-89 используется ключ шифрования k длиной 256 бит, который может рассматриваться как массив из восьми 32-битных элементов k_0, k_1, …, k_7 (внутренних ключей). Дополнительным ключевым элементом алгоритма является таблица замен S, представляющая собой матрицу из восьми строк и шестнадцати столбцов, элементы которой – целые числа от 0 до 15. Каждая строка таблицы замен должна содержать 16 различных чисел. Таким образом, общий размер таблицы замен составляет 512 бит.

Таким образом, алгоритм шифрования данных представляет собой 64-битовый блочный алгоритм с 256-битовым ключом.

При описании алгоритма используются следующие обозначения: L и R - последовательности битов; LR - конкатенация последовательностей L и R, в которой биты последовательности R следуют за битами последовательности L; + - операция побитового сложения по модулю 2; [+]- операция сложения по модулю 2^{32} двух 32-разрядных двоичных чисел; {+} - операция сложения двух 32-разрядных чисел по модулю 2^{32} -1.

Два целых числа a, b, где $0 <= a, b <= 2^{32}-1$,

$a= (a_{32}a_{31} \ldots a_2a_1)$, $b = (b_{32}, b_{31}.....b_2, b_1)$,

представленные в двоичном виде, т.е.

$a = a_{32}2^{31} + a_{31}2^{30} + ... + a_2 2^1 + a_1$,

$b = b_{32}2^{31} + b_{31}2^{30} + ... + b_2 2^1 + b_1$,

суммируются по модулю 2^{32} (операция [+]) по следующему правилу:

$a [+] b = a + b$, если $a + b < 2^{32}$,

$a [+] b = a + b - 2^{32}$ если $a + b > 2^{32}$.

Правила суммирования чисел по модулю $2^{32}- 1$:

$a \{+\} b = a + b$, если $a + b < 2^{32}- 1$,

$a \{+\} b = a + b-(2^{32}-1)$, если $a + b >= 2^{32}-1$.

Алгоритм предусматривает четыре режима работы:

- шифрование данных в режиме простой замены;
- шифрование данных в режиме гаммирования;
- шифрование данных в режиме гаммирования с обратной связью;
- выработка имитовставки.

Режим простой замены.

Для реализации алгоритма шифрования данных в режиме простой замены используется только часть блоков общей криптосистемы, [illegible] значения на их

N_1, N_2 32-разрядные накопители;

$СМ_1$ - 32-разрядный сумматор по модулю 2^{32} ([+]);

$СМ_2$ - 32-разрядный сумматор по модулю 2 (+);

R - 3[illegible] регистр цик[illegible]

КЗУ [illegible] запоминающее устройство на 256 бит, состоящее [illegible] 32-разрядных накопителей X_0, X_1, X_2, ..., X_7;

S - б[illegible]новки, состоящий из восьми узлов замены (8-блоков S_2, S_3, ..., S

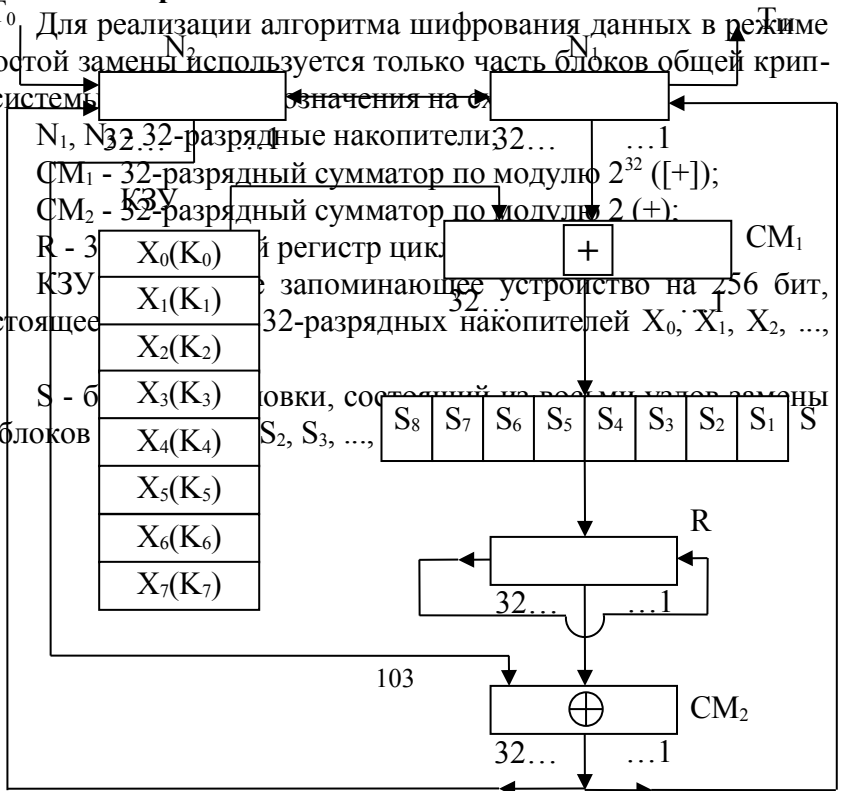


Рис. 23. Схема реализации режима простой замены

Зашифрование открытых данных в режиме простой замены.

1. Открытые данные, подлежащие зашифрованию, разбивают на 64-разрядные блоки T_0.

2. Процедура зашифрования 64-разрядного блока T_0 в режиме простой замены включает 32 цикла (i = 1...32).

3. В ключевое запоминающее устройство (КЗУ) вводят 256 бит ключа К в виде восьми 32-разрядных подключей (чисел) K_i:

$K=K_7K_6K_5K_4K_3K_2K_1K_o$.

4. Последовательность битов блока открытого текста

$T_0 = (a_1(0), a_2(0),\ldots,a_{31}(0), a_{32}(0), b_1(O), b_2(0),...,b_{31}(0), b_{32}(0))$ разбивают на две последовательности по 32 бита: b(0) a(0), где b(0) - левые или старшие биты, a(0) - правые или младшие биты.

Эти последовательности вводят в накопители N_1 и N_2 перед началом первого цикла зашифрования. В результате начальное заполнение накопителя N_1

$a(0) = (a_{32}(0), a_{31}(0),\ldots,a_2(0), a_1(0),$

32, 31,…, 2, 1 номер разряда N_1

начальное заполнение накопителя N_2

$b(0) = (b_{32}(0), b_{31}(0),\ldots,b_2(0), b_1(0))$.

32, 31, ... 2, 1 номер разряда N_2.

5. Первый цикл (i=1) процедуры зашифрования 64-разрядного блока открытых данных можно описать уравнениями:

$a(1) = f(a(0)\ [+]\ K_0) + b(0)$,

$b(1) = a(0)$.

Здесь a(1) - заполнение N_1 после 1-го цикла зашифрования; b(1) - заполнение N_2 после 1-го цикла зашифрования; f - функция шифрования.

Аргументом функции f является сумма по модулю 2^{32} числа a(0) (начального заполнения накопителя N_1) и числа K_0 - подключа, считываемого из накопителя X_0 КЗУ. Каждое из этих чисел равно 32 битам.

Функция f включает две операции над полученной 32-разрядной суммой (a (0) [+] K_0).

Первая операция называется подстановкой (заменой) и выполняется блоком подстановки S. Блок подстановки S состоит из восьми узлов замены (S-блоков замены) $S_1,S_2,\ldots,S_8$ с памятью 64 бит каждый. Поступающий из $СМ_1$ на блок подстановки S 32-разрядный вектор разбивают на восемь последовательно идущих 4-разрядных векторов, каждый из которых преобразуется в четырехразрядный вектор соответствующим узлом замены. Каждый узел замены можно представить в виде таблицы-перестановки шестнадцати четырехразрядных двоич-

ных чисел в диапазоне 0000... 1111. Входной вектор указывает адрес строки в таблице, а число в этой строке является выходным вектором. Затем четырехразрядные выходные векторы последовательно объединяют в 32-разрядный вектор. Узлы замены (таблицы-перестановки) представляют собой ключевые элементы, которые являются общими для сети ЭВМ и редко изменяются. Эти узлы замены должны сохраняться в секрете.

Вторая операция - циклический сдвиг влево (на 11 разрядов) 32-разрядного вектора, полученного с выхода блока подстановки S. Циклический сдвиг выполняется регистром сдвига R.

Далее результат работы функции шифрования f суммируют поразрядно по модулю 2 в сумматоре $СМ_2$ с 32-разрядным начальным заполнением b(0) накопителя N_2. Затем полученный на выходе $СМ_2$ результат (значение a(1)) записывают в накопитель N_1, а старое значение N_1 (значение a(0)) переписывают в накопитель N_2 (значение Ь(1) = a(0)). Первый цикл завершен.

6. Последующие циклы осуществляются аналогично, при этом во втором цикле из КЗУ считывают заполнение X_1 – подключ $К_1$, в третьем цикле - подключ $К_2$ и т.д., в восьмом цикле - подключ $К_7$. В циклах с 9-го по 16-й, а также в циклах с 17-го по 24-й подключи из КЗУ считываются в том же порядке: $К_0$, $К_1$,$К_2$,...,$К_6$,$К_7$. В последних восьми циклах с 25-го по 32-й порядок считывания подключей из КЗУ обратный: $К_7$, $К_6$,…, К., $К_1$, $К_0$.

Таким образом, при зашифровании в 32 циклах осуществляется следующий порядок выборки из КЗУ подключей:

$К_0$, $К_1$, $К_2$, $К_3$, $К_4$, $К_5$, $К_6$, $К_7$, $К_0$, $К_1$, $К_2$, Кз, $К_4$, $К_5$, $К_6$, $К_7$,

$К_0$, $К_1$, $К_2$, $К_3$, $К_4$, $К_5$, $К_6$, $К_7$, $К_7$, $К_6$, $К_5$, $К_4$, Кз, $К_2$, $К_1$, $К_0$.

В 32-м цикле результат из сумматора $СМ_2$ вводится в накопитель N_2, а в накопителе N_1 сохраняется прежнее заполнение. Полученные после 32-го цикла зашифрования заполнения накопителей N_1 и N_2 являются блоком зашифрованных данных Тш, соответствующим блоку открытых данных То.

Уравнения зашифрования в режиме простой замены имеют вид:

$a(j) = f(a(j-1)\ [+]\ K_{(j-1)\ \mathrm{mod}\ 8}) + b(j-1)$ при j =1…24,
$b(j) = a(j-1)$

$a(j) = f(a(j-1)\ [+]\ K_{32-j}) + b(j-1)$ при j = 25…31
$b(j) = a(j-1)$

$a(32) = f(a(31)\ [+]\ K_0) + b(31)$ при j= 32
$b(32) = b(31)$

где $a(j) = (a_{32}(j), a_{31}(j),\ldots, a_1(j))$ - заполнение N_1 после j-го цикла зашифрования;

$b(j) = (b_{32}(j), b_{31}(j),\ldots, b_1(j))$ – заполнение N_2 после j-го цикла зашифрования, j =1…32.

7. Блок зашифрованных данных Тш (64 разряда) выводится из накопителей N_1, N_2 в следующем порядке: из разрядов 1...32 накопителя N_1, затем из разрядов 1...32 накопителя N_2, т.е. начиная с младших разрядов:

$Тш= (a_1(32), a_2(32), ..., a_{32}(32), b_1(32), b_2(32), ..., b_{32}(32))$.

Остальные блоки открытых данных зашифровываются в режиме простой замены аналогично.

Расшифрование в режиме простой замены.

Криптосхема, реализующая алгоритм расшифрования в режиме простой замены, имеет тот же вид, что и при зашифровании (см. рис. 15).

В КЗУ вводят 256 бит ключа, на котором осуществлялось зашифрование. Зашифрованные данные, подлежащие расшифрованию, разбиты на блоки Тш по 64 бита в каждом. Ввод любого блока

$Тш= (a_1(32), a_2(32), ..., a_{32}(32), b_1(32), b_2(32), ..., b_{32}(32))$

в накопители N_1 и N_2 производят так, чтобы начальное значение накопителя N_1 имело вид

$(a_{32}(32), a_{31}(32), ..., a_2(32), a_1(32))$,

32, 31,..., 2, 1 <-номер разряда N_1

а начальное заполнение накопителя N_2 - вид
($b_{32}(32), b_{31}(32), ..., b_2(32), b_1(32)$).
32, 31,..., 2, 1 <-номер разряда N_2

Расшифрование осуществляется по тому же алгоритму, что и зашифрование, с тем изменением, что заполнения накопителей X_0, X_1, ..., X_7 считываются из КЗУ в циклах расшифрования в следующем порядке:

$K_0, K_1, K_2, K_3, K_4, K_5, K_6, K_7, K_7, K_6, K_5, K_4, K_3, K_2, K_1, K_0,$
$K_7, K_6, K_5, K_4, K_3, K_2, K_1, K_0, K_7, K_6, K_5, K_4, K_3, K_2, K_1, K_0.$

Уравнения расшифрования имеют вид:

$a(32-j) = f(a(32-j+1)\ [+]\ K_{j-1}) + b(32-j+1)$ при $j = 1 \ldots 8$;
$b(32-j) = a(32-j+1)$

$a(32-j) = f(a(32-j+1)\ [+]\ K_{(32-j) \bmod 8}) + b(32-j+1)$ при $j=9 \ldots 31$;
$b(32-j) = a(32-j+1)$

$a(0) = f(a(1)\ [+]\ K_0) + b(1)$
$b(0) = b(1)$ при $j = 32$.

Полученные после 32 циклов работы заполнения накопителей N_1 и N_2 образуют блок открытых данных

$T_0 = (a_1(0), a_2(0), ..., a_{32}(0), b_1(0), b_2(0)\ \ b_{32}(0))$,

соответствующий блоку зашифрованных данных Тш. При этом состояние накопителя N_1

($a_{32}(0), a_{31}(0), ... a_2(0), a_1(0)$),
32, 31,…, 2, 1 номер разряда N_1

состояние накопителя N_2

($b_{32}(0), b_{31}(0), ... b_2(0), b_1(0)$).
32, 31,…, 2, 1 номер разряда N_2

Аналогично расшифровываются остальные блоки зашифрованных данных.

Если алгоритм зашифрования в режиме простой замены 64-битового блока T_0 обозначить через А, то

$A(To) = A(a\ (0), Б(0)) = (a\ (32), b(32)) = T_Ш$.

Следует иметь в виду, что режим простой замены допустимо использовать для шифрования данных только в ограниченных случаях - при выработке ключа и зашифровании его с обеспечением имитозащиты для передачи по каналам связи или для хранения в памяти ЭВМ.

Режим гаммирования.

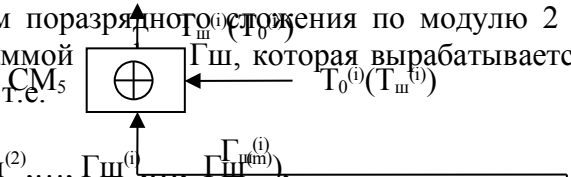
Зашифрование открытых данных в режиме гаммирования. Криптосхема, реализующая алгоритм зашифрования в режиме гаммирования, показана на рис. 24.

Открытые данные разбивают на 64-разрядные блоки

$T_0^{(1)}, T_0^{(2)},\ldots,T_0^{(i)},\ldots,T_0^{(m)}$,

где $T_0^{(i)}$ - i-й 64-разрядный блок открытых данных, i = 1...m, m определяется объемом шифруемых данных.

Эти блоки поочередно зашифровываются в режиме гаммирования путем поразрядного сложения по модулю 2 в сумматоре $СМ_5$ с гаммой Гш, которая вырабатывается блоками по 64 бита, т.е.



Гш = ($Гш^{(1)}$, $Гш^{(2)}$,…, $Гш^{(i)}$,…, $Гш^{(m)}$),

где $Гш^{(i)}$ - i-й 64-разрядный блок, i = 1...m.

Число двоичных разрядов в блоке $T_0^{(m)}$ может быть меньше 64, поэтому использованная часть гаммы шифра из блока $Гш^{(m)}$ отбрасывается.

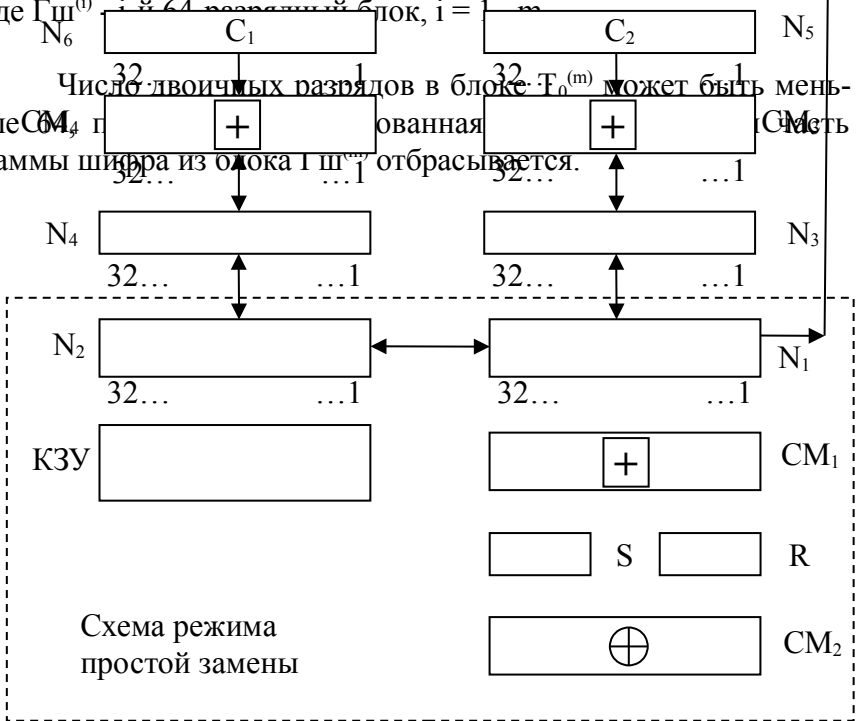


Рис. 24. Схема реализации режима гаммирования

Уравнение зашифрования данных в режиме гаммирования имеет вид

$Тш^{(i)} = T_0^{(i)} + Гш^{(i)}$,

где $Гш^{(i)} = A(У_{i-1} [+] C_2, Z_{i-1} \{+\} C_1)$, i=1 ...m; $Тш^{(i)}$ - i-й блок 64-разрядного блока зашифрованного текста; A() - функция зашифрования в режиме простой замены; C_1, C_2 - 32-разрядные двоичные константы; Yi, Zi - 32-разрядные двоичные последовательности.

Величины Yi, Zi определяются итерационно по мере формирования гаммы Гш следующим образом:

$(Y_0, Z_0) = A(S)$,

где S - синхропосылка (64-разрядная двоичная последовательность),

$(Yi, Zi) = (Y_{i-1} [+] C_2, Z_{i-1} \{+\} C_1)$, i = 1...m.

Рассмотрим реализацию процедуры зашифрования в режиме гаммирования.

1. В накопители N_6 и N_5 заранее записаны 32-разрядные двоичные константы C_1 и C_2, имеющие следующие значения (в шестнадцатеричной форме):

$C_1 = 01010104_{(16)}$, $C_2 = 01010101_{(16)}$.

2. В КЗУ вводится 256 бит ключа.

3. В накопители N_1 и N_2 вводится 64-разрядная двоичная последовательность (синхропосылка)

$S = (S_1, S_2, ..., S_{64})$.

Синхропосылка S является исходным заполнением накопителей N_1 и N_2 для последовательной выработки m блоков гаммы шифра.

Исходное заполнение накопителя N_1:

$(S_{32}, S_{31}, ...,S_2, S_1)$;

32, 31, ..., 2, 1 номер разряда N_1.

Исходное заполнение накопителя N_2:

$(S_{64}, S_{63}, \dots, S_{34}, Sзз)$-

32, 31, ..., 2, 1 номер разряда N_2.

4. Исходное заполнение N_1 и N_2 (синхропосылка S) зашифровывается в режиме простой замены. Результат зашифрования

$A(S) = (Yo,Z_0)$

переписывается в 32-разрядные накопители N_3 и N_4 так, что заполнение N_1 переписывается в N_3, а заполнение N_2 - в N_4.

5. Заполнение накопителя N_4 суммируют по модулю $(2^{32} - 1)$ в сумматоре $СМ_4$ с 32-разрядной константой C_1 из накопителя N_6. Результат записывается в N_4. Заполнение накопителя N_3 суммируется по модулю 2^{32} в сумматоре $СМ_3$ с 32-разрядной константой C_2 из накопителя N_5. Результат записывается в N_3.

6. Заполнение N_3 переписывают в N_1, а заполнение N_4 - в N_2, при этом заполнения N_3, N_4 сохраняются.

7. Заполнение накопителей N_1 и N_2 зашифровывается в режиме простой замены.

8. Полученное в результате зашифрования заполнение накопителей N_1, N_2 образует первый 64-разрядный блок гаммы шифра $Г_Ш^{(1)}=(y_1^{(1)}, y_2^{(1)}, \ldots, y_{63}^{(1)}, y_{64}^{(1)})$. Блок гаммы суммируют поразрядно по модулю 2 в сумматоре $СМ_5$ с первым 64-разрядным блоком открытых данных

$T_0^{(1)} = (t_1^{(1)}, t_2^{(1)}, \ldots, t_{63}^{(1)}, t_{64}^{(1)})$.

В результате суммирования по модулю 2 значений $Г_ш^{(1)}$ и $T_0^{(1)}$ получают первый 64-разрядный блок зашифрованных данных.

9. Для получения следующего 64-разрядного блока гаммы шифра $Г_ш^{(2)}$ заполнение N_4 суммируется по модулю $(2^{32} - 1)$ в сумматоре $СМ_4$ с константой C_1 из N_6. Результат записывается в N_4. Заполнение N_3 суммируется по модулю 2^{32} в сумматоре $СМ_3$ с константой C_2 из N_5. Результат записывается в N_3. Новое заполнение N_3 переписывают в N_1, а новое заполнение N_4 - в N_2, при этом заполнения N_3 и N_4 сохраняют. Заполнения N_1, N_2 зашифровывают в режиме простой замены.

10. Полученное в результате зашифрования заполнение накопителей N_1 и N_2 образует второй 64-разрядный блок гаммы шифра $Г_ш^{(2)}$, который суммируется поразрядно по модулю 2 в сумматоре $СМ_5$ со вторым блоком открытых данных $T_0^{(2)}$:

$Тш^{(2)} = Гш^{(2)} + T_0^{(2)}$.

11. Аналогично вырабатываются блоки гаммы шифра $Г_ш^{(3)}$, $Гш^4$, …, $Г_ш^{(m)}$ и зашифровываются блоки открытых данных $T_0^{(3)}$, $T_0^{(4)}$, …, $T_0^{(m)}$.

В канал связи или память ЭВМ передаются синхропосылка S и блоки зашифрованных данных

$T_ш^{(1)}$, $T_ш^{(2)}$, …, $T_ш^{(m)}$.

Предполагается также, что получатель знает ключ шифрования данных.

Расшифрование в режиме гаммирования. При расшифровании криптосхема имеет тот же вид, что и при зашифровании (см. рис. 16).

Уравнение расшифрования:

$To^{(i)} = T_ш^{(i)} + Г_ш^{(i)} = T_ш^{(i)} + A(Y_{i-1}\ [+]\ C_2, Z_{i-1}\ \{+\}\ C_1)$, i =1…m.

Следует отметить, что 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.

Рассмотрим реализацию процедуры расшифрования.

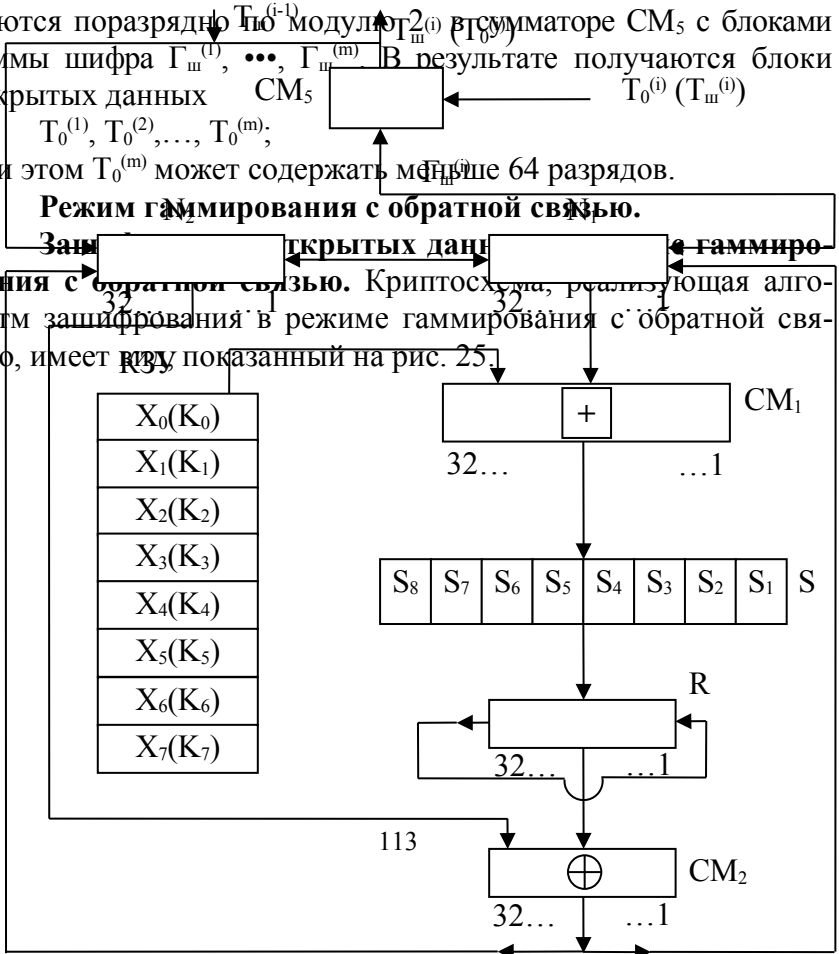
1. В КЗУ вводят 256 бит ключа, с помощью которого осуществляется зашифрование данных $T_0^{(1)}$, $T_0^{(2)}$, ..., $T_0^{(m)}$.

2. В накопители N_1 и N_2 вводится синхропосылка, и осуществляется процесс выработки m блоков гаммы шифра $Г_ш^{(1)}$, $Г_ш^{(2)}$, ..., $Г_ш^{(m)}$.

3. Блоки зашифрованных данных $T_ш^{(1)}, T_ш^{(2)},\ldots, T_ш^{(m)}$ суммируются поразрядно по модулю 2 в сумматоре $СМ_5$ с блоками гаммы шифра $Г_ш^{(1)}$, •••, $Г_ш^{(m)}$. В результате получаются блоки открытых данных

$T_0^{(1)}, T_0^{(2)},\ldots, T_0^{(m)}$;

при этом $T_0^{(m)}$ может содержать меньше 64 разрядов.



Режим гаммирования с обратной связью.

Зашифрование открытых данных в режиме гаммирования с обратной связью. Криптосхема, реализующая алгоритм зашифрования в режиме гаммирования с обратной связью, имеет вид показанный на рис. 25.

Рис. 25. Схема реализации режима гаммирования с обратной связью

Открытые данные, разбитые на 64-разрядные блоки $T_0^{(1)}$, $To^{(2)}$……$T_0^{(m)}$, зашифровываются в режиме гаммирования с обратной связью путем поразрядного сложения по модулю 2 с гаммой шифра $\Gamma_ш$, которая вырабатывается блоками по 64 бита:

$\Gamma_ш = (\Gamma_ш^{(1)}, \Gamma_ш^{(2)}, \ldots, \Gamma_ш^{(m)})$.

Число двоичных разрядов в блоке $T_0^{(m)}$ может быть меньше 64, при этом неиспользованная для шифрования часть гаммы шифра из блока $\Gamma_ш^{(m)}$ отбрасывается.

Уравнения зашифрования в режиме гаммирования с обратной связью имеют вид:

$T_ш^{(1)} = A(S) + T_0^{(1)} = Г_ш^{(1)} + T_0^{(1)}$,

$T_ш^{(i)} = A\,(T_ш^{(i-1)}) + T_0^{(i)} = Г_ш^{(i)} + T_0^{(i)}$, $i = 2....m$.

Здесь $T_ш^{(i)}$ - i-й 64-разрядный блок зашифрованного текста; A(-) -функция зашифрования в режиме простой замены; m - определяется объемом открытых данных.

Аргументом функции A (•) на первом шаге итеративного алгоритма является 64-разрядная синхропосылка S, а на всех последующих шагах - предыдущий блок зашифрованных данных $T_ш^{(i-1)}$.

Процедура зашифрования данных в режиме гаммирования с обратной связью реализуется следующим образом.

1. В КЗУ вводят 256 бит ключа.

2. В накопители N_1 и N_2 записывают синхропосылку $S = (Si, S_2, ..., S_{64})$ из 64 бит.

3. Исходное заполнение накопителей N_1 и N_2 зашифровывают в режиме простой замены. Полученное в результате зашифрования заполнение накопителей N_1 и N_2 образует первый 64-разрядный блок гаммы шифра $Гш^{(1)}=A(S)$.

4. Блок гаммы суммируют поразрядно по модулю 2 в сумматоре $СМ_5$ с первым 64-разрядным блоком открытых данных

$T_0^{(1)} = (t_1^{(1)}, t_2^{(1)}, \ldots, t_{64}^{(1)})$.

В результате получают первый 64-разрядный блок зашифрованных данных

$T_ш^{(1)} = Г_ш^{(1)} + T_0^{(1)}$,

где $T_ш^{(1)} = (r_1^{(1)}, r_2^{(1)}, \ldots, r_{64}^{(1)})$.

5. Блок зашифрованных данных $T_ш^{(1)}$ одновременно является также исходным состоянием накопителей N_1, N_2 для выработки второго блока гаммы шифра $Г_ш^{(2)}$, и поэтому по обратной связи $T_ш^{(1)}$ записывается в указанные накопители N_1 и N_2.

Заполнение накопителя N_1

$(r_{32}^{(l)}, r_{31}^{(l)}, \ldots, r_2^{(l)}, r_1^{(l)})$

32, 31,… 2, 1 номер разряда N_1.

Заполнение накопителя N_2

($r_{64}^{(1)}$, $r_{63}^{(1)}$,…, $r_{34}^{(1)}$,$r_{33}^{(1)}$).

32, 31, ..., 2, 1 номер разряда N_2.

6. Содержимое накопителей N_1 и N_2 зашифровывают в режиме простой замены. Полученное в результате зашифрования заполнение накопителей N_1 и N_2 образует второй 64-разрядный блок гаммы шифра $\Gamma_ш^{(2)}$, который суммируется поразрядно по модулю 2 в сумматоре $СМ_5$ со вторым блоком открытых данных $T_0^{(2)}$:

$\Gamma_ш^{(2)} + T_0^{(2)} = T_ш^{(2)}$.

7. Выработка последующих блоков гаммы шифра $\Gamma_ш^{(i)}$ и зашифрование соответствующих блоков открытых данных $T_0^{(i)}$ (i=3..,m) производится аналогично.

Если длина последнего m-го блока открытых данных $T_0^{(m)}$ меньше 64 разрядов, то из $\Gamma_ш^{(m)}$ используется только соответствующее число разрядов гаммы шифра, остальные разряды отбрасываются.

В канал связи или память ЭВМ передаются синхропосылка S и блоки зашифрованных данных $T_ш^{(1)}$, $T_ш^{(2)}$,…, $T_ш^{(m)}$.

Расшифрование в режиме гаммирования с обратной связью. При расшифровании криптосхема имеет тот же вид, что и при зашифровании (см. рис. 17).

Уравнения расшифрования:

$To^{(1)} = A(S) + T_ш^{(1)} = \Gamma_ш^{(1)} + T_ш^{(1)}$,

$T_0^{(i)} = \Gamma_ш^{(i)} + T_ш^{(i)} = A(T_Ш^{(i-1)}) + T_ш^{(i)}$, i = 2...m.

Реализация процедуры расшифрования зашифрованных данных в режиме гаммирования с обратной связью происходит следующим образом.

1. В КЗУ вводят 256 бит того же ключа, на котором осуществлялось зашифрование открытых блоков $T_0^{(1)}$, $T_0^{(2)}$, ..., $T_0^{(m)}$.

2. В накопители N_1 и N_2 вводят синхропосылку S.

3. Исходное заполнение накопителей N_1 и N_2 (синхропосылка S) зашифровывают в режиме простой замены. Полученное в результате зашифрования заполнение N_1 и N_2 образует первый блок гаммы шифра

$\Gamma_{Ш}^{(1)} = A(S)$,

который суммируется поразрядно по модулю 2 в сумматоре $СМ_5$ с блоком зашифрованных данных $Т_ш^{(1)}$.

В результате получается первый блок открытых данных

$То^{(l)} = \Gamma_ш^{(l)} + Т_ш^{(l)}$.

4. Блок зашифрованных данных $Т_ш^{(1)}$ является исходным заполнением накопителей N_1 и N_2 для выработки второго блока гаммы шифра $\Gamma_ш^{(2)}$: $\Gamma_ш^{(2)} = A(Т_Ш^{(1)})$. Полученное заполнение накопителей N_1 и N_2 зашифровывается в режиме простой замены. Образованный в результате зашифрования блок $\Gamma_ш^{(2)}$ суммируется поразрядно по модулю 2 в сумматоре $СМ_5$ со вторым блоком зашифрованных данных $Т_ш^{(2)}$. В результате получают второй блок открытых данных.

5. Аналогично в N_1, N_2 последовательно записывают блоки зашифрованных данных $Т_ш^{(2)}$, $Т_ш^{(3)}$, ... $Т_ш^{(m)}$, из которых в режиме простой замены вырабатываются блоки гаммы шифра $\Gamma_ш^{(3)}$, $\Gamma_ш^{(4)}$, ... $\Gamma_ш^{(m)}$.

Блоки гаммы шифра суммируются поразрядно по модулю 2 в сумматоре $СМ_5$ с блоками зашифрованных данных $Т_Ш^{(3)}$,$Т_Ш^{(4)}$, …, $Т_ш^{(m)}$.

В результате получают блоки открытых данных

$Т_0^{(3)}$, $Т_0^{(4)}$, …, $Т_0^{(m)}$,

при этом последний блок открытых данных $Т_0^{(m)}$ может содержать меньше 64 разрядов.

Режим выработки имитовставки.

Имитовставка - это блок из Р бит, который вырабатывают по определенному правилу из открытых данных с использованием ключа и затем добавляют к зашифрованным данным для обеспечения их имитозащиты.

Имитозащита - это защита системы шифрованной связи от навязывания ложных данных.

В стандарте **ГОСТ 28147-89** определяется процесс выработки имитовставки, который единообразен для любого из режимов шифрования данных. Имитовставка $И_р$ вырабатывается из блоков открытых данных 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 зашифровываются.

Значение параметра P (число двоичных разрядов в имитовставке) определяется криптографическими требованиями с учетом того, что вероятность навязывания ложных помех равна $1/2^p$.

Для выработки имитовставки открытые данные представляют в виде последовательности 64-разрядных блоков $T_0^{(i)}$, i=1…m..

Первый блок открытых данных $T_0^{(1)}$ подвергают преобразованию A(), соответствующему первым 16 циклам алгоритма зашифрования в режиме простой замены. В качестве ключа для выработки имитовставки используют ключ длиной 256 бит, по которому шифруют данные.

Полученное после 16 циклов 64-разрядное число A ($T_0^{(1)}$) суммируют по модулю 2 со вторым блоком открытых данных $T_0^{(2)}$.

Результат суммирования ($A(T_0^{\{1)}) + T_0^{(2)}$) снова подвергают преобразованию A (.).

Полученное 64-разрядное число $A(A(T_0^{(1)}) + T_0^{(2)})$ суммируют по модулю 2 с третьим блоком $T_0^{(3)}$ и снова подвергают преобразованию A (•), получая 64-разрядное число

$$A\,(A\,(A\,(T_0^{(1)}) + T_0^{(2)}) + To^{(3)}),$$ и т.д.

Последний блок $T_0^{(m)}$ (при необходимости дополненный нулями до полного 64-разрядного блока) суммируют по модулю 2 с результатом вычислений на шаге (m-1), после чего зашифровывают в режиме простой замены, используя преобразование A (•).

Из полученного 64-разрядного числа выбирают отрезок $И_p$ (имитовставку) длиной P бит:

$И_P = [a^{(m)}_{32-p+1}(16),\ a^{(m)}_{32-p+2}(16),\ldots,\ a^{(m)}_{32}(16)]$,

где $a_i^{(m)}$ - i-й бит 64-разрядного числа, полученного после 16-го цикла последнего преобразования A (•), 32 - p + 1 <= i <= 32.

Имитовставка $И_p$ передается по каналу связи или в память ЭВМ в конце зашифрованных данных, т.е.

$$Т_ш^{(1)}, Т_ш^{(2)},\ldots, Т_ш^{(m)}, Ир.$$

Поступившие к получателю зашифрованные данные

$$Т_ш^{(1)}, Т_ш^{(2)},\ldots, Т_ш^{(m)}$$

расшифровываются, и из полученных блоков открытых данных $To^{(1)}$, $T_0^{(2)}$, ..., $T_0^{(m)}$ аналогичным образом вырабатывается имитовставка $И_p'$. Эта имитовставка $И_p'$ сравнивается с имитовставкой $И_p$, полученной вместе с зашифрованными данными из канала связи или из памяти ЭВМ. В случае несовпадения имитовставок полученные при расшифровании блоки открытых данных $T_0^{(1)}$, $T_0^{(2)}$, ..., $T_0^{(m)}$ считают ложными.

Первоначально алгоритм ГОСТ 28147-89 предназначался для аппаратной реализации. Но с ростом производительности современных компьютеров стала возможной и эффективная программная реализация этой криптографической системы.

4.4. Асимметричные криптографические системы

Концепция асимметричных криптосистем. Эффективными системами криптографической защиты данных являются асимметричные криптосистемы, называемые также криптосистемами с открытым ключом. В таких системах для зашифрования данных используется один ключ, а для расшифрования - другой ключ (отсюда и название - асимметричные). Первый ключ является открытым и может быть опубликован для использования всеми пользователями системы, которые зашиф-

ровывают данные. Расшифрование данных с помощью открытого ключа невозможно.

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

Обобщенная схема асимметричной криптосистемы с открытым ключом показана на рис. 26.

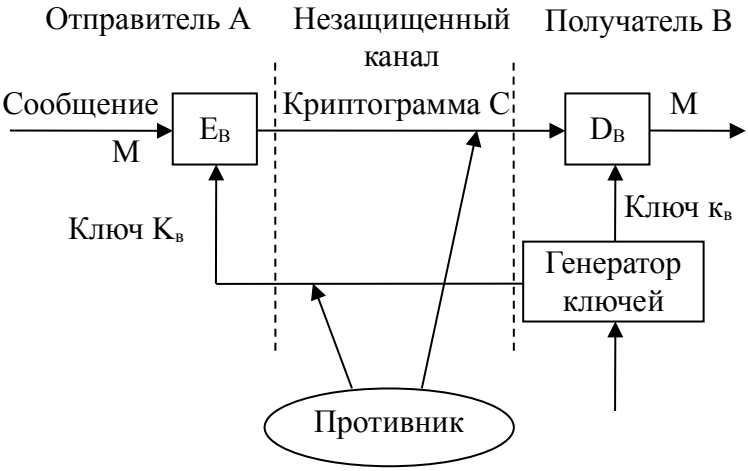


Рис. 26. Обобщенная схема асимметричной криптосистемы с открытым ключом

В этой криптосистеме применяют два различных ключа: Кв - открытый ключ отправителя А; k_B -секретный ключ получателя В. Генератор ключей целесообразно располагать на стороне получателя В (чтобы не пересылать секретный ключ k_B по незащищенному каналу). Значения ключей Кв и kв зависят от начального состояния генератора ключей.

Раскрытие секретного ключа kB по известному открытому ключу Кв должно быть вычислительно неразрешимой задачей.

Характерные особенности асимметричных криптосистем:

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

2. Алгоритмы шифрования и расшифрования являются открытыми.

E_B: M → C,

D_B: C → M

Защита информации в асимметричной криптосистеме основана на секретности ключа k_B.

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

1. Вычисление пары ключей (Кв, k_B) получателем В на основе начального условия должно быть простым.

2. Отправитель А, зная открытый ключ Кв и сообщение М, может легко вычислить криптограмму

$C = E_{Кв}(M) = E_B(M)$.

3. Получатель В, используя секретный ключ k_B и криптограмму С, может легко восстановить исходное сообщение

$M = D_{kB}(C) = D_B(C) = DB [E_B(M)]$.

4. Противник, зная открытый ключ Кв, при попытке вычислить секретный ключ k_B наталкивается на непреодолимую вычислительную проблему.

5. Противник, зная пару (Кв, С), при попытке вычислить исходное сообщение М наталкивается на непреодолимую вычислительную проблему

Однонаправленные функции. В основе асимметричных криптографических систем лежит понятие однонаправленной функции f, обладающей свойствами:

простое (не требующее больших ресурсов) вычисление значения функции y=f(x);

существование обратной функции f^{-1};

сложное (требующее ресурсов за пределами возможностей современных компьютеров) вычисление значения обратной функции $x=f^{-1}(y)$.

Фактически в асимметричной криптографии используется подкласс однонаправленных функций – однонаправленные функции с обходными путями, для которых обратная функция может быть вычислена так же просто, как и прямая, только если известна специальная информация об обходных путях. Эта специальная информация исполняет роль секретного ключа.

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

$N = P*Q$,

является относительно несложной задачей для ЭВМ.

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

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

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

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

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

$f_{AN}(x) = A^x \pmod N$,

где X - целое число, $1 \leq x \geq N-1$.

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

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

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

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

$A^x \bmod N = y$.

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

По современным оценкам теории чисел при целых числах $A \approx 2^{664}$ и $N \approx 2^{664}$ решение задачи дискретного логарифмирования (нахождение показателя степени x для известного y) потребует около 10^{26} операций, т.е. эта задача имеет в 10^3 раз большую вычислительную сложность, чем задача разложения на множители. При увеличении длины чисел разница в оценках сложности задач возрастает.

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

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

$f: X \rightarrow Y$

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

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

Области применения. К основным применениям асимметричных криптосистем относятся:

передача ключа симметричного шифрования по открытой сети (отправитель зашифровывает этот ключ с помощью открытого ключа получателя, который только и сможет расшифровать полученное сообщение с помощью своего секретного ключа);

системы электронной цифровой подписи для защиты электронных документов (создатель документа удостоверяет его подлинность с помощью своего секретного ключа, после этого любой владелец соответствующего открытого ключа сможет проверить аутентичность данного документа).

В отличие от классической симметричной криптографии криптография с открытым ключом появилась сравнительно недавно – во второй половине XX в. К особенностям современных асимметричных криптосистем, которые не позволяют им полностью заменить симметричные криптосистемы, относятся:

большая продолжительность процедур шифрования и расшифрования (примерно в 1000 раз больше);

необходимость использования существенно более длинного ключа шифрования для обеспечения той же криптостойкости шифра (например, симметричному ключу длиной 56 бит будет соответствовать асимметричный ключ длиной 384 бита, а симметричному ключу длиной 112 бит – асимметричный ключ длиной 1792 бита).

К наиболее известным асимметричным криптографическим системам относятся RSA, Диффи-Хеллмана, Эль-Гамаля и криптосистема на основе эллиптических кривых.

4.5. Криптосистема шифрования данных RSA

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

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

В криптосистеме RSA открытый ключ Кв, секретный ключ kв, сообщение М и криптограмма С принадлежат множеству целых чисел

$Z_N = \{0, 1,2, ...,N-1\}$, (4.1)

где N - модуль: $N = P*Q$. (4.2)

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

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

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

$1 < К_В <= f(N)$, НОД $(К_В, f(N)) = 1$, (4.3)

$f(N) = (P-1)(Q-1)$ (4.4)

где f(N) - функция Эйлера.

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

Второе из указанных выше условий означает, что открытый ключ K_B и функция Эйлера f(N) должны быть взаимно простыми.

Далее, используя расширенный алгоритм Евклида, вычисляют секретный ключ k_B, такой, что

$$k_B * K_B = 1 \pmod{f(N)} \qquad (4.5)$$

или

$$k_B = K_B^{-1} \pmod{(P-1)(Q-1)}.$$

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

Открытый ключ K_B используют для шифрования данных, а секретный ключ k_B - для расшифрования.

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

$$C = E_{KB}(M) = E_B(M) = M^{KB} \pmod{N}. \qquad (4.6)$$

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

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

Однако обратную задачу, т.е. задачу расшифрования криптограммы C, можно решить, используя пару (секретный ключ k_B, криптограмма C) по следующей формуле:

$$M = Dk_B(C) = D_B(C) = C^{kB} \pmod{N}. \qquad (4.7)$$

Процесс расшифрования можно записать так:

$$D_B(E_B(M)) = M. \qquad (4.8)$$

Подставляя в (4.8) значения (4.6) и (4.7), получаем:

$(M^{K_B})^{k_B} = M \pmod N$

или

$M^{K_B k_B} = M \pmod N.$ (4.9)

Величина f(N) играет важную роль в теореме Эйлера, которая утверждает, что если НОД (x, N) =1, то

$X^{f(N)} = 1 \pmod N$,

или в несколько более общей форме

$X^{nf(N)+1} = X \pmod N$ (4.10)

Сопоставляя выражения (4.9) и (4.10), получаем

$K_B * k_B = n * f(N) + 1$

или, что то же самое,

$K_B * k_B = 1 \pmod{f(N)}$.

Именно поэтому для вычисления секретного ключа k_B используют соотношение (4.5).

Таким образом, если криптограмму

$C = M^{K_B} \pmod N$

возвести в степень k_B, то в результате восстанавливается исходный открытый текст M, так как

$(M^{KB})^{kB} = м^{KBkB} = M^{nf(N)+1} = M \pmod N$.

Таким образом, получатель B, который создает криптосистему, защищает два параметра: 1) секретный ключ k_B и 2) пару чисел (P, Q), произведение которых дает значение модуля N. С другой стороны, получатель B открывает значение модуля N и открытый ключ K_B.

Противнику известны лишь значения K_B и N. Если бы он смог разложить число N на множители P и Q, то он узнал бы "потайной ход" - тройку чисел {P, Q, K_B}, вычислил значение функции Эйлера

$f(N) = (P-1)(Q-1)$

и определил значение секретного ключа k_B.

Однако, как уже отмечалось, разложение очень большого N на множители вычислительно не осуществимо (при условии, что длины выбранных P и Q составляют не менее 100 десятичных знаков).

Процедуры шифрования и расшифрования в криптосистеме RSA.

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

1. Пользователь B выбирает два произвольных больших простых числа P и Q.

2. Пользователь B вычисляет значение модуля $N = P * Q$.

3. Пользователь B вычисляет функцию Эйлера

$f(N) = (P-1)(Q-1)$

и выбирает случайным образом значение открытого ключа K_B с учетом выполнения условий:

$1 < K_B <= f(N)$, НОД(K_B, $f(N)$) = 1.

4. Пользователь B вычисляет значение секретного ключа k_B, используя расширенный алгоритм Евклида при решении сравнения

$k_B = K_B^{-1} \pmod{f(N)}$.

5. Пользователь B пересылает пользователю A пару чисел (N, K_B) по незащищенному каналу.

Если пользователь A хочет передать пользователю B сообщение M, он выполняет следующие шаги.

1. Пользователь A разбивает исходный открытый текст M на блоки, каждый из которых может быть представлен в виде числа $M_i = 0, 1, 2, ..., N-1$.

2. Пользователь A шифрует текст, представленный в виде последовательности чисел M_i по формуле $C_i = M_i^{K_B} \pmod{}$

N) и отправляет криптограмму C_1, C_2, C_3, ..., C_i, ... пользователю B.

3. Пользователь B расшифровывает . принятую криптограмму C_1, C_2, C_3, ..., C_i, ... используя секретный ключ k_B, по формуле $M_i = C_i^{k_B} (\text{mod } N)$.

В результате будет получена последовательность чисел Mi, которые представляют собой исходное сообщение M. Чтобы алгоритм RSA имел практическую ценность, необходимо иметь возможность без существенных затрат генерировать большие простые числа, уметь оперативно вычислять значения ключей K_B и k_B.

Пример. ***Шифрование сообщения CAB***. Для простоты вычислений будут использоваться небольшие числа. На практике применяются очень большие числа.

Действия пользователя B.

Выбирает P = 3 и Q = 11.

Вычисляет модуль N = P*Q = 3*11 = 33.

Вычисляет значение функции Эйлера для N = 33:

f(N) = f(33) = (P -1)(Q -1) = 2*10 = 20.

Выбирает в качестве открытого ключа K_B произвольное число с учетом выполнения условий:

$1 < K_B <= 20$, НОД (K_B, 20) = 1.

Пусть $K_B = 7$.

4. Вычисляет значение секретного ключа k_B, используя расширенный алгоритм Евклида при решении сравнения

$k_B = 7^{-1} (\text{mod } 20)$.

Решение дает $k_B = 3$.

5. Пересылает пользователю A пару чисел (N = 33, $K_B = 7$).

Рассмотрим подробнее, как получено решение $k_B=3$.

Расчет k_B осуществляется с использованием частного режима работы расширенного алгоритма Евклида, использующего равенство НОД(K_B, f(N))=1.

1. Введем трехмерные векторы u, v, t

u = {0, 1, f(N)}, v = (1, 0, K_B)

2. Если u_3=1, то переход к пункту 4; в противном случае, переход к пункту 3.

3. S=[u_3 / v_3]целая часть; t=u-v*s; u=v; v=t; переход к пункту 2.

Конец, $k_B = u_1$.

Процесс поиска решения можно задать таблицей.

S	U_1	U_2	U_3	V_1	V_2	V_3
	0	1	20	1	0	7
2	1	0	7	-2	1	6
1	-2	1	6	3	-1	1
1	3	-1	1			

В результате решения получаем, что k_B=3.

Действия пользователя А.

1. Представляет шифруемое сообщение как последовательность целых чисел в диапазоне 0 ... 32. Пусть буква А представляется как число 1, буква В – как число 2, буква С - как число 3. Тогда сообщение САВ можно представить как последовательность чисел 312, т.е. M_1 = 3, M_2 = 1, Мз= 2.

2. Шифрует текст, представленный в виде последовательности чисел M_1, M_2 и M_3, используя ключ $К_B$= 7 и N = 33, по формуле

$Ci = M_i^{KB} (mod N) = M_i^7 (mod 33)$.

Получает

$C_1 = 3^7 (mod 33) = 2187 (mod 33) = 9$,

$C_2 = 1^7 (mod 33) = 1 (mod 33) = 1$,

$Сз = 2^7 (mod 33) = 128 (mod 33) = 29$.

Отправляет пользователю В криптограмму

$C_1, C_2, C_3 = 9, 1, 29$.

Действия пользователя В.

Расшифровывает принятую криптограмму C_1, C_2, C_3, используя секретный ключ k_B= 3, по формуле

$M_i = C_i^{kB} (mod N) = C_i^3 (mod 33)$.

Получает

$M_1 = 9^3 \pmod{33} = 729 \pmod{33} = 3$,

$M_2 = 1^3 \pmod{33} = 1 \pmod{33} = 1$,

$M_3 = 29^3 \pmod{33} = 24389 \pmod{33} = 2$.

Таким образом, восстановлено исходное сообщение: С А В - 3 1 2.

Безопасность и быстродействие криптосистемы RSA.

Безопасность алгоритма РЗА базируется на трудности решения задачи факторизации больших чисел, являющихся произведениями двух больших простых чисел. Действительно, криптостойкость алгоритма RSA определяется тем, что после формирования секретного ключа k_B и открытого ключа K_B "стираются" значения простых чисел P и Q, и тогда исключительно трудно определить секретный ключ k_B по открытому ключу K_B, поскольку для этого необходимо решить задачу нахождения делителей P и Q модуля N.

Разложение величины N на простые множители P и Q позволяет вычислить функцию $f(N) = (P-1)(Q-1)$ и затем определить секретное значение k_B, используя уравнение

$K_B * k_B = 1 \pmod{f(N)}$.

Другим возможным способом криптоанализа алгоритма RSA является непосредственное вычисление или подбор значения функции $f(N) = (P-1)(Q-1)$. Если установлено значение $f(N)$, то сомножители P и Q вычисляются достаточно просто. В самом деле, пусть

$x = P + Q = N + 1 - f(N)$,

$y = (P-Q)^2 = (P+Q)^2 - 4*N$.

Зная $f(N)$, можно определить x и затем y; зная x и y, можно определить числа P и Q из следующих соотношений:

$$P = 1/2\,(x + \sqrt{y}), \qquad Q = 1/2\,(x - \sqrt{y}).$$

Однако эта атака не проще задачи факторизации модуля N.

Задача факторизации является трудно разрешимой задачей для больших значений модуля N.

Сначала авторы алгоритма RSA предлагали для вычисления модуля N выбирать простые числа P и Q случайным образом, по 50 десятичных разрядов каждое. Считалось, что такие большие числа N очень трудно разложить на простые множители. Один из авторов алгоритма RSA, P. Райвест, полагал, что разложение на простые множители числа из почти 130 десятичных цифр, приведенного в их публикации, потребует более 40 квадриллионов лет машинного времени. Однако этот прогноз не оправдался из-за сравнительно быстрого прогресса компьютеров и их вычислительной мощности, а также улучшения алгоритмов факторизации.

Ряд алгоритмов факторизации приведен в [40]. Один из наиболее быстрых алгоритмов, известных в настоящее время, алгоритм NFS (Number Field Sieve) может выполнить факторизацию большого числа N (с числом десятичных разрядов больше 120) за число шагов, оцениваемых величиной

$$e^{2(\ln n)1/3(\ln(\ln n))2/3}$$

В 1994 г. было факторизовано число со 129 десятичными цифрами. Это удалось осуществить математикам А. Ленстра и М. Манасси посредством организации распределенных вычислений на 1600 компьютерах, объединенных сетью, в течение восьми месяцев. По мнению А. Ленстра и М. Манасси, их работа компрометирует криптосистемы RSA и создает большую угрозу их дальнейшим применениям. Теперь разработчикам криптоалгоритмов с открытым ключом на базе RSA приходится избегать применения чисел длиной менее 200 десятичных разрядов. Самые последние публикации предлагают применять для этого числа длиной не менее 250 - 300 десятичных разрядов.

В [102] сделана попытка расчета оценок безопасных длин ключей асимметричных криптосистем на ближайшие 20 лет исходя из прогноза развитии компьютеров и их вычислительной мощности, а также возможного совершенствования алго-

ритмов факторизации. Эти оценки (табл. 4.8.) даны для трех групп пользователей (индивидуальных пользователей, корпораций и государственных организаций), в соответствии с различием требований к их информационной безопасности. Конечно, данные оценки следует рассматривать как сугубо приблизительные, как возможную тенденцию изменений безопасных длин ключей асимметричных криптосистем со временем.

Таблица 4.8

Оценки длин ключей для асимметричных криптосистем, бит

Год	Отдельные пользователи	Корпорации	Государственные организации
1995	768	1280 .	1536
2000	1024	1280	1536
2005	1280	1536	2048
2010	1280	1536	2048
2015	1536	2048	2048

Криптосистемы RSA реализуются как аппаратным, так и программным путем.

Для аппаратной реализации операций зашифрования и расшифрования RSA разработаны специальные процессоры. Эти процессоры, реализованные на сверхбольших интегральных схемах (СБИС), позволяют выполнять операции RSA, связанные с возведением больших чисел в колоссально большую степень по модулю N, за относительно короткое время. И все же аппаратная реализация RSA примерно в 1000 раз медленнее аппаратной реализации симметричного криптоалгоритма DES.

Одна из самых быстрых аппаратных реализаций RSA с модулем 512 бит на сверхбольшой интегральной схеме имеет быстродействие 64 Кбит/с. Лучшими из серийно выпускаемых СБИС являются процессоры фирмы CYLINK, выполняющие 1024-битовое шифрование RSA.

Программная реализация RSA примерно в 100 раз медленнее программной реализации DES. С развитием технологии эти оценки могут несколько изменяться, но асимметричная криптосистема RSA никогда не достигнет быстродействия симметричных криптосистем.

Следует отметить, что малое быстродействие криптосистем RSA ограничивает область их применения, но не перечеркивает их ценность.

4.6. Криптосистемы Диффи-Хеллмана и Эль-Гамаля

Криптосистемы Диффи-Хеллмана и Эль-Гамаля основаны на вычислительной сложности задачи дискретного логарифмирования. Вычисление $y=a^x \{\mathrm{mod}\ p\}$ (p – простое число или степень простого числа, $1<x<p-1$, $1<a<p-1$, $1<b<p-1$, $a^c = b\{\mathrm{mod}\ p\}$) выполняется просто, но вычисление $x = \log_a y\{\mathrm{mod}\ p\}$ выполняется достаточно сложно.

Алгоритм Диффи-Хеллмана предназначен только для генерации ключа симметричного шифрования, который затем будет использован субъектами A и B для защищенного обмена сообщениями по открытой сети.

1. A: выбирает xa и вычисляет $y_a = a^{xa} \{\mathrm{mod}\ p\}$.
2. B: выбирает xb и вычисляет $y_b = a^{xb} \{\mathrm{mod}\ p\}$.
3. A→B: y_a.

B→A: y_b.

A: вычисляет $k_a = (y_b)^{xa} \{\mathrm{mod}\ p\}$.

B: вычисляет $k_b = (y_a)^{xb} \{\mathrm{mod}\ p\}$.

7. Конец ($k_a = k_b$ и созданный ключ может теперь использоваться для защищенного обмена сообщениями между A и B).

Значения a и p в алгоритме Диффи-Хеллмана не являются секретными, поскольку, даже зная их, нарушитель не сможет решить задачу дискретного логарифмирования и найти

значения xa и xb, чтобы вычислить сгенерированный ключ симметричного шифрования.

В криптосистеме Эль-Гамаля значение a вместе с значениями p и y составляет открытый ключ, а секретным ключом является значение x ($y = a^x \{mod\ p\}$). Шифрование открытого текста P в криптосистеме Эль-Гамаля выполняется по следующему алгоритму.

1. Выбор случайного целого числа k ($1<k<p-1$ и НОД(k, p-1) = 1).
2. $C1 = a^k \{mod\ p\}$.
3. $C2 = Py^k \{mod\ p\}$.
4. Конец (шифротекстом являются значения C_1 и C_2).

Расшифрование в криптосистеме Эль-Гамаля производится путем составления сравнения $PC_1^x = C2 \{mod\ p\}$ и решения его относительно P. Действительно: $PC_1^x \{mod\ p\} = P(a^k)^x \{mod\ p\} = P(a^x)^k \{mod\ p\} = Py^k \{mod\ p\} = C_2 \{mod\ p\}$.

Если P>=p, то открытый текст должен быть разбит на блоки, длина которых равна длине числа p. В п. 3 алгоритма шифрования вместо операции умножения может использоваться операция сложения по модулю 2 ($C_2 = P [+] y^k \{mod\ p\}$). Тогда при расшифровании восстановление открытого текста выполняется следующим образом:
$P = (C_1^x \{mod\ p\}) [+] C_2$ (так как $C_1^x \{mod\ p\} = y^k \{mod\ p\}$).

Недостатком этого варианта является то, что открытый текст должен разбиваться на блоки заранее неизвестной длины $y^k \{mod\ p\}$.

4.7. Электронная цифровая подпись и ее применение

Механизм электронной цифровой подписи должен обеспечить защиту от следующих угроз безопасности электронных документов, передаваемых по открытым компьютерным сетям или хранящихся на открытых носителях:

подготовка документа от имени другого субъекта («маскарад»);

отказ автора документа от факта его подготовки (ренегатство);

изменение получателем документа его содержания (подмена);

изменения содержания документа третьим лицом (активный перехват);

повторная передача по компьютерной сети ранее переданного документа (повтор).

Электронная цифровая подпись (ЭЦП) представляет собой относительно небольшой по объему блок данных, передаваемый (хранящийся) вместе (реже – отдельно) с подписываемым с ее помощью документом. Механизм ЭЦП состоит из двух процедур: получение (простановка) подписи с помощью секретного ключа автора документа и проверка ЭЦП при помощи открытого ключа автора документа.

Алгоритм получения ЭЦП под документом P.

1. Вычисление хеш-значения H(P) для документа P.

2. Шифрование H(P) с помощью секретного ключа автора документа ska – $E_{ska}(H(P))$ (полученный шифротекст и будет являться ЭЦП).

Алгоритм проверки ЭЦП C под документом P.

1. Вычисление хеш-значения H(P) для документа P.

2. Расшифрование ЭЦП с помощью открытого ключа автора документа pka – $D_{pka}(C) = D_{pka}(E_{ska}(H(P)) = H(P)$.

3.Сравнение вычисленного и расшифрованного хеш-значения для документа P.

Перед получением ЭЦП в подписываемый документ должны быть включены дополнительные сведения:

дата и время простановки подписи;

срок окончания действия секретного ключа данной подписи;

реквизиты (фамилия, имя, отчество подписывающего лица, его должность и название представляемой организации);

идентификатор секретного ключа (для возможности выбора лицом, проверяющим ЭЦП, нужного открытого ключа).

В системе ЭЦП подпись под электронным документом невозможно подделать без знания секретного ключа автора документа, поэтому компрометация секретного ключа недопустима.

Известны следующие системы ЭЦП:

RSA (на основе асимметричной криптосистемы RSA);

DSS (Dijital Signature Standard, стандарт США на основе асимметричной криптосистемы Эль-Гамаля);

ГОСТ Р 34.10-94 (российский стандарт ЭЦП на основе асимметричной криптосистемы Эль-Гамаля);

ГОСТ Р 34.10-2001 (российский стандарт ЭЦП, использующий асимметричную криптосистему на основе эллиптических кривых).

Алгоритмы получения и проверки ЭЦП в системе RSA не отличаются от алгоритмов шифрования и расшифрования в аналогичной криптосистеме, за исключением того, что получение ЭЦП производится с применением секретного ключа, а проверка ЭЦП – с применением открытого ключа x.

Алгоритмы получения и проверки ЭЦП в системе Эль-Гамаля отличаются от алгоритмов шифрования и расшифрования в аналогичной криптосистеме.

Алгоритм получения ЭЦП под документом P.

1. Выбор случайного целого числа k(1<k<p-1 и НОД(k,p-1) = 1) (k – случайная составляющая ЭЦП, изменяющая подпись под вновь отправляемым по сети тем же самым документом).

2. $C_1 = a^k \{\mathrm{mod}\ p\}$.

3. Определение C_2 из сравнения $P = xC_1 + kC_2 \{\mathrm{mod}\ p-1\}$ (C1 и C2 образуют ЭЦП для P).

Проверка ЭЦП в системе Эль-Гамаля сводится к проверке сравнения $y^{C1}C1^{C2} \{\mathrm{mod}\ p\} = a^P \{\mathrm{mod}\ p\}$.

Действительно: $y^{C1}C1^{C2}$ {mod p} = $a^{xC1}a^{kC2}${mod p} = $a^{xC1+kC2}$ {mod p} = $a^{P\ \{mod\ p-1\}}$ {mod p} = $a^{P+m(p-1)}$ {mod p} = $a^{P}(a^{m})^{p-1}$ {mod p} = a^{P} {mod p}1 (из малой теоремы Ферма).

Алгоритм получения ЭЦП под документом P в системе на базе эллиптических кривых (целое число g выбирается из условия gG = 0).

1. Выбор случайного целого числа k (1<k<g) – случайной составляющей ЭЦП, изменяющей подпись под вновь отправляемым по сети тем же самым документом).
2. Вычисление c= kG.
3. r = x_C {mod g} (x_C – хкоордината точки c).
4. s = rx + kP {mod g}.
5. Если r ≠ 0 и s ≠ 0, то эти значения образуют ЭЦП, в противном случае выбирается другое k и пп. 2…4 повторяются.

Алгоритм проверки ЭЦП в системе на основе эллиптических кривых.

1. Вычисление V = P-1 {mod g}.
2. z_1 = sV {mod g}; z_2 = -rV {mod g}.
3. Вычисление c = z_1G + z_2y; R = x_C {mod g}.
4. Проверка R=r.

Защищенность системы ЭЦП от угрозы аутентичности и целостности подписанных документов зависит не только от стойкости алгоритмов используемой асимметричной криптосистемы, но и от стойкости функции хеширования. На функции хеширования, используемые в системах ЭЦП, налагаются очевидные дополнительные условия:

чувствительность к любым изменениям в документе (вставкам, удалениям, перестановкам, заменам фрагментов и отдельных символов);

минимальность вероятности того, что хеш-значения двух разных документов, независимо от их длин, совпадут.

К наиболее известным функциям хеширования относятся:

MD2, MD4, MD5 (Message Digest) – получают хеш-значение длиной 128 бит и используются в системе ЭЦП RSA;

SHA (Secure Hash Algorithm) – получает хеш-значение длиной 160 бит и используется в системе ЭЦП DSS;

ГОСТ Р 34.11-94 – получает хеш-значение длиной 256 бит и используется в российских стандартах ЭЦП;

RIPEMD (Race Integrity Primitives Evaluation Message Digest) – получает хеш-значение длиной 128 или 160 бит (две модификации).

5. ЗАЩИТА ИНФОРМАЦИИ В ОС

5.1. Дискреционное управление доступом к объектам компьютерных систем

Все элементы КС разделяются на множество субъектов и объектов. Понятие субъекта отличается от понятия пользователя КС. Пользователь КС – это физическое лицо, обладающее некоторой идентифицирующей его информацией. Возможны псевдопользователи, например, сама система. Пользователь управляет работой субъекта с помощью его интерфейсных элементов (команд меню, кнопок и т.п.).

Дискреционное управление доступом (Discretionary Access Control, DAC) к объектам КС предполагает выполнение следующих требований:

все субъекты и объекты КС должны быть однозначно идентифицированы;

для любого объекта КС должен быть определен пользователь-владелец;

владелец объекта должен обладать правом определения прав доступа к объекту со стороны любых субъектов КС;

в КС должен существовать привилегированный пользователь, обладающий правом полного доступа к любому объекту (или правом становиться владельцем любого объекта).

Последнее свойство определяет невозможность существования в КС потенциально недоступных объектов, владелец которых отсутствует.

Дискреционное управление доступом к объектам КС реализуется обычно в виде матрицы доступа, строки которой соответствуют субъектам КС, а столбцы – ее объектам. Элементы матрицы доступа определяют права доступа субъектов к объектам. В целях сокращения затрат памяти матрица доступа может задаваться в виде списков прав субъектов (для каждого из них создается список всех объектов, к которым разрешен доступ со стороны данного субъекта) или в виде списков контроля доступа (для каждого объекта КС создается список всех субъектов, которым разрешен доступ к данному объекту).

К **достоинствам дискреционного управления доступом к объектам КС** относятся относительно простая реализация (проверка прав доступа субъекта к объекту производится в момент открытия этого объекта в процессе субъекта) и хорошая изученность (в наиболее распространенных ОС универсального назначения применяется разграничение доступа на основе дискреционного управления).

Недостатки дискреционного управления доступом к объектам КС.

1. Статичность разграничения доступа – права доступа к уже открытому субъектом объекту в дальнейшем не изменяются, независимо от изменения состояния КС.

2. При использовании дискреционного управления доступом к объектам КС не существует возможности проверки, не приведет ли разрешение доступа к объекту для некоторого субъекта к нарушению безопасности информации в КС (например, владелец файла с конфиденциальной информацией, дав разрешение на его чтение другому пользователю, делает этого пользователя фактически владельцем защищаемой информации). Иначе говоря, дискреционное управление до-

ступом к объектам КС не обеспечивает защиты от утечки конфиденциальной информации.

3. Дискреционное управление доступом к объектам КС не позволяет обеспечить надежную защиту от проникновения в КС вредоносных программ. Что может произойти при передаче прав доступа к объекту.

4. Автоматическое назначение прав доступа субъектам (из-за большого числа объектов в КС в качестве субъектов доступа остаются только пользователи КС, а значение элемента матрицы доступа вычисляется с помощью функции, определяющей права доступа порожденного пользователем субъекта к данному объекту КС).

5.2. Мандатное управление доступом к объектам компьютерных систем

Мандатное управление доступом (Mandatory Access Control, MAC) к объектам КС во многом лишено отмеченных недостатков. Характерные признаки мандатного управления доступом.

1. Все субъекты и объекты КС должны быть однозначно идентифицированы.

2. Должен существовать линейно упорядоченный набор меток конфиденциальности и соответствующих им степеней допуска (нулевая метка или степень соответствует открытому объекту и степени допуска к работе только с открытыми объектами).

3. Каждому объекту КС должна быть присвоена метка конфиденциальности.

4. Каждому субъекту КС должна быть присвоена степень допуска.

5. В процессе своего существования каждый субъект должен иметь свой уровень конфиденциальности, равный максимуму из меток конфиденциальности объектов, к которым данный субъект получил доступ.

6. В КС должен существовать привилегированный пользователь, имеющий полномочия на удаление любого объекта системы.

7. Понизить метку конфиденциальности объекта может только субъект, имеющий доступ к данному объекту и обладающий специальной привилегией.

8. Право чтения информации из объекта получает только тот субъект, чья степень допуска не больше метки конфиденциальности данного объекта (правило «не читать выше»).

9. Право на запись информации в объект получает только тот субъект, чей уровень конфиденциальности не меньше метки конфиденциальности данного объекта (правило «не записывать ниже»).

Основной целью мандатного управления доступом к объектам КС является предотвращение утечки информации из объектов с высокой меткой конфиденциальности в объекты с низкой меткой конфиденциальности (противодействие созданию каналов передачи информации «сверху вниз»).

Достоинства мандатного управления доступом.

1. Более высокая надежность работы самой КС, так как при разграничении доступа к объектам контролируется и состояние самой системы, а не только соблюдение установленных правил.

2. Большая простота определения правил разграничения доступом по сравнению с дискреционным управлением (эти правила более ясны для разработчиков и пользователей КС).

Недостатки мандатного управления доступом к объектам КС.

1. Сложность программной реализации, которая увеличивает вероятность внесения ошибок и появления каналов утечки конфиденциальной информации.

2. Снижение эффективности работы КС, так как проверка прав доступа субъекта к объекту выполняется не только при

открытии объекта в процессе субъекта, но и перед выполнением любой операции чтения из объекта или записи в объект.

3. Создание дополнительных неудобств в работе пользователей КС, связанных с невозможностью изменения информации в неконфиденциальном объекте, если тот же самый процесс использует информацию из конфиденциального объекта (его уровень конфиденциальности больше нуля).

5.3. Классы защищенности

В руководящих документах Гостехкомиссии России определены девять классов защищенности АС от несанкционированного доступа, объединенных в три группы (наиболее защищенным является первый класс):

однопользовательские АС с информацией, размещенной на носителях одного уровня конфиденциальности (класс 3Б и 3А);

многопользовательские АС с одинаковыми полномочиями пользователей и информацией на носителях разного уровня конфиденциальности (классы 2Б и 2А);

многопользовательские АС с разными полномочиями пользователей и информацией разного уровня конфиденциальности (в порядке возрастания защищенности от класса 1Д до класса 1А).

В руководящих документах Гостехкомиссии России для АС класса защищенности 1Г и ниже должно быть обеспечено дискреционное управление доступом к объектам КС, а для КС класса 1В, 1Б, 1А – мандатное управление доступом к объектам КС.

5.4. Подсистема безопасности защищенных версий ОС Windows

К защищенным версиям ОС Windows относятся Windows NT/2000/XP Professional. В состав этих ОС входит подсистема безопасности, архитектура которой представлена на рис. 27.

Ядром подсистемы безопасности является локальная служба безопасности (Local Security Authority, LSA), размещающаяся в файле lsass.exe.

После загрузки ОС автоматически запускается **процесс входа** (winlogon.exe), который остается активным до перезагрузки ОС или выключения питания компьютера. Аварийное завершение процесса входа приводит к аварийному завершению работы ОС. Этим обеспечивается практическая невозможность подмены процесса входа при функционировании системы.

После нажатия пользователем комбинации клавиш Ctrl+Alt+Delete процесс входа обращается к **диспетчеру аутентификации** (msgina.dll), динамически компонуемой библиотеке функций, DLL, для приема от пользователя его логического имени и пароля, идентифицирующей информации и т.д. Диспетчер аутентификации реализован в виде библиотеки msgina.dll и использует пароли пользователей.

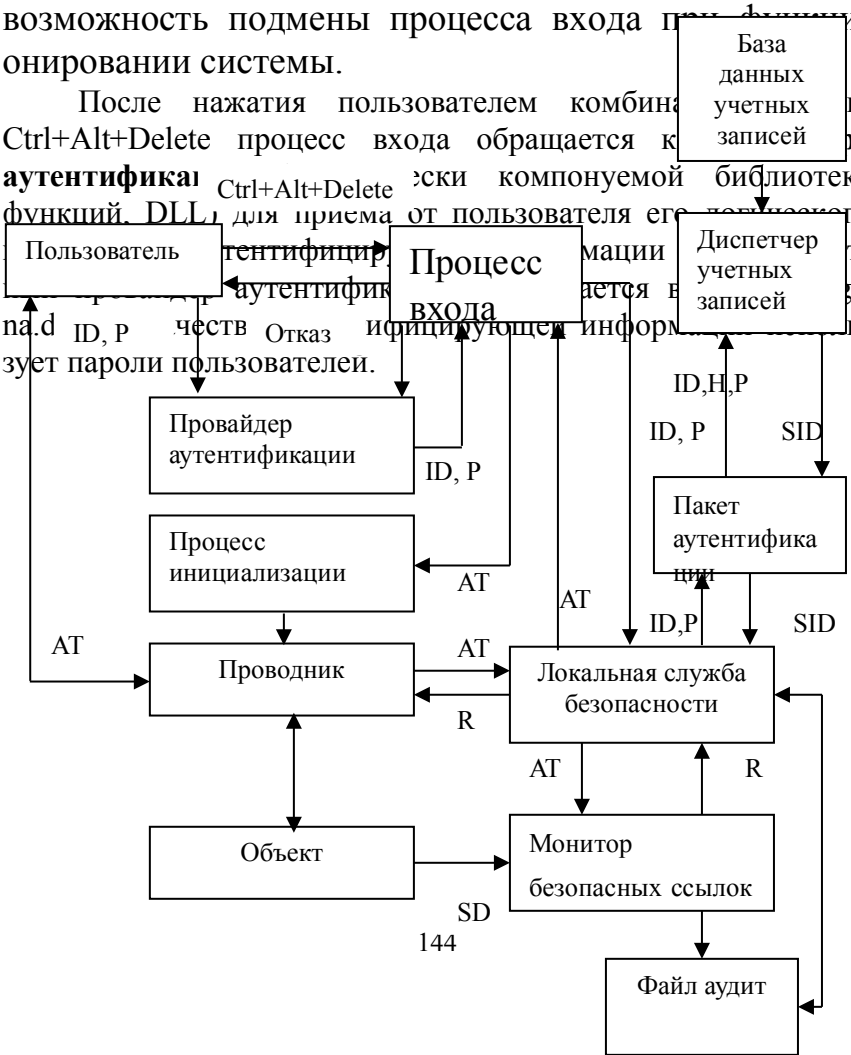


Рис. 27. Архитектура подсистемы безопасности защищенных версий Windows

Возможно использование других провайдеров аутентификации (например, считывающих ключевую информацию со смарт-карт). В этом случае необходимо, чтобы интерфейс к предоставляемым провайдером аутентификации функциям соответствовал определениям, содержащимся в файле winwlx.h (входит в состав любой системы программирования на языке C++ для Windows).

Путь к используемому провайдеру аутентификации должен быть записан в соответствующем разделе реестра.

Введенное пользователем логическое имя и пароль передаются процессом ввода в службу LSA, которая обращается к **пакету аутентификации** (библиотеке функций) для подтверждения подлинности пользователя. Если пользователь зарегистрирован на локальном компьютере, то пакет аутентификации вычисляет хеш-значение пароля H (P) и обращается к диспетчеру учетных записей (Security Account Manager, SAM) для проверки правильности введенного пароля и возможности для пользователя с введенным логическим именем начать ра-

боту в системе (не истек ли срок действия пароля, не заблокирована ли учетная запись пользователя и т.п.). Пакет аутентификации является заменяемым компонентом подсистемы безопасности (стандартный пакет аутентификации размещается в файле msvl_0.dll).

Диспетчер учетных записей обращается к базе данных учетных записей (базе данных SAM) для извлечения информации из учетной записи пользователя с введенным логическим именем. База данных учетных записей содержится в разделе реестра HKEY_LOCAL_MACHINE\SAM (а также в файле Windows\System32\Config\SAM). К базе данных SAM не может быть получен доступ для чтения или изменения с помощью штатных средств ОС даже администратором. Для ее редактирования предназначены специальные функции из набора Windows API и специальное системное приложение (в Windows XP – функция «Администрирование» панели управления).

Пароль пользователя в базе данных SAM хранится в виде двух хеш-значений, каждое из которых имеет длину 128 бит. Первое хеш-значение пароля пользователя вычисляется по алгоритму Windows NT. Второе хеш-значение пароля пользователя вычисляется по алгоритму LAN Manager.

Если проверка подтвердила подлинность пользователя и отсутствие препятствий для начала его работы в КС, то **пакет аутентификации получает от SAM уникальный идентификатор безопасности пользователя SID** (security identifier), который затем передается в LSA.

Идентификатор безопасности представляет собой структуру переменной длины, которая однозначно определяет пользователя или группу и сохраняется в регистрационной базе данных.

Получив идентификатор безопасности пользователя, локальная служба безопасности LSA создает для него **маркер доступа AT** (access token), который идентифицирует пользователя во всех его действиях с объектами КС.

В маркере доступа содержится следующая информация:

SID пользователя;

идентификаторы безопасности его групп;

полномочия пользователя;

идентификаторы безопасности пользователя и его первичной группы, которые будут использованы при создании пользователем новых объектов в КС;

дискреционный список контроля доступа по умолчанию для вновь создаваемого объекта;

источник выдачи маркера доступа и т.д.

Полномочия (привилегии) пользователя и группы назначаются администратором КС и представляют собой права субъектов на выполнение действий, относящихся к системе в целом, а не к отдельным ее объектам.

Перечислим наиболее важные привилегии, которые могут быть назначены пользователям и группам:

завершение работы системы;

изменение системного времени;

отладка программ;

архивирование файлов и каталогов;

восстановление файлов и каталогов;

управление аудитом и журналом безопасности;

смена владельцев файлов или иных объектов и т.д.

Созданный LSA маркер доступа АТ передается процессу входа, который с помощью провайдера аутентификации завершает процесс авторизации пользователя в КС, запуская процесс его инициализации (userinit.exe) и передавая ему АТ. Процесс инициализации на основе содержащегося в АТ идентификатора безопасности пользователя загружает из реестра Windows его профиль и загружает программную оболочку – проводник Windows (explorer.exe), передавая ему маркер доступа пользователя. После этого процесс инициализации завершает свою работу.

Концепция рабочего стола пользователя (desktop) в защищенных версиях Windows отличается от аналогичного понятия в открытых версиях этой операционной системы. Рабочий

стол в защищенных версиях Windows представляет собой совокупность окон, одновременно видимых на экране. Только процессы, окна которых расположены на одном рабочем столе, могут взаимодействовать между собой, используя средства графического интерфейса пользователя Windows (GUI).

Процесс входа (winlogon), получающий от пользователя имя и пароль, выполняется на отдельном рабочем столе (рабочем столе аутентификации). Никакой другой процесс, в том числе и программная закладка, внедренная нарушителем для перехвата паролей, не имеет доступа к этому рабочему столу.

Переключение экрана компьютера с одного рабочего стола на другой производится при нажатии комбинации клавиш Ctrl+Alt+Delete. В защищенных версиях Windows эта комбинация обрабатывается иначе – сообщение о нажатии данной комбинации клавиш посылается только процессу входа, который остается активным до перезагрузки ОС или выключения питания. Для всех других процессов (в частности, для всех прикладных программ, запущенных пользователем) нажатие этой комбинации клавиш совершенно незаметно.

Защита рассматриваемых версий ОС Windows от программных закладок такого рода может считаться весьма надежной.

Для управления списками пользователей и групп в КС, назначения им полномочий, определения параметров политики безопасности в ОС Windows 2000/XP администратором используются функции «Администрирование» и «Локальная политика безопасности» панели управления, а в ОС Windows NT для этого предназначалась системная программа «Диспетчер пользователей» (User Manager).

С помощью данных функций и программ можно изменить свойства учетной записи пользователя или свойства и состав группы пользователей.

В *параметрах локальной политики безопасности* определены две группы параметров учетных записей – параметры политики паролей и параметры политики блокировки учетных

записей. *Параметры политики паролей* позволяют усилить парольную аутентификацию.

К *параметрам политики блокировки учетной записи* относятся:

пороговое значение блокировки (максимальное число ошибок входа в систему);

длительность блокировки учетной записи;

интервал времени, через который происходит сброс счетчика блокировок.

5.5. Разграничение доступа субъектов к объектам КС

Для **разграничения доступа субъектов к объектам КС** в защищенных версиях ОС Windows используется **дискреционное управление доступом**.

С объектом разграничения доступа связывается дескриптор безопасности SD (security descriptor), содержащий следующую информацию:

идентификатор безопасности (SID) владельца объекта;

идентификатор безопасности первичной группы владельца;

дискреционный список контроля доступа (DACL);

системный список контроля доступа (SACL).

Список SACL управляется администратором системы. Список DACL управляется владельцем объекта и предназначен для идентификации пользователей и групп, которым предоставлен или запрещен определенный тип доступа к объекту. Каждый элемент списка DACL (access control entry, ACE) определяет права доступа к объекту для одного пользователя или группы. Каждый ACE содержит следующую информацию:

идентификатор безопасности SID субъекта, для которого определяются права доступа;

маска доступа (access mask, AM), которая специфицирует права доступа к контролируемым данным;

тип ACE;

признак наследования прав доступа к объекту, определенных для родительского объекта.

Элементы списка DACL могут быть двух типов: разрешающие и запрещающие права доступа. Элементы, запрещающие доступ, располагаются в начале списка перед элементами, разрешающими доступ.

Право доступа субъекта к объекту означает возможность обращения субъекта к объекту с помощью определенного метода (типа) доступа. В защищенных версиях ОС Windows различают специальные, стандартные и общие (generic) права доступа к объектам.

Специальные права доступа к объектам определяют возможность обращения к объекту по свойственному только данной категории объектов методу: чтение данных из объекта, запись данных в объект, чтение атрибутов объекта, выполнение программного файла и т.д.

Стандартные права доступа к объектам определяют возможность доступа к объекту по методу, применимому к любому объекту, - изменение владельца объекта, изменение списка DACL объекта, удаление объекта и т.д.

Каждое из общих прав доступа к объектам представляет собой комбинацию специальных и стандартныъ прав и предоставляет возможность обращения к объекту с помощью некоторого набора методов доступа.

Определены следующие общие права доступа:

чтение, включающее в себя чтение DACL объекта, чтение данных из объекта, чтение его атрибутов и расширенных атрибутов, использование объекта для синхронизации;

запись, включающая в себя чтение DACL объекта, запись и добавление данных в объект, запись его атрибутов и расширенных атрибутов, использование объекта для синхронизации;

выполнение, включающее в себя чтение DACL объекта, чтение его атрибутов, выполнение программного файла и использование объекта для синхронизации;

все действия с объектом.

Маркер доступа субъекта, обращающегося к некоторому объекту КС, поступает в локальную службу безопасности LSA. От LSA маркер доступа поступает к монитору безопасных ссылок (security reference monitor, SRM), который просматривает DACL из дескриптора безопасности SD соответствующего объекта и принимает решение R о предоставлении доступа субъекту или отказе в доступе (рис.28).

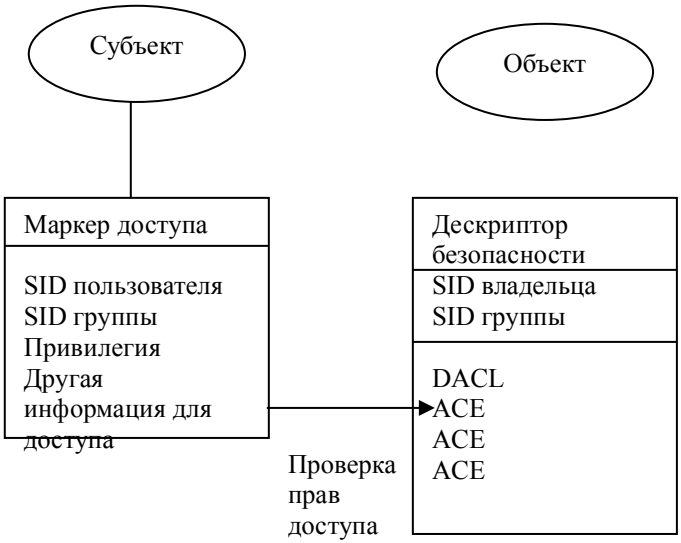


Рис. 28. Проверка прав доступа субъекта к объекту

Получив от SRM результат R, LSA передает его субъекту, запросившему доступ к объекту.

Монитор безопасных ссылок использует следующий алгоритм проверки запрошенных субъектом прав доступа к объекту.

1. Если SID из маркера доступа субъекта АТ не совпадает с SID, содержащемся в элементе ACE списка контроля доступа к объекту, то осуществляется переход к следующему ACE, в противном случае – переход к п.2.

2. Если в элементе ACE запрешается доступ к объекту для субъекта с данным SID, но этот субъект является владельцем объекта (его идентификатор безопасности совпадает с SID владельца из дескриптора безопасности SD объекта) и запрашиваемая маска доступа содержит только попытку доступа к объекту по методу «чтение (или) изменение дискреционного списка контроля доступа к объекту», то доступ субъекта к объекту разрешается, в противном случае – осуществляется переход к п. 3.

3. Если в элементе ACE запрещается доступ к объекту для субъекта с данным SID , то сравниваются запрашиваемая маска доступа и маска доступа, определенная в ACE. Если при сравнении находится хотя бы один общий метод доступа, то попытка доступа субъекта к объекту отклоняется, в противном случае – происходит переход к следующему ACE.

4. Если в элементе ACE разрешается доступ к объекту для субъекта с данным SID, то также сравниваются запрашиваемая маска доступа и маска доступа, определенная в ACE. Если при этом маски доступа полностью совпадают, то доступ субъекта к объекту разрешается, в противном случае – происходитпереход к следующему ACE.

5. Если достигнут конец списка DACL из дескриптора безопасности объекта, то попытка доступа субъекта к объекту отклоняется.

Если у объекта КС нет дескриптора безопасности (например, у папок и файлов, размещенных на дисках под управлением файловой системы FAT), то любые пользователи и группы могут получить любые права доступа к данному объекту.

Пользователи КС для назначения субъектам КС прав доступа к файлам и папкам на дисках с файловой системой

NTFS, чьими создателями-владельцами они являются, должны применять средства проводника Windows. Для этого выполняются команды «Общий доступ и безопасность» или «Свойства» контекстного меню папки либо команда «Свойства» контекстного меню файла (в операционной системе Windows XP необходимо выключить режим «Использовать простой общий доступ к файлам» на вкладке «Вид» окна свойств папки). Кнопки «Добавить» и «Удалить» позволяют изменять число элементов ACE в списке DACL, а в окне «Разрешения для имя субъекта» можно устанавливать общие права доступа к объекту конкретным пользователям и группам.

Нажатие кнопки «Дополнительно» позволяет отобразить окно настроек дополнительных параметров безопасности для объекта. На вкладке «Разрешения» можно посмотреть и при необходимости изменить любые (в том числе и специальные) права доступа к объекту. На вкладке «Владелец» можно просмотреть и при наличии соответствующей привилегии изменить информацию о владельце объекта (записать в SID владельца в дескрипторе безопасности объекта свой SID).

На вкладке «Действующие разрешения» можно проверить, какие права доступа к объекту установлены для конкретного пользователя или группы, которые выбираются с помощью кнопки «Выбрать».

Разграничение доступа субъектов к разделам реестра Windows XP производится с помощью системной программы regedit (команда «Разрешения» меню «Правка»), а в ОС Windows NT/2000 – с помощью системной программы regedt32 (меню «Безопасность»). Отметим, что разграничение доступа к разделам реестра возможно при любой используемой для хранения реестра файловой системе.

В защищенных версиях ОС Windows реализован подход, в соответствии с которым каждому процессу выделяется индивидуальное адресное пространство, которое аппаратно изолировано от адресных пространств других процессов. В этом случае, какой бы адрес оперативной памяти не использовался

в процессе, невозможно обращение к памяти, выделенной другому процессу, так как одному и тому же значению адреса в разных адресных пространствах соответствуют различные физические адреса оперативной памяти компьютера.

6. АЛГОРИТМЫ АУТЕНТИФИКАЦИИ ПОЛЬЗОВАТЕЛЕЙ

6.1. Способы аутентификации пользователей в КС

Основными способами защиты от несанкционированного доступа к информации в КС являются

аутентификация пользователей;

авторизация (определение прав доступа субъекта к объекту с конфиденциальной информацией);

шифрование информации.

Способы аутентификации пользователей в КС можно разделить на три группы.

К ***первой группе*** относятся способы аутентификации, основанные на том, что пользователь знает некоторую подтверждающую его подлинность информацию (парольная защита и аутентификация на основе модели «рукопожатия»).

Ко ***второй группе*** относятся способы аутентификации, основанные на том, что пользователь имеет некоторый материальный объект, который может подтвердить его подлинность (например, пластиковую карту с идентифицирующей пользователя информацией).

К ***третьей группе*** относятся способы аутентификации, основанные на таких данных, которые позволяют однозначно считать, что пользователь и есть тот самый субъект, за кото-

рого себя выдает (биометрические данные, особенности клавиатурного почерка и росписи мышью и т.п.).

В руководящих документах Гостехкомиссии России в АС, отнесенных к классу защищенности 1Д, должна осуществляться идентификация и проверка подлинности субъекта при входе в систему по паролю условно-постоянного действия длиной не менее шести буквенно-цифровых символов. Для классов защищенности 1Г и 1В дополнительно требуется использовать идентификатор (код, логическое имя) пользователя. Для отнесения АС к классу защищенности 1Б дополнительно необходимо использовать пароль временного действия длиной не менее восьми буквенно-цифровых символов. В требованиях к классу защищенности 1А определена необходимость применения пользователями при входе в АС биометрических характеристик или специальных устройств (жетонов, карт, электронных ключей) и пароля временного действия длиной не менее восьми буквенно-цифровых символов.

6.2. Аутентификация пользователей на основе паролей и модели «рукопожатия»

При выборе паролей пользователи КС должны руководствоваться двумя, по сути взаимоисключающими, правилами — пароли должны трудно подбираться и легко запоминаться (поскольку пароль ни при каких условиях не должен нигде записываться, так как в этом случае необходимо будет дополнительно решать задачу защиты носителя пароля).

Сложность подбора пароля определяется, в первую очередь, мощностью множества символов, используемого при выборе пароля *(N),* и минимально возможной длиной пароля *(к).* В этом случае число различных паролей может быть оценено снизу как $C_p = N^k$. Например, если множество символов пароля образуют строчные латинские буквы, а минимальная длина пароля равна 3, то C_P 26^3= 17576 (что совсем немного для программного подбора). Если же множество символов пароля

состоит из строчных и прописных латинских букв, а также из цифр (26+ 26 + 10) и минимальная длина пароля равна 6, то $C_p = 62^6 = 56800235584$.

Сложность выбираемых пользователями КС паролей должна устанавливаться администратором при реализации установленной для данной системы политики безопасности. Другими параметрами политики учетных записей при использовании парольной аутентификации должны быть:

- максимальный срок действия пароля (любой секрет не может сохраняться в тайне вечно);
- несовпадение пароля с логическим именем пользователя, под которым он зарегистрирован в КС;
- неповторяемость паролей одного пользователя.

Требование неповторяемости паролей может быть реализовано **двумя способами**. Во-первых, можно установить **минимальный срок действия пароля** (в противном случае, пользователь, вынужденный после истечения срока действия своего пароля поменять его, сможет тут же сменить пароль на старый). Во-вторых, можно вести **список уже использовавшихся данным пользователем паролей** (максимальная длина списка при этом может устанавливаться администратором).

К сожалению, обеспечить реальную уникальность каждого вновь выбираемого пользователем пароля с помощью приведенных выше мер практически невозможно. Пользователь может, не нарушая установленных ограничений, выбирать пароли «Al», «A2», ... где А — первый пароль пользователя, удовлетворяющий требованиям сложности.

Обеспечить приемлемую степень сложности паролей и их реальную уникальность можно путем назначения паролей всем пользователям администратором КС с одновременным запретом па изменение пароля самим пользователем. Для генерации паролей администратор при этом может использовать **программный генератор**, позволяющий создавать пароли различной сложности.

Однако при таком способе назначения паролей возникают проблемы, связанные с необходимостью создания защищенного канала для передачи пароля от администратора к пользователю, трудностью проверки сохранения пользователем не им выбранного пароля только в своей памяти и потенциальной возможностью администратора, знающего пароли всех пользователей, злоупотребления своими полномочиями. Поэтому наиболее целесообразным является выбор пароля пользователем на основе установленных администратором правил с возможностью задания администратором нового пароля пользователю в случае, если тот забыл свой пароль.

Еще одним аспектом политики учетных записей пользователей КС должно стать определение противодействия системы попыткам подбора паролей.

Могут применяться следующие правила:

- скрытие логического имени последнего работавшего пользователя (знание логического имени может помочь нарушителю подобрать или угадать его пароль);
- учет всех попыток (успешных и неудачных) входа в систему в журнале аудита.

Реакцией системы на неудачную попытку входа пользователя могут быть:

- блокировка учетной записи, под которой осуществляется попытка входа, при превышении максимально возможного числа попыток (на заданное время или до ручного снятия блокировки администратором);
- нарастающее увеличение временной задержки перед предоставлением пользователю следующей попытки входа.

Постоянная блокировка учетной записи при обнаружении попытки подбора пароля (до снятия блокировки администратором) менее целесообразна, поскольку она позволит нарушителю намеренно заблокировать работу в КС легального пользователя (реализовать угрозу нарушения доступности информации).

При любой реакции системы на попытку подбора пароля необходимо в настройках параметров политики учетных записей обеспечить сброс значения счетчика попыток входа в систему под конкретной учетной записью через заданный промежуток времени, иначе значения счетчика будут суммироваться для разных сеансов работы пользователя.

При **первоначальном вводе или смене пароля пользователя обычно применяются два классических правила**:

- символы вводимого пароля не отображаются на экране (это же правило, применяется и для ввода пользователем пароля при его входе в систему);
- для подтверждения правильности ввода пароля (с учетом первого правила) этот ввод повторяется дважды.

Одним из следствий первого правила является нецелесообразность назначения пользователю пароля системой, поскольку в этом случае пароль должен быть выведен пользователю в открытом виде или записан на специальном носителе (второй способ противоречит принципу сохранения пароля только в памяти пользователя).

Однако отказ от отображения символов вводимого пароля может создать проблему, так как увеличивается вероятность того, что случайная ошибка, допущенная при вводе пароля, останется не замеченной, а это может привести к блокировке учетной записи легального пользователя. Поэтому, если вход пользователя в КС происходит в защищенном помещении, в которое не могут попасть посторонние лица, от правила скрытия символов вводимого пароля можно и отказаться.

Очевидно, что в базе данных учетных записей пользователей КС пароли не могут храниться в открытом виде (иначе к ним может получить доступ как минимум администратор системы). Для хранения паролей возможно их предварительное шифрование или хеширование. Шифрование паролей имеет два недостатка:

поскольку при шифровании необходимо использовать ключ, требуется обеспечить его защищенное хранение в КС

(знание ключа шифрования пароля позволит выполнить его расшифрование и осуществить несанкционированный доступ к информации);

существует опасность расшифрования любого пароля и получения его в открытом виде.

Хеширование является необратимым преобразованием и знание хеш-значения пароля не даст нарушителю возможности его получения в открытом виде (он сможет только пытаться подобрать пароль при известной функции хеширования). Поэтому гораздо более безопасным является хранение паролей в хешированном виде. Недостатком является то, что не существует даже теоретической возможности восстановить забытый пользователем пароль.

Несмотря на то, что с помощью применения перечисленных выше правил парольную аутентификацию можно сделать более безопасной, она все-таки остается весьма уязвимой. Для ее усиления могут использоваться так называемые **одноразовые пароли**. Пусть пользователь КС получает список паролей $P_ь P_2$, ..., $P_{,,}$, ..., $P_{,,,}$. Каждый из паролей действует только на один сеанс входа (P_1 — на первый, P_2 — на второй и т.д.). В этом случае знание уже использовавшегося пользователем пароля ничего не даст нарушителю, а при каждом входе легального пользователя возможна проверка на использование данного пароля кем-либо еще.

Недостатки схемы одноразовых паролей:

- организация защищенного хранения длинного списка паролей (либо его запоминание, что маловероятно);
- неясность с номером следующего пароля, если после ввода предыдущего пароля из списка вход пользователя в систему не был осуществлен из-за сбоя в работе КС.

Эти недостатки могут быть устранены, если список паролей генерировать на основе некоторой необратимой функции, например функции хеширования.

Пусть P — начальный пароль пользователя, а F — необратимая функция. Обозначим: $F^i(P) = F(F(...F(P)...))$ (функция

F применяется последовательно i раз). Тогда список одноразовых паролей создается следующим образом:

$P_1 = F^n(P)$, $P_2 = F^{n-1}(P)$, ..., $P_{n-1} = F(F(P))$, $Pn = F(P)$.

При сбое в процессе входа пользователя в КС всегда осуществится выбор следующего пароля из списка, а система последовательно применяет функцию F к введенному пользователем паролю, вплоть до совпадения с последним принятым от него паролем (и тогда пользователь допускается к работе в системе) или до превышения длины списка паролей (в этом случае попытка входа пользователя в КС отвергается).

Но в любом варианте парольной аутентификации подтверждение подлинности пользователя осуществляется на основе ввода им некоторой конфиденциальной информации, которую можно подсмотреть, выманить, подобрать, угадать и т.п.

Рассмотрим **аутентификацию пользователей на основе модели «рукопожатия»**, во многом свободную от указанных недостатков.

В соответствии с этой моделью пользователь П и система С согласовывают при регистрации пользователя в КС функцию f, известную только им. Протокол аутентификации пользователя в этом случае выглядит следующим образом:

1. С: генерация случайного значения x; вычисление $y = f(x)$; вывод x.
2. П: вычисление $y' = f'(x)$; ввод y'.
3. С: если y и y' совпадают, то пользователь допускается к работе в системе, иначе попытка входа в систему отклоняется.

К функции f предъявляется требование, чтобы по известным x и $f(x)$ нельзя было угадать f.

Преимущества аутентификации на основе модели «рукопожатия» перед парольной аутентификацией:

- между пользователем и системой не передается никакой конфиденциальной информации, которую нужно сохранять в тайне;

• каждый следующий сеанс входа пользователя в систему отличен от предыдущего, поэтому даже длительное наблюдение за этими сеансами ничего не даст нарушителю.

К **недостаткам аутентификации на основе модели «рукопожатия»** относится большая длительность этой процедуры по сравнению с парольной аутентификацией.

Парольная аутентификация совершенно неприменима в случае взаимного подтверждения подлинности пользователей компьютерной сети. Действительно, пусть А и Б обозначают двух пользователей сети, имеющих соответственно пароли P_A и $P_Б$. Тогда протокол взаимной аутентификации А и Б мог бы выглядеть следующим образом:

1. А->Б: А, запрос $P_Б$.
2. Б—>А: Б, запрос P_A.
3. А—>Б: А, P_A.
4. Б—>А: Б, $P_Б$.

Но в момент отправки своего пароля (неважно, в открытой или защищенной форме) А не может быть уверен в подлинности Б, который может воспользоваться паролем А, чтобы выдать себя за А при взаимодействии еще с одним пользователем компьютерной сети В.

Модель «рукопожатия» вполне приемлема для взаимной аутентификации:

1. А: выбор значения x; вычисление y =f(x).
2. А—>Б: А, x.
3. Б: вычисление y' =f(x).
4. Б—>А: Б, y'.
5. А: если y и y' совпадают, то А может доверять Б.

Затем процедура аутентификации повторяется с переменой ролей (теперь Б начинает процесс и выбирает значение x), чтобы Б мог быть также уверен в подлинности А.

Для повышения безопасности протокола взаимной аутентификации перед отправкой по сети значения x и могут быть зашифрованы на секретном ключе, которым должны предварительно обменяться по защищенному каналу А и Б. В этом слу-

чае потенциальному нарушителю, который имеет возможность перехвата всех передаваемых по сети данных и желает выдать себя за одного из легальных пользователей сети, придется не только определить функцию f, но и предварительно взломать шифротекст.

При интерактивном доступе пользователя к системе функция f может быть задана таблицей своих значений. Рассмотрим два примера. В первом примере система предлагает пользователю ответить при регистрации его в КС на несколько вопросов, имеющих частично объективное и частично вымышленное содержание (например: «девичья фамилия Вашей матери», «в каком городе Вы проживали в июне 2002 г.», «где находится клуб», «когда откроется пул» и т. п.). При входе в систему пользователю предлагается ответить на другой список вопросов, среди которых есть некоторые из заданных ему при регистрации. Для правильной аутентификации пользователь должен дать те же ответы, которые он давал на аналогичные вопросы при регистрации.

Второй пример — аутентификация на основе модели «рукопожатия». При регистрации в КС пользователю предлагается набор небольших изображений (например, пиктограмм), среди которых он должен выбрать заданное число картинок. При последующем входе в систему ему выводится другой набор изображений, часть из которых он видел при регистрации. Для правильной аутентификации пользователь должен отметить те картинки, которые он выбрал при регистрации.

6.3. Аутентификация пользователей по их биометрическим характеристикам

К **основным биометрическим характеристикам пользователей КС**, которые могут применяться при их аутентификации, относится:

отпечатки пальцев;

геометрическая форма руки;
узор радужной оболочки глаза;
рисунок сетчатки глаза;
геометрическая форма и размеры лица;
тембр голоса;
геометрическая форма и размеры уха и др.

Наиболее распространенными являются **программно-аппаратные средства аутентификации пользователей по их отпечаткам пальцев**.

Для считывания этих отпечатков обычно применяются! оснащенные специальными сканерами клавиатуры и мыши. Наличие достаточно больших банков данных с отпечатками пальцев граждан является основной причиной достаточно широкого применения подобных средств аутентификации в государственных структурах, а также в крупных коммерческих организациях.

Недостатком таких средств является потенциальная возможность применения отпечатков пальцев пользователей для контроля над их частной жизнью.

Если по объективным причинам (например, из-за загрязненности помещений, в которых проводится аутентификация) получение четкого отпечатка пальца невозможно, то может применяться аутентификация по геометрической форме руки пользователя. В этом случае сканеры могут быть установлены на стене помещения.

Наиболее достоверными (но и наиболее дорогостоящими) являются средства аутентификации пользователей, основанные на характеристиках глаза (узоре радужной оболочки или рисунке сетчатки). Вероятность повторения этих признаков оценивается в 10^{-78}.

Наиболее дешевыми (но и наименее достоверными) являются средства аутентификации, основанные на геометрической форме и размере лица пользователя или на тембре его голоса. Это позволяет ис-

пользовать эти средства и для аутентификации при удаленном доступе пользователей к КС.

Основные достоинства аутентификации пользователей по их биометрическим характеристикам:

трудность фальсификации этих признаков;

высокая достоверность аутентификации из-за уникальности таких признаков;

неотделимость биометрических признаков от личности пользователя.

Для сравнения аутентификации пользователей на основе тех или иных биометрических характеристик применяются оценки вероятностей ошибок первого и второго рода.

Вероятность ошибки первого рода (отказа в доступе к КС легальному пользователю) составляет 10^{-6}... 10^{-3}. Вероятность ошибки второго рода (допуска к работе в КС незарегистрированного пользователя) в современных системах биометрической аутентификации составляет 10^{-5}... 10^{-2}.

Общим недостатком средств аутентификации пользователей КС по их биометрическим характеристикам является их более высокая стоимость по сравнению с другими средствами аутентификации, что обусловлено, в первую очередь, необходимостью приобретения дополнительных аппаратных средств.

6.4. Способы аутентификации, основанные на особенностях клавиатурного почерка и росписи мышью пользователей

Способы аутентификации, основанные на особенностях клавиатурного почерка и росписи мышью пользователей, не требуют применения специальной аппаратуры.

Одним из первых идею аутентификации пользователей по особенностям их работы с клавиатурой и мышью предложил С. П. Расторгуев. При разработке математической моде-

ли аутентификации на основе клавиатурного почерка пользователей было сделано предположение, что временные интервалы между нажатиями соседних символов ключевой фразы и между нажатиями конкретных сочетаний клавиш в ней подчиняются нормальному закону распределения. **Сутью данного способа аутентификации** является проверка гипотезы о **равенстве центров распределения двух нормальных генеральных совокупностей** (полученных при настройке системы на характеристики пользователя и при его аутентификации).

Рассмотрим вариант аутентификации пользователя по набору ключевой фразы (одной и той же в режимах настройки и подтверждения подлинности).

Процедура настройки на характеристики регистрируемого в КС пользователя:

1) выбор пользователем ключевой фразы (ее символы должны быть равномерно разнесены по клавиатуре);

2) набор ключевой фразы несколько раз;

3) исключение грубых ошибок (по специальному алгоритму);

4) расчет и сохранение оценок математических ожиданий, дисперсий и числа наблюдений для временных интервалов между наборами каждой пары соседних символов ключевой фразы.

Процедура аутентификации пользователя может проводиться в двух вариантах. **Первый вариант** процедуры аутентификации:

набор ключевой фразы пользователем несколько раз;

исключение грубых ошибок (по специальному алгоритму);

расчет оценок математических ожиданий и дисперсий для временных интервалов между нажатиями каждой пары соседних символов ключевой фразы;

решение задачи проверки гипотезы о равенстве дисперсий двух нормальных генеральных совокупностей для каждой

пары соседних символов ключевой фразы (по специальному алгоритму);

если дисперсии равны, то решение задачи проверки гипотезы о равенстве центров распределения двух нормальных генеральных совокупностей при неизвестной дисперсии для каждой пары соседних символов ключевой фразы (по специальному алгоритму);

вычисление вероятности подлинности пользователя как отношения числа сочетаний соседних клавиш, для которых подтверждены гипотезы (пп. 4 и 5), к общему числу сочетаний соседних символов ключевой фразы;

сравнение полученной оценки вероятности с выбранным пороговым значением для принятия решения о допуске пользователя.

Второй вариант процедуры аутентификации:

набор ключевой фразы один раз;

решение задачи проверки гипотезы о равенстве дисперсий двух нормальных генеральных совокупностей для временных интервалов между нажатиями соседних символов ключевой фразы;

если дисперсии равны, то исключение временных интервалов между нажатиями соседних символов ключевой фразы, которые существенно отличаются от эталонных (полученных при настройке);

вычисление вероятности подлинности пользователя как отношения числа оставшихся интервалов к общему числу интервалов в ключевой фразе;

сравнение полученной оценки вероятности с выбранным пороговым значением для принятия решения о допуске пользователя.

Вместо использования постоянной для пользователя КС ключевой фразы можно проводить аутентификацию с помощью набора псевдослучайного текста. В этом случае клавиатура разделяется на поля и вводится понятие расстояния dij между клавишами *i* и *j,* под которым понимается число кла-

виш, расположенных на соединяющей i и *j* прямой линии. Клавиша i принадлежит полю *m,* если У j из *m dij* ≤ *к.*

Величину *к* назовем степенью поля *m* (если *к* = 0, то *m* — отдельная клавиша). Обозначим через x_{ij} временной интервал между нажатиями клавиш, принадлежащих полям i и j.

Введем следующие допущения:

характеристики нажатия клавиш одного поля тем ближе друг к другу, чем меньше к;

для пользователя, работающего двумя руками, получение характеристик клавиатурного почерка возможно с помощью исследования работы только с одной половиной клавиатуры;

ключевой фразой может быть любой набор символов;

число полей должно быть одним и тем же в режимах настройки и аутентификации.

Процедура настройки при наборе псевдослучайного текста:

1) генерация и вывод пользователю текста из фиксированного множества слов, символы которых максимально разбросаны по клавиатуре;

2) набор текста пользователем;

3) фиксация и сохранение значений x_{ij}, которые затем используются для расчета статистических характеристик клавиатурного почерка.

Процедура аутентификации совпадает с процедурой аутентификации, используемой при наборе ключевой фразы.

Достоверность аутентификации на основе клавиатурного почерка пользователя ниже, чем при использовании его биометрических характеристик.

Однако этот способ аутентификации имеет и свои преимущества:

• возможность скрытия факта применения дополнительной аутентификации пользователя, если в качестве ключевой фразы используется вводимая пользователем парольная фраза;

• возможность реализации данного способа только с помощью программных средств (снижение стоимости средств аутентификации).

Теперь рассмотрим способ аутентификации, основанный на росписи мышью (с помощью этого манипулятора, естественно, нельзя выполнить реальную роспись пользователя, поэтому данная роспись будет достаточно простым росчерком). Назовем линией росписи ломаную линию, полученную соединением точек от начала росписи до ее завершения (соседние точки при этом не должны иметь одинаковых координат). Длину линии росписи рассчитаем как сумму длин отрезков, соединяющих точки росписи.

Введем понятие разрыва в линии росписи, признаком которого будет выполнение условия

$$d_{i,i-1} > d/k,$$

где $d_{i,i-1}$ — расстояние между двумя соседними точками линии росписи; d — длина всей линии; $к$ — число точек в линии.

Для устранения разрывов в линии росписи С. П. Расторгуевым предложен алгоритм ее сглаживания, состоящий в добавлении в линию в точках ее разрывов дополнительных точек. Каждая дополнительная точка a с координатами x_a и y_a, добавляемая между точками i-1 и i линии росписи, должна удовлетворять условию

$$\min (d_{i-1,a} + d_{a,i}) \text{ V } a \; d_{i-1,a} \leq 2.$$

По сглаженной линии росписи можно выделить все замкнутые контуры в ней (по специальному алгоритму).

Процедура настройки на характеристики пользователя может состоять из следующих этапов:

1) ввод нескольких эталонных росписей;

2) для каждой росписи получение числа точек в ней и длины ее линии, определение числа и местоположения разрывов в линии росписи;

3) для каждой линии росписи выполнение сглаживания, получение числа и местоположения замкнутых контуров;

4) расчет среднего значения полученных характеристик росписи и их допустимых отклонений.

Процедура аутентификации состоит из следующих этапов:

1) ввод росписи;

2) расчет числа точек и длины линии росписи;

3) получение числа и местоположения разрывов в линии росписи;

4) сглаживание линии росписи;

5) получение числа и местоположения замкнутых контуров;

6) сравнение полученных характеристик росписи с эталонными;

7) принятие решения о допуске пользователя к работе в КС.

Подобно аутентификации на основе клавиатурного почерка подлинность пользователя по его росписи мышью подтверждается прежде всего темпом его работы с этим устройством ввода.

К **достоинствам аутентификации пользователей по их росписи мышью**, подобно использованию клавиатурного почерка, относится возможность реализации этого способа только с помощью программных средств; к **недостаткам** — меньшая достоверность аутентификации по сравнению с применением биометрических характеристик пользователя, а также необходимость достаточно уверенного владения пользователем навыками работы с мышью.

Общей особенностью способов аутентификации, основанных на клавиатурном почерке и росписи мышью является **нестабильность их характеристик** у одного и того же пользователя, которая может быть вызвана:

1) естественными изменениями, связанными с улучшением навыков пользователя по работе с клавиатурой и мышью или, наоборот, с их ухудшением из-за старения организма;

2) изменениями, связанными с ненормальным физическим или эмоциональным состоянием пользователя.

Изменения характеристик пользователя, вызванные причинами первого рода, не являются скачкообразными, поэтому могут быть нейтрализованы изменением эталонных характеристик после каждой успешной аутентификацией пользователя.

Изменения характеристик пользователя, вызванные причинами второго рода, могут быть скачкообразными и привести к отклонению его попытки входа в КС. Однако эта особенность аутентификации на основе клавиатурного почерка и росписи мышью может стать и достоинством, если речь идет о пользователях КС военного, энергетического и финансового назначения.

Перспективным направлением развития способов аутентификации пользователей КС, основанных на их личных особенностях, может стать подтверждение подлинности пользователя на основе его знаний и навыков, характеризующих уровень образования и культуры.

6.5. Двухфакторная аутентификация

Отмеченные недостатки парольной аутентификации пользователей КС могут быть устранены применением так называемой **двухфакторной аутентификации**. В этом случае пользователь для входа в систему должен не только ввести пароль, но и предъявить элемент аппаратного обеспечения, содержащий подтверждающую его подлинность ключевую информацию. Такими элементами аппаратного обеспечения могут быть:

магнитные диски, не требующие установки на компьютере пользователя КС никаких дополнительных аппаратных

средств, но наиболее уязвимые с точки зрения копирования хранящейся на них ключевой информации;

элементы Touch Memory (аналогичные изделия других производителей именуются iButton), включающие в себя энергонезависимую память в виде постоянного запоминающего устройства (ПЗУ) с уникальным для каждого изделия серийным номером и (в более дорогих вариантах) оперативного запоминающего устройства (ОЗУ) для хранения идентифицирующей пользователя информации, а также встроенный элемент питания со сроком службы до 10 лет (элемент Touch Memory напоминает миниатюрную батарейку диаметром 16 мм и толщиной 3...6 мм, он имеет один сигнальный контакт и один контакт заземления, а для контакта элемента с устройством чтения достаточно простого касания);

пластиковые карты с магнитной полосой, на которой помимо ключевой информации могут размещаться и дополнительные реквизиты пользователя (его фамилия, имя, отчество, фотография, название организации и ее подразделения и т.п.); подобные карты наиболее дешевы, но и наименее защищены от копирования и подделки;

карты со штрихкодом, покрытым непрозрачным составом, считывание информации с которых происходит в инфракрасных лучах; эти карты также относительно дешевы, но уязвимы для подделки-;

смарт-карты, носителем ключевой информации в которых является специальная бескорпусная микросхема, включающая в себя только память для хранения ключевой информации (простые смарт-карты) или микропроцессор (интеллектуальные карты), позволяющий реализовывать достаточно сложные процедуры аутентификации;

• маркеры eToken (USB-брелки), представляющие собой подключаемое к USB-порту компьютера устройство, которое включает в себя аналогичную смарт-карте микросхему с процессором и защищенной от несанкционированного доступа памятью (в отличие от пластиковых карт не требуется установ-

ка устройства их чтения с кабелем для подключения этого устройства к компьютеру).

7. МЕТОДЫ И СРЕДСТВА ЗАЩИТЫ ИНФОРМАЦИИ В СЕТИ

7.1. Проблемы информационной безопасности при подключении к глобальной сети

Интенсивное развитие глобальных компьютерных сетей, появление новых технологий поиска-информации привлекают все больше внимания к сети Internet со стороны частных лиц и различных организаций. Многие организации принимают решение об интеграции своих локальных и корпоративных сетей в глобальную сеть. Использование глобальных сетей в коммерческих целях, а также передача информации, содержащей сведения конфиденциального характера, влекут за собой необходимость построения эффективной системы защиты информации. В настоящее время в России глобальные сети применяются для передачи коммерческой информации различного уровня конфиденциальности, например для связи с удаленными офисами из головной штаб-квартиры организации или создания Web-страницы организации с размещенной на ней рекламой и деловыми предложениями.

Вряд ли нужно перечислять все преимущества, которые получает современное предприятие, имея доступ к глобальной сети Internet. Но, как и многие другие новые технологии, использование Internet имеет и негативные последствия. Развитие глобальных сетей привело к многократному увеличению количества пользователей и увеличению количества атак на компьютеры, подключенные к сети Internet. Ежегодные потери, обусловленные недостаточным уровнем защищенности компьютеров, оцениваются десятками миллионов долларов. При подключении к Internet локальной или корпоративной сети необходимо позаботиться об обеспечении информационной безопасности этой сети.

Глобальная сеть Internet создавалась как открытая система, предназначенная для свободного обмена информацией. В силу открытости своей идеологии Internet предоставляет для злоумышленников значительно большие возможности по сравнению с традиционными информационными системами. Через Internet нарушитель может:

вторгнуться во внутреннюю сеть предприятия и получить несанкционированный доступ к конфиденциальной информации;

незаконно скопировать важную и ценную для предприятия информацию;

получить пароли, адреса серверов, а подчас и их содержимое;

входить в информационную систему предприятия под именем зарегистрированного пользователя и т.д.

С помощью полученной злоумышленником информации может быть серьезно подорвана конкурентоспособность предприятия и доверие его клиентов.

Проблемы недостаточной информационной безопасности являются "врожденными" практически для всех протоколов и служб Internet. Большая часть этих проблем связана с исторической зависимостью internet от операционной системы UNIX. Известно, что сеть Arpanet (прародитель Internet) строилась как сеть, связывающая исследовательские центры, научные, военные и правительственные учреждения, крупные университеты США. Эти структуры использовали операционную систему UNIX в качестве платформы для коммуникаций и решения собственных задач. Поэтому особенности методологии программирования в среде UNIX и ее архитектуры наложили отпечаток на реализацию протоколов обмена и политики безопасности в сети. Из-за открытости и распространенности система UNIX стала любимой добычей хакеров. Поэтому совсем не удивительно, что набор протоколов TCP/IP, который обеспечивает коммуникации в глобальной сети Internet и в получающих все большую популярность интрасетях, имеет "врож-

денные" недостатки защиты. То же самое можно сказать и о ряде служб Internet.

Набор протоколов управления передачей сообщений в Internet (Transmission Control Protocol/Internet Protocol - TCP/IP) используется для организации коммуникаций в неоднородной сетевой среде, обеспечивая совместимость между компьютерами разных типов. Совместимость - одно из основных преимуществ TCP/IP, поэтому большинство локальных компьютерных сетей поддерживает эти протоколы. Кроме того, протоколы TCP/IP предоставляют доступ к ресурсам глобальной сети Internet. Поскольку TCP/IP поддерживает маршрутизацию пакетов, он обычно используется в качестве межсетевого протокола. Благодаря своей популярности TCP/IP стал стандартом де-факто для межсетевого взаимодействия.

В заголовках пакетов TCP/IP указывается информация, которая может подвергнуться нападениям хакеров. В частности, хакер может подменить адрес отправителя в своих "вредоносных" пакетах, после чего они будут выглядеть, как пакеты, передаваемые авторизированным клиентом.

Отметим "врожденные слабости" некоторых распространенных служб internet [41].

Простой протокол передачи электронной почты (Simple Mail Transfer Protocol - SMTP) позволяет осуществлять почтовую транспортную службу Internet. Одна из проблем безопасности, связанная с этим протоколом, заключается в том, что пользователь не может проверить адрес отправителя в заголовке сообщения электронной почты. В результате хакер может послать во внутреннюю сеть большое количество почтовых сообщений, что приведет к перегрузке и блокированию работы почтового сервера.

Популярная в Internet *программа электронной почты Sendmail* использует для работы некоторую сетевую информацию - IP-адрес отправителя. Перехватывая сообщения, отправляемые с помощью Sendmail, хакер может употребить эту

информацию для нападений, например для спуфинга (подмены адресов).

Протокол передачи файлов (File Transfer Protocol - FTP) обеспечивает передачу текстовых и двоичных файлов, поэтому его часто используют в Internet для организации совместного доступа к информации. Его обычно рассматривают как один из методов работы с удаленными сетями. На FTP-серверах хранятся документы, программы, графика и другие виды информации. К данным этих файлов на FTP-серверах нельзя обратиться напрямую. Это можно сделать, только переписав их целиком с FTP-сервера на локальный сервер. Некоторые FTP-серверы ограничивают доступ пользователей к своим архивам данных с помощью пароля, другие же предоставляют свободный доступ (так называемый анонимный FTP-сервер). При использовании опции анонимного FTP для своего сервера пользователь должен быть уверен, что на нем хранятся только файлы, предназначенные для свободного распространения.

Служба сетевых имен (Domain Name System - DNS) представляет собой распределенную базу данных, которая преобразует имена пользователей и хост-компьютеров в IP-адреса, указываемые в заголовках пакетов, и наоборот. DNS также хранит информацию о структуре сети компании, например количестве компьютеров с IP-адресами в каждом домене. Одной из проблем DNS является то, что эту базу данных очень трудно "скрыть" от неавторизированных пользователей. В результате DNS часто используется хакерами как источник информации об именах доверенных хост-компьютеров.

*Служба эмуляции удаленного терминала (TELNET) использует*ся для подключения к удаленным системам, присоединенным к сети; применяет базовые возможности по эмуляции терминала. При использовании этого сервиса Internet пользователи должны регистрироваться на сервере TELNET, вводя свои имя и пароль. После аутентификации пользователя его рабочая станция функционирует в режиме "тупого" терминала, подключенного к внешнему хост-компьютеру. С этого

терминала пользователь может вводить команды, которые обеспечивают ему доступ к файлам и запуск программ. Подключившись к серверу TELNET, хакер может сконфигурировать его программу таким образом, чтобы она записывала имена и пароли пользователей.

Всемирная паутина (World Wide Web - WWW) - это система, основанная на сетевых приложениях, которые позволяют пользователям просматривать содержимое различных серверов в internet или интрасетях. Самым полезным свойством WWW является использование гипертекстовых документов, в которые встроены ссылки на другие, документы и Web-узлы, что дает пользователям возможность легко переходить от одного узла к другому. Однако это же свойство является и наиболее слабым местом системы WWW, поскольку ссылки на Web-узлы, хранящиеся в гипертекстовых документах, содержат информацию о том, как осуществляется доступ к соответствующим узлам. Используя эту информацию, хакеры могут разрушить Web-узел или получить доступ к хранящейся в нем конфиденциальной информации.

К уязвимым службам и протоколам Internet относятся также протокол копирования UUCP, протокол маршрутизации RIP, графическая оконная система X Windows и др.

7.2. Межсетевой экран и политика сетевой безопасности

Ряд задач по отражению наиболее вероятных, угроз для внутренних сетей способны решать ***межсетевые экраны***. В отечественной литературе до последнего времени использовались вместо этого термина другие термины иностранного происхождения: брандмауэр и firewall. Вне компьютерной сферы брандмауэром (или firewall) называют стену, сделанную из негорючих материалов и препятствующую распространению пожара. В сфере компьютерных сетей межсетевой экран представляет собой барьер, защищающий от фигурального пожара - попыток злоумышленников вторгнуться во внутреннюю сеть

для того, чтобы скопировать, изменить или стереть информацию либо воспользоваться памятью или вычислительной мощностью работающих в этой сети компьютеров. Межсетевой экран призван обеспечить безопасный доступ к внешней сети и ограничить доступ внешних пользователей к внутренней сети.

Межсетевой экран (МЭ) - это система межсетевой защиты, позволяющая разделить общую сеть на две части или более и реализовать набор правил, определяющих условия прохождения пакетов с данными через границу из одной части общей сети в другую (рис. 29). Как правило, эта граница проводится между корпоративной (локальной) сетью предприятия и глобальной сетью Internet, хотя ее можно провести и внутри корпоративной сети предприятия.

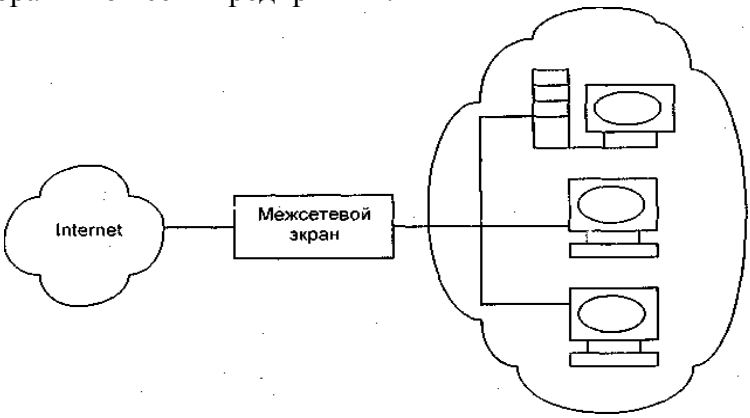


Рис. 29. Схема установления межсетевого экрана

Межсетевой экран (FIRE WALL) - набор программных и аппаратных средств управления доступом одной сети в другую, защищающий внутреннюю сеть от внешний вторжений.

МЭ пропускает через себя весь трафик, принимая для каждого проходящего пакета решение - пропускать его или нет. Для того чтобы МЭ мог осуществить это, ему необходимо определить набор правил фильтрации.

Обычно межсетевые экраны защищают внутреннюю сеть предприятия от "вторжений" из глобальной сети Internet, однако они могут использоваться и для защиты от "нападений" из корпоративной интрасети, к которой подключена локальная сеть предприятия. Ни один межсетевой экран не может гарантировать полной защиты внутренней сети при всех возможных обстоятельствах. Однако для большинства коммерческих организаций установка межсетевого экрана является необходимым условием обеспечения безопасности внутренней сети. Главный довод в пользу применения межсетевого экрана состоит в том, что без него системы внутренней сети подвергаются опасности со стороны слабо защищенных служб сети Internet, а также зондированию и атакам с каких-либо других хост-компьютеров внешней сети.

Решение о том, фильтровать ли с помощью межсетевого экрана конкретные протоколы и адреса, зависит от принятой в защищаемой сети политики безопасности. **Межсетевой экран является набором компонентов**, настраиваемых таким образом, чтобы реализовать выбранную политику безопасности. В частности, необходимо решить, будет ли ограничен доступ пользователей к определенным службам Internet на базе протоколов TCP/IP и если будет, то до какой степени.

Политика сетевой безопасности каждой организации должна включать две составляющие:

политику доступа к сетевым сервисам;

политику реализации межсетевых экранов.

В соответствии с политикой доступа к сетевым сервисам; определяется список сервисов Internet, к которым пользователи должны иметь ограниченный доступ. Задаются также ограничения: на методы доступа, например, на использование протоколов SLIP (Serial Line Internet Protocol) и PPP (Point-to-Point Protocol). Ограничение методов доступа необходимо для того, чтобы пользователи не могли обращаться к "запрещенным" сервисам Internet обходными путями. Например, если для ограничения доступа в Internet сетевой администратор

устанавливает специальный шлюз, который не дает возможности пользователям работать в системе WWW, они могли бы установить PPP-соединения с Web-серверами по коммутируемой линии.

Политика доступа к сетевым сервисам обычно основывается на одном из следующих принципов:

1) запретить доступ из Internet во внутреннюю сеть, но разрешить доступ из внутренней сети в Internet;

2) разрешить ограниченный доступ во внутреннюю сеть из Internet, обеспечивая работу только отдельных "авторизированных" систем, например почтовых серверов.

В соответствии с политикой реализации межсетевых экранов определяются правила доступа к ресурсам внутренней сети. Прежде всего, необходимо установить, насколько "доверительной" или "подозрительной" должна быть система защиты. Иными словами, правила доступа к внутренним ресурсам должны базироваться на одном из следующих принципов:

1) запрещать все, что не разрешено в явной форме;

2) разрешать все, что не запрещено в явной форме.

Реализация межсетевого экрана на основе первого принципа обеспечивает значительную защищенность. Однако правила доступа, сформулированные в соответствии с этим принципом могут доставлять большие неудобства пользователям, а кроме того, их реализация обходится достаточно дорого. При реализации второго принципа внутренняя сеть оказывается менее защищенной от нападений хакеров, однако пользоваться ей будет удобнее и потребуется меньше затрат.

Эффективность защиты внутренней сети с помощью межсетевых экранов зависит не только от выбранной политики доступа к сетевым сервисам и ресурсам внутренней сети, но и от рациональности выбора и использования основных компонентов межсетевого экрана.

Функциональные требования к межсетевым экранам включают:

требования к фильтрации на сетевом уровне;

требования к фильтрации на прикладном уровне;

требования по настройке правил фильтрации и администрированию;

требования к средствам сетевой аутентификации;

требования по внедрению журналов и учету.

7.3. Основные компоненты межсетевых экранов

Большинство компонентов межсетевых экранов можно отнести к одной из трех категорий:

- фильтрующие маршрутизаторы;
- шлюзы сетевого уровня;
- шлюзы прикладного уровня.

Эти категории можно рассматривать как базовые компоненты реальных межсетевых экранов. Лишь немногие межсетевые экраны включают только одну из перечисленных категорий. Тем не менее эти категории отражают ключевые возможности, отличающие межсетевые экраны друг от друга.

Фильтрующие маршрутизаторы. Фильтрующий маршрутизатор представляет собой маршрутизатор или работающую на сервере программу, сконфигурированные таким образом, чтобы фильтровать входящие и исходящие пакеты. Фильтрация пакетов осуществляется на основе информации, содержащейся в TCP- и IP-заголовках пакетов. Процесс инкапсуляции передаваемых данных и формирования TCP- и IP-заголовков пакетов с данными в стеке протоколов TCP/IP показан на рис. 30.

Фильтрующий маршрутизатор обычно может фильтровать IP-пакеты на основе группы следующих полей заголовка пакета:

- IP-адрес отправителя (адрес системы, которая послала пакет);

• IP-адрес получателя (адрес системы, которая принимает пакет);

• порт отправителя (порт соединения в системе-отправителе);

• порт получателя (порт соединения в системе-получателе).

Порт - это программное понятие, которое используется клиентом или сервером для посылки или приема сообщений; порт идентифицируется 16-битовым числом.

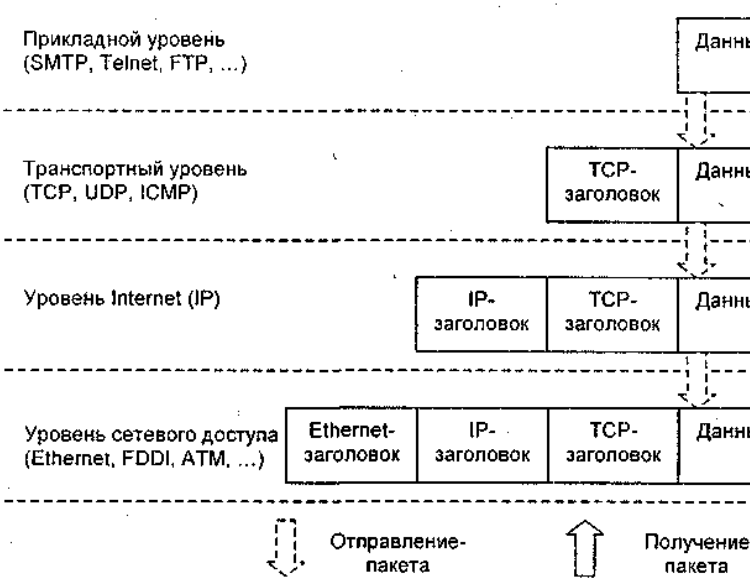


Рис. 30. Схема инкапсуляции данных в стеке протоколов TCP/IP

В настоящее время не все фильтрующие маршрутизаторы фильтруют пакеты по TCP/UDP-порту отправителя, однако многие производители маршрутизаторов начали обеспечивать такую возможность. Некоторые маршрутизаторы проверяют, с

какого сетевого интерфейса маршрутизатора пришел пакет, и затем используют эту информацию как дополнительный критерий фильтрации.

Фильтрация может быть реализована различным образом для блокирования соединений с определенными хост-компьютерами или портами. Например, можно блокировать соединения, идущие от конкретных адресов тех хост-компьютеров и сетей, которые считаются враждебными или ненадежными.

Добавление фильтрации по портам TCP и UDP к фильтрации по IP-адресам обеспечивает большую гибкость. Известно, что такие серверы, как демон TELNET, обычно связаны с конкретными портами (например, порт 23 протокола TELNET). Если межсетевой экран может блокировать соединения TCP или UDP с определенными портами или от них, то можно реализовать политику безопасности, при которой некоторые виды соединений устанавливаются только с конкретными хост-компьютерами.

Например, внутренняя сеть может блокировать все входные соединения со всеми хост-компьютерами за исключением нескольких систем. Для этих систем могут быть разрешены только определенные сервисы (SMTP для одной системы и TELNET или FTP - для другой). При фильтрации по портам TCP и UDP эта политика может быть реализована фильтрующим маршрутизатором или хост-компьютером с возможностью фильтрации пакетов (рис. 31).

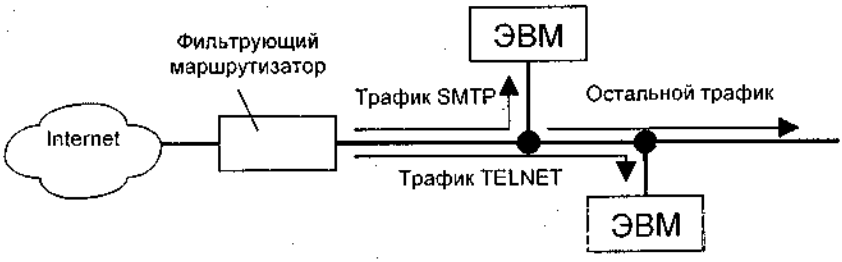


Рис. 31. Схема фильтрации трафика SMTP и TELNET

В качестве примера работы фильтрующего маршрутизатора рассмотрим реализацию политики безопасности, допускающей определенные соединения с внутренней сетью с адресом 123.4.*.* Соединения TELNET разрешаются только с одним хост-компьютером с адресом 123.4.5.6, который может быть прикладным TELNET-шлюзом, а SMTP-соединения - только с двумя хост-компьютерами с адресами 123.4.5.7 и 123.4.5.8, которые могут быть двумя шлюзами электронной почты. Обмен по NNTP (Network News Transfer Protocol) разрешается только от сервера новостей с адресом 129.6.48.254 и только с NNTP-сервером сети с адресом 123.4.5.9, а протокол NTP (сетевого времени) - для всех хост-компьютеров. Все другие серверы и пакеты блокируются. Соответствующий набор правил представлен в таблице.

Правила фильтрации

Тип	Адрес отправителя	Адрес получателя	Порт отправителя	Порт получателя	Действие
TCP	*	123.4.5.6	> 1023	23	Разрешить
TCP	*	123.4.5.7	> 1023	25	Разрешить
TCP	*	123.4.5.8	> 1023	25	Разрешить
TCP	129.6.48.254	123.4.5.9	> 1023	119	Разрешить
UDP	*	123.4.*.*	> 1023	123	Разрешить
*	*	*	*	*	Запретить

Первое правило позволяет пропускать пакеты TCP из сети Internet от любого источника с номером порта большим, чем

1023, к получателю с адресом 123.4.5.6 в порт 23. Порт 23 связан с сервером TELNET, а все клиенты TELNET должны иметь непривилегированные порты с номерами не ниже 1024.

Второе и третье правила работают аналогично и разрешают передачу пакетов к получателям с адресами 123.4.5.7 и 123.4.5.8 в порт 25, используемый SMTP.

Четвертое правило пропускает пакеты к NNTP-серверу сети, но только от отправителя с адресом 129.6.48.254 к получателю с адресом 123.4.5.9 с портом назначения 119 (129.6.48.254 -единственный NNTP-сервер, от которого внутренняя сеть получает новости, поэтому доступ к сети для выполнения протокола NNTP ограничен только этой системой).

Пятое правило разрешает трафик NTP, который использует протокол UDP вместо TCP, от любого источника к любому получателю внутренней сети.

Наконец, шестое правило блокирует все остальные пакеты. Если бы этого правила не было, маршрутизатор мог бы блокировать, а мог бы и не блокировать другие типы пакетов. Выше был рассмотрен очень простой пример фильтрации пакетов. Реально используемые правила позволяют осуществить более сложную фильтрацию и являются более гибкими.

Правила фильтрации пакетов формулируются сложно, и обычно нет средств для тестирования их корректности, кроме медленного ручного тестирования. У некоторых фильтрующих маршрутизаторов нет средств протоколирования, поэтому, если правила фильтрации пакетов все-таки позволят опасным пакетам пройти через маршрутизатор, такие пакеты не смогут быть выявлены до обнаружения последствий проникновения.

Даже если администратору сети удастся создать эффективные правила фильтрации, их возможности остаются ограниченными. Например, администратор задает правило, в соответствии с которым маршрутизатор будет отбраковывать все пакеты с неизвестным адресом отправителя. Однако хакер может использовать в качестве адреса отправителя в своем "вредоносном" пакете реальный адрес доверенного (авторизи-

рованного) клиента. В этом случае фильтрующий маршрутизатор не сумеет отличить поддельный пакет от настоящего и пропустит его. Практика показывает, что подобный вид нападения, называемый *подменой адреса,* довольно широко распространен в сети Internet и часто оказывается эффективным.

Межсетевой экран с фильтрацией пакетов, работающий только на сетевом уровне эталонной модели взаимодействия открытых систем OSI-ISO, обычно проверяет информацию, содержащуюся только в IP-заголовках пакетов. Поэтому обмануть его несложно: хакер создает заголовок, который удовлетворяет разрешающим правилам фильтрации. Кроме заголовка пакета, никакая другая содержащаяся в нем информация межсетевыми экранами данной категории не проверяется.

К **положительным качествам фильтрующих маршрутизаторов следует отнести**:

сравнительно невысокую стоимость;

гибкость в определении правил фильтрации;

небольшую задержку при прохождении пакетов.

Недостатками фильтрующих маршрутизаторов являются:

внутренняя сеть видна (маршрутизируется) из сети Internet;

правила фильтрации пакетов трудны в описании и требуют очень хороших знаний технологий TCP и UDP;

при нарушении работоспособности межсетевого экрана с фильтрацией пакетов все компьютеры за ним становятся полностью незащищенными либо недоступными;

аутентификацию с использованием IP-адреса можно обмануть путем подмены IP-адреса (атакующая система выдает себя за другую, используя ее IP-адрес);

отсутствует аутентификация на пользовательском уровне.

Шлюзы сетевого уровня. Шлюз сетевого уровня иногда называют системой трансляции сетевых адресов или шлюзом

сеансового уровня модели OSI. Такой шлюз исключает прямое взаимодействие между авторизированным клиентом и внешним хост-компьютером.

Шлюз сетевого уровня принимает запрос доверенного клиента на конкретные услуги и после проверки допустимости запрошенного сеанса устанавливает соединение с внешним хост-компьютером. После этого шлюз копирует пакеты в обоих направлениях, не осуществляя их фильтрации.

Шлюз следит за подтверждением (квитированием) связи между авторизированным клиентом и внешним хост-компьютером, определяя, является ли запрашиваемый сеанс связи допустимым. Чтобы выявить допустимость запроса на сеанс связи, шлюз выполняет следующую процедуру.

Когда авторизированный клиент запрашивает некоторый сервис, шлюз принимает этот запрос, проверяя, удовлетворяет ли этот клиент базовым критериям фильтрации (например, может ли DNS-сервер определить IP-адрес клиента и ассоциированное с ним имя). Затем, действуя от имени клиента, шлюз устанавливает соединение с внешним хост-компьютером и следит за выполнением процедуры квитирования связи по протоколу TCP. Эта процедура состоит из обмена TCP-пакетами, которые помечаются флагами SYN (синхронизировать) и ACK (подтвердить) (рис. 32).

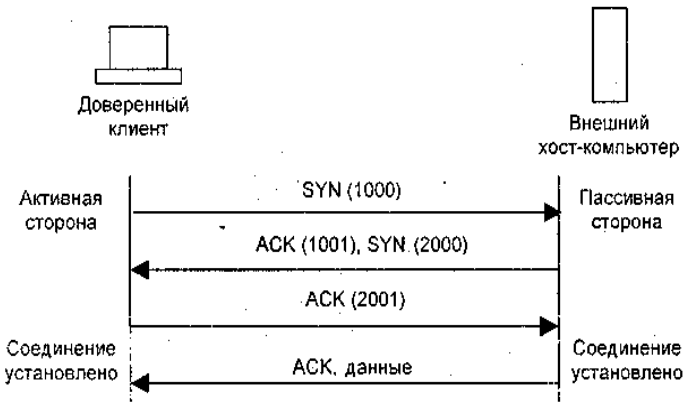


Рис. 32. Последовательность передачи пакетов SYN, ASC в процессе квитирования связи по протоколу TCP

Первый пакет сеанса TCP, помеченный флагом SYN и содержащий произвольное число, например 1000, является запросом клиента на открытие сеанса. Внешний хост-компьютер, получивший этот пакет, посылает в ответ пакет, помеченный флагом АСК и содержащий число, на единицу большее, чем в принятом пакете (в нашем случае 1001), подтверждая тем самым прием пакета SYN от клиента.

Далее осуществляется, обратная процедура: хост-компьютер посылает клиенту пакет SYN с исходным числом (например, 2000), а клиент подтверждает его получение передачей пакета АСК, содержащего число 2001. На этом процесс квитирования связи завершается.

Шлюз сетевого уровня признает запрошенное соединение допустимым только в том случае, если при выполнении процедуры квитирования связи флаги SYN и АСК, а также числа, содержащиеся в TCP-пакетах, оказываются логически связанными между собой.

После того как шлюз определил, что доверенный клиент и внешний хост-компьютер являются авторизированными участниками сеанса TCP, и проверил допустимость этого сеанса, он устанавливает соединение. Начиная с этого момента, шлюз копирует и перенаправляет пакеты туда и обратно, не проводя никакой фильтрации. Он поддерживает таблицу установленных соединений, пропуская данные, относящиеся к одному из сеансов связи, зафиксированных в этой таблице. Когда сеанс завершается, шлюз удаляет соответствующий элемент из таблицы и разрывает цепь, использовавшуюся в данном сеансе.

Для копирования и перенаправления пакетов в шлюзах сетевого уровня применяются специальные приложения, которые называют ***канальными посредниками,*** поскольку они устанавливают между двумя сетями виртуальную цепь или

канал, а затем разрешают пакетам, которые генерируются приложениями TCP/IP, проходить по этому каналу. Канальные посредники поддерживают несколько служб TCP/IP, поэтому шлюзы сетевого уровня могут использоваться для расширения возможностей шлюзов прикладного уровня, работа которых основывается на программах-посредниках конкретных приложений.

Фактически большинство **шлюзов сетевого уровня** не являются самостоятельными продуктами, а поставляются в комплекте со шлюзами прикладного уровня. Примерами таких шлюзов являются **Gauntlet Internet Firewall** компании Trusted Information Systems, **Alta Vista Firewall** компании DEC и **ANS Interlock** компании ANS. Например, Alta Vista Firewall использует канальные посредники прикладного уровня для каждой из шести служб TCP/IP, к которым относятся, в частности, FTP, HTTP (Hyper Text Transport Protocol) и TELNET. Кроме того, межсетевой экран компании DEC обеспечивает шлюз сетевого уровня, поддерживающий другие общедоступные службы TCP/IP, такие как Gopher и SMTP, для которых межсетевой экран не предоставляет посредников прикладного уровня.

Шлюз сетевого уровня выполняет еще одну важную функцию защиты: он используется в качестве *сервера-посредника.* Этот сервер-посредник выполняет *процедуру трансляции адресов,* при которой происходит преобразование внутренних IP-адресов в один "надежный" IP-адрес. Этот адрес ассоциируется с межсетевым экраном, из которого передаются все исходящие пакеты. В результате в сети со шлюзом сетевого уровня все исходящие пакеты оказываются отправленными из этого шлюза, что исключает прямой контакт между внутренней (авторизированной) сетью и потенциально опасной внешней сетью. IP-адрес шлюза сетевого уровня становится единственно активным IP-адресом, который попадает во внешнюю сеть. Таким образом шлюз сетевого уровня и другие серверы-посредники защищают внутренние сети от нападений типа подмены адресов.

После установления связи шлюзы сетевого уровня фильтруют пакеты только на сеансовом уровне модели OSI, т.е. не могут проверять содержимое пакетов, передаваемых между внутренней и внешней сетью на уровне прикладных программ. И поскольку эта передача осуществляется "вслепую", хакер, находящийся во внешней сети, может "протолкнуть" свои "вредоносные" пакеты через такой шлюз. После этого хакер обратится напрямую к внутреннему Web-серверу, который сам по себе не может обеспечивать функции межсетевого экрана. Иными словами, если процедура квитирования связи успешно завершена, шлюз сетевого уровня установит соединение и будет "слепо" копировать и перенаправлять все последующие пакеты независимо от их содержимого.

Чтобы фильтровать пакеты, генерируемые определенными сетевыми службами, в соответствии с их содержимым необходим шлюз прикладного уровня.

Шлюзы прикладного уровня. Для устранения ряда недостатков, присущих фильтрующим маршрутизаторам, межсетевые экраны должны использовать дополнительные программные средства для фильтрации сообщений сервисов типа TELNET и FTP. Такие программные средства называются *полномочными серверами (серверами-посредниками),* а хост-компьютер, на котором они выполняются, - *шлюзом прикладного уровня*.

Шлюз прикладного уровня исключает прямое взаимодействие между авторизированным клиентом и внешним хост-компьютером. Шлюз фильтрует все входящие и исходящие пакеты на прикладном уровне. Связанные с приложениями серверы-посредники перенаправляют через шлюз информацию, генерируемую конкретными серверами.

Для достижения более высокого уровня безопасности и гибкости шлюзы прикладного уровня и фильтрующие маршрутизаторы могут быть объединены в одном межсетевом экране. В качестве примера рассмотрим сеть, в которой с помощью фильтрующего маршрутизатора блокируются входящие соеди-

нения TELNET и FTP. Этот маршрутизатор допускает прохождение пакетов TELNET или FTP только к одному хост-компьютеру - шлюзу прикладного уровня TELNET/FTP. Внешний пользователь, который хочет соединиться с некоторой системой в сети, должен сначала соединиться со шлюзом прикладного уровня, а затем уже с нужным внутренним хост-компьютером. Это осуществляется следующим образом:

1) сначала внешний пользователь устанавливает TELNET-соединение со шлюзом прикладного уровня с помощью протокола TELNET и вводит имя интересующего его внутреннего хост-компьютера;

2) шлюз проверяет IP-адрес отправителя и разрешает или запрещает соединение в соответствии с тем или иным критерием доступа;

3) пользователю может потребоваться аутентификация (возможно, с помощью одноразовых паролей);

4) сервер-посредник устанавливает TELNET-соединение между шлюзом и внутренним хост-компьютером;

5) сервер-посредник осуществляет передачу информации между этими двумя соединениями;

6) шлюз прикладного уровня регистрирует соединение.

Этот пример наглядно показывает преимущества использования полномочных серверов-посредников.

Полномочные серверы-посредники пропускают только те службы, которые им поручено обслуживать. Иначе говоря, если шлюз прикладного уровня наделен полномочиями (и полномочными серверами-посредниками) для служб FTP и TELNET, то в защищаемой сети будут разрешены только FTP и TELNET, а все другие службы будут полностью блокированы. Для некоторых организаций такой вид безопасности имеет большое значение, так как он гарантирует, что через межсетевой экран будут пропускаться только те службы, которые считаются безопасными.

Полномочные серверы-посредники обеспечивают возможность фильтрации протокола. Например, некоторые межсетевые экраны, использующие шлюзы прикладного уровня, могут фильтровать FTP-соединения и запрещать использование команды FTP *put,* что гарантированно не позволяет пользователям записывать информацию на анонимный FTP-сервер.

В дополнение к фильтрации пакетов многие шлюзы прикладного уровня регистрируют все выполняемые сервером действия и, что особенно важно, предупреждают сетевого администратора о возможных нарушениях защиты. Например, при попытках проникновения в сеть извне **BorderWare Firewall Server** компании Secure Computing позволяет фиксировать адреса отправителя и получателя пакетов, время, в которое эти попытки были предприняты, и используемый протокол. Межсетевой экран **Black Hole** компании Milkyway Networks регистрирует все действия сервера и предупреждает администратора о возможных нарушениях, посылая ему сообщение по электронной почте или на пейджер. Аналогичные функции выполняют и ряд других шлюзов прикладного уровня.

Шлюзы прикладного уровня позволяют обеспечить наиболее высокий уровень защиты, поскольку взаимодействие с внешним миром реализуется через небольшое число прикладных полномочных программ-посредников, полностью контролирующих весь входящий и выходящий трафик.

Шлюзы прикладного уровня имеют ряд серьезных преимуществ по сравнению с обычным режимом, при котором прикладной трафик пропускается непосредственно к внутренним хост-компьютерам. Перечислим эти преимущества.

Невидимость структуры защищаемой сети из глобальной сети Internet. Имена внутренних систем можно не сообщать внешним системам через DNS, поскольку шлюз прикладного уровня может быть единственным хост-компьютером, имя которого должно быть известно внешним системам.

Надежная аутентификация и регистрация. Прикладной трафик может быть аутентифицирован, прежде чем он достигнет внутренних хост-**компьютеров**, и может быть зарегистрирован более эффективно, чем с помощью стандартной регистрации.

Оптимальное соотношение между ценой и эффективностью. Дополнительные программные или аппаратные средства для аутентификации или регистрации нужно устанавливать только на шлюзе прикладного уровня.

Простые правила фильтрации. Правила на фильтрующем маршрутизаторе оказываются менее сложными, чем они были бы, если бы маршрутизатор сам фильтровал прикладной трафик и отправлял его большому числу внутренних систем. Маршрутизатор должен пропускать прикладной трафик, предназначенный только для шлюза прикладного уровня, и блокировать весь остальной трафик.

Возможность организации большого числа проверок. Защита на уровне приложений позволяет осуществлять большое количество дополнительных проверок, что снижает вероятность взлома с использованием "дыр" в программном обеспечении.

К **недостаткам шлюзов прикладного уровня** относятся:

более низкая производительность по сравнению с фильтрующими маршрутизаторами; в частности, при использовании клиент-серверных протоколов, таких как TELNET, требуется двух шаговая процедура для входных и выходных соединений;

более высокая стоимость по сравнению с фильтрующими маршрутизаторами.

Помимо TELNET и FTP шлюзы прикладного уровня обычно используются для электронной почты, X Windows и некоторых других служб.

7.4. Основные схемы сетевой защиты на базе межсетевых экранов

При подключении корпоративной или локальной сети к глобальным сетям администратор сетевой безопасности должен решать следующие задачи:

• защита корпоративной или локальной сети от несанкционированного удаленного доступа со стороны глобальной сети;

• скрытие информации о структуре сети и ее компонентов от пользователей глобальной сети;

• разграничение доступа в защищаемую сеть из глобальной сети и из защищаемой сети в глобальную сеть.

Необходимость работы с удаленными пользователями требует установления жестких ограничений доступа к информационным ресурсам защищаемой сети. При этом часто возникает потребность в организации в составе корпоративной сети нескольких сегментов с разными уровнями защищенности:

свободно доступные сегменты (например, рекламный WWW-сервер),

сегмент с ограниченным доступом (например, для доступа сотрудникам организации с удаленных узлов),

• закрытые сегменты (например, финансовая локальная сеть организации).

Для защиты корпоративной или локальной сети применяются следующие основные схемы организации межсетевых экранов:

межсетевой экран - фильтрующий маршрутизатор;

межсетевой экран на основе двухпортового шлюза;

межсетевой экран на основе экранированного шлюза;

межсетевой экран - экранированная подсеть

Межсетевой экран - фильтрующий маршрутизатор.

Межсетевой экран, основанный на фильтрации пакетов, является самым распространенным и наиболее простым в реализации. Он состоит из фильтрующего маршрутизатора, расположенного между защищаемой сетью и сетью internet (рис. 33). Фильтрующий маршрутизатор сконфигурирован для блокирования или фильтрации входящих и исходящих пакетов на основе анализа их адресов и портов.

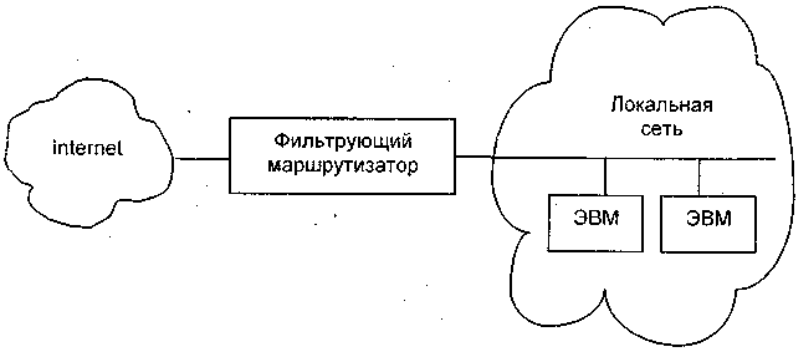


Рис. 33. Межсетевой экран на основе фильтрующего маршрутизатора

Компьютеры, находящиеся в защищаемой сети, имеют прямой доступ в сеть Internet, в то время как большая часть доступа к ним из Internet блокируется. Часто блокируются такие опасные службы, как X Windows, NIS и NFS. В принципе фильтрующий маршрутизатор может реализовать любую из политик безопасности, описанных ранее. Однако если маршрутизатор не фильтрует пакеты по порту источника и номеру входного и выходного порта, то реализация политики "запрещено все, что не разрешено в явной форме" может быть затруднена.

Межсетевые экраны, основанные на фильтрации пакетов, имеют такие же недостатки, что и фильтрующие маршрутизаторы, причем эти недостатки становятся более ощутимыми при ужесточении требований к безопасности защищаемой сети. Отметим некоторые из них:

сложность правил фильтрации; в некоторых случаях совокупность этих правил может стать неуправляемой;

невозможность полного тестирования правил фильтрации; это приводит к незащищенности сети от непротестированных атак; практически отсутствующие возможности регистрации событий; в результате администратору трудно определить, подвергался ли маршрутизатор атаке и скомпрометирован ли он;

каждый хост-компьютер, связанный с сетью Internet, нуждается в своих средствах усиленной аутентификации.

Межсетевой экран на основе двухпортового шлюза.

Межсетевой экран на базе двухпортового прикладного шлюза включает двудомный хост-компьютер с двумя сетевыми интерфейсами. При передаче информации между этими интерфейсами и осуществляется основная фильтрация. Для обеспечения дополнительной защиты между прикладным шлюзом и сетью Internet обычно размещают фильтрующий маршрутизатор (рис. 34). В результате между прикладным шлюзом и маршрутизатором образуется внутренняя экранированная подсеть. Эту подсеть можно использовать для размещения доступных извне информационных серверов.

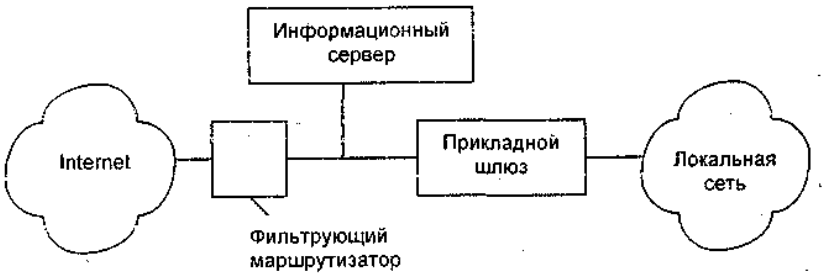


Рис.34. Межсетевой экран с прикладным шлюзом и фильтрующим маршрутизатором

В отличие от схемы межсетевого экрана с фильтрующим маршрутизатором прикладной шлюз полностью блокирует трафик IP между сетью internet и защищаемой сетью. Только полномочные серверы-посредники, располагаемые на прикладном шлюзе, могут предоставлять услуги и доступ пользователям.

Данный вариант межсетевого экрана реализует политику безопасности, основанную на принципе "запрещено все, что не разрешено в явной форме", при этом пользователю недоступны все службы, кроме тех, для которых определены соответствующие полномочия. Такой подход обеспечивает высокий уровень безопасности, поскольку маршруты к защищенной подсети известны только межсетевому экрану и скрыты от внешних систем.

Рассматриваемая схема организации межсетевого экрана является довольно простой и достаточно эффективной.

Следует отметить, что безопасность двудомного хост компьютера, используемого в качестве прикладного шлюза, должна поддерживаться на высоком уровне. Любая брешь в его защите может серьезно ослабить безопасность защищаемой сети. Если шлюз окажется скомпрометированным, у злоумышленника появится возможность проникнуть в защищаемую сеть.

Этот межсетевой экран может требовать от пользователей применения средств усиленной аутентификации, а также регистрации доступа, попыток зондирования и атак системы нарушителем.

Для некоторых сетей может оказаться неприемлемой недостаточная гибкость схемы межсетевого экрана с прикладным шлюзом.

Межсетевой экран на основе экранированного шлюза.

Межсетевой экран на основе экранированного шлюза объединяет фильтрующий маршрутизатор и прикладной шлюз, размещаемый со стороны внутренней сети. Прикладной шлюз реализуется на хост-компьютере и имеет только один сетевой интерфейс (рис. 35).

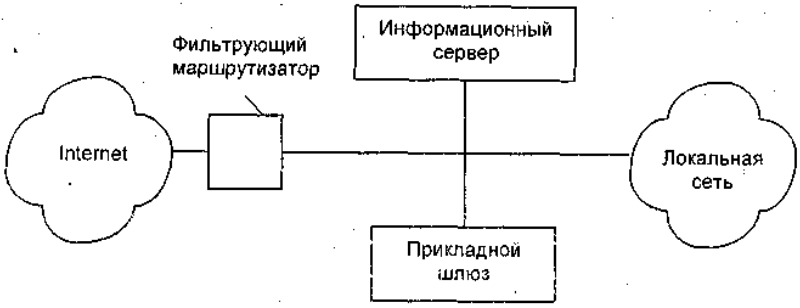


Рис. 35. Межсетевой экран с экранированным шлюзом

В этой схеме первичная безопасность обеспечивается фильтрующим маршрутизатором.. Пакетная фильтрация в фильтрующем маршрутизаторе может быть реализована одним из следующих способов:

позволять внутренним хост-компьютерам открывать соединения с хост-компьютерами, в сети Internet для определенных сервисов (разрешая доступ к ним средствами пакетной фильтрации);

запрещать все соединения от внутренних хост-компьютеров (заставляя их использовать полномочные серверы-посредники на прикладном шлюзе).

Эти подходы можно комбинировать для различных сервисов, разрешая некоторым сервисам соединение непосредственно через пакетную фильтрацию, в то время как другим только непрямое соединение через полномочные серверы-посредники. Все зависит от конкретной политики безопасности, принятой во внутренней сети. В частности, пакетная фильтрация на фильтрующем маршрутизаторе может быть

организована таким образом, чтобы прикладной шлюз, используя свои полномочные серверы-посредники, обеспечивал для систем защищаемой сети такие сервисы, как TELNET, FTP, SMTP.

Межсетевой экран, выполненный по данной схеме, получается более гибким, но менее безопасным по сравнению с межсетевым экраном с прикладным шлюзом на базе двудомного хост-компьютера. Это обусловлено тем, что в схеме межсетевого экрана с экранированным шлюзом существует потенциальная возможность передачи трафика в обход прикладного шлюза непосредственно к системам локальной сети.

Основной недостаток схемы межсетевого экрана с экранированным шлюзом заключается в том, что если атакующий нарушитель сумеет проникнуть в хост-компьютер, то перед ним окажутся незащищенные системы внутренней сети. Другой недостаток связан с возможной компрометацией маршрутизатора. Если маршрутизатор окажется скомпрометированным, внутренняя сеть станет доступна атакующему нарушителю.

По этим причинам в настоящее время все более популярной становится схема межсетевого экрана с экранированной подсетью.

Межсетевой экран - экранированная подсеть. Межсетевой экран, состоящий из экранированной подсети, представляет собой развитие схемы межсетевого экрана на основе экранированного шлюза. Для создания экранированной подсети используются два экранирующих маршрутизатора (рис. 36). Внешний маршрутизатор располагается между сетью internet и экранируемой подсетью, а внутренний - между экранируемой подсетью и защищаемой внутренней сетью. Экранируемая подсеть содержит прикладной шлюз, а также может включать информационные серверы и другие системы, требующие контролируемого доступа. Эта схема межсетевого экрана обеспечивает хорошую безопасность благодаря организации экра-

нированной подсети, которая еще лучше изолирует внутреннюю защищаемую сеть от Internet.

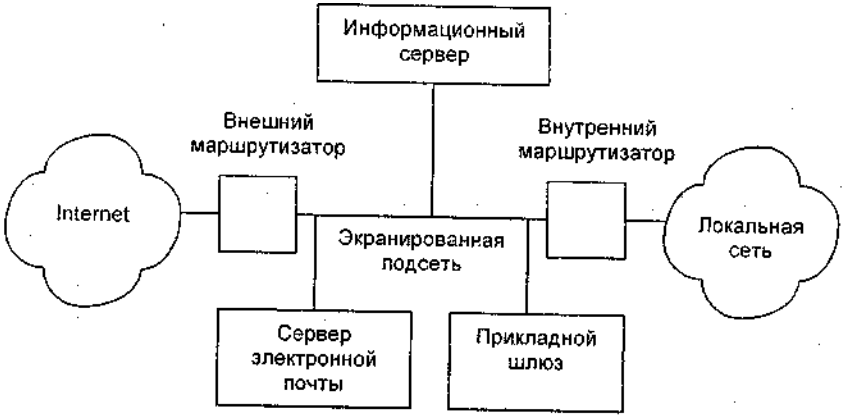


Рис. 36. Межсетевой экран - экранированная подсеть

Внешний маршрутизатор защищает от сети Internet как экранированную подсеть, так и внутреннюю сеть. Он должен пересылать трафик согласно следующим правилам:

разрешается трафик от объектов Internet к прикладному шлюзу;

разрешается трафик от прикладного шлюза к Internet;

разрешается трафик электронной почты от Internet к серверу электронной почты;

разрешается трафик электронной почты от сервера электронной почты к Internet;

разрешается трафик FTP, Gopher и т.д. от Internet к информационному серверу;

запрещается остальной трафик.

Внешний маршрутизатор запрещает доступ из Internet к системам внутренней сети и блокирует весь трафик к Internet, идущий от систем, которые не должны являться инициаторами соединений (в частности, информационный сервер и др:). Этот маршрутизатор может быть использован также для блокирова-

ния других уязвимых протоколов, которые не должны передаваться к хост-компьютерам внутренней сети или от них [75].

Внутренний маршрутизатор защищает внутреннюю сеть как от Internet, так и от экранированной подсети (в случае ее компрометации). Внутренний маршрутизатор осуществляет большую часть пакетной фильтрации. Он управляет трафиком к системам внутренней сети и от них в соответствии со следующими правилами:

разрешается трафик от прикладного шлюза к системам сети;

разрешается прикладной трафик от систем сети к прикладному шлюзу;

разрешается трафик электронной почты от сервера электронной почты- к системам сети;

разрешается трафик электронной почты от систем сети к серверу электронной почты;

разрешается трафик FTP, Gopher и т.д. от систем сети к информационному серверу;

запрещается остальной трафик.

Чтобы проникнуть во внутреннюю сеть при такой схеме межсетевого экрана, атакующему нужно пройти два фильтрующих маршрутизатора. Даже если атакующий каким-то образом проник в хост-компьютер прикладного шлюза, он должен еще преодолеть внутренний фильтрующий маршрутизатор. Таким образом, ни одна система внутренней сети не достижима непосредственно из internet, и наоборот. Кроме того, четкое разделение функций между маршрутизаторами и прикладным шлюзом позволяет достигнуть более высокой пропускной способности.

Прикладной шлюз может включать программы усиленной аутентификации.

Межсетевой экран с экранированной подсетью хорошо подходит для защиты сетей с большими объемами трафика или с высокими скоростями обмена.

Межсетевой экран с экранированной подсетью имеет и недостатки:

• пара фильтрующих маршрутизаторов нуждается в большом внимании для обеспечения необходимого уровня безопасности, поскольку из-за ошибок при их конфигурировании могут возникнуть провалы в безопасности всей сети;

• существует принципиальная возможность доступа в обход прикладного шлюза.

7.5. Защищенные сетевые протоколы

К программным методам защиты в сети Internet могут быть отнесены защищенные криптопротоколы, которые позволяют надежно защищать соединения. В процессе развития Internet были созданы различные защищенные сетевые протоколы, использующие как симметричную криптографию с закрытым ключом, так и асимметричную криптографию с открытым ключом. К основным на сегодняшний день подходам и протоколам, обеспечивающим защиту соединений, относятся SKIP-технология и протокол защиты соединения SSL.

SKIP (Secure Key Internet Protocol) - технологией называется стандарт защиты трафика IP-пакетов, позволяющий на сетевом уровне обеспечить защиту соединения и передаваемых по нему данных.

Возможны два способа реализаций SKIP-защиты трафика IP-пакетов:

шифрование блока данных IP-пакета;

инкапсуляция IP-пакета в SKIP-пакет.

Шифрование блока данных IP-пакета иллюстрируется рис. 37. В этом случае шифруются методом симметричной криптографии только данные IP-пакета, а его заголовок, содержащий помимо прочего адреса отправителя и получателя, остается открытым, и пакет маршрутизируется в соответствии с истинными адресами.

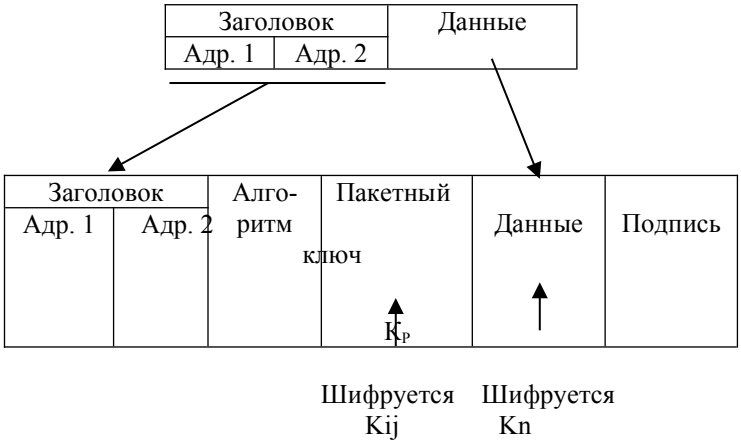


Рис. 37. Схема шифрования блока данных IP-пакетов

Закрытый ключ K_{ij} разделяемый парой узлов сети I и J, вычисляется по схеме Диффи-Хеллмана.

Инкапсуляция IP-пакета в SKIP-пакет показана на рис. 38.

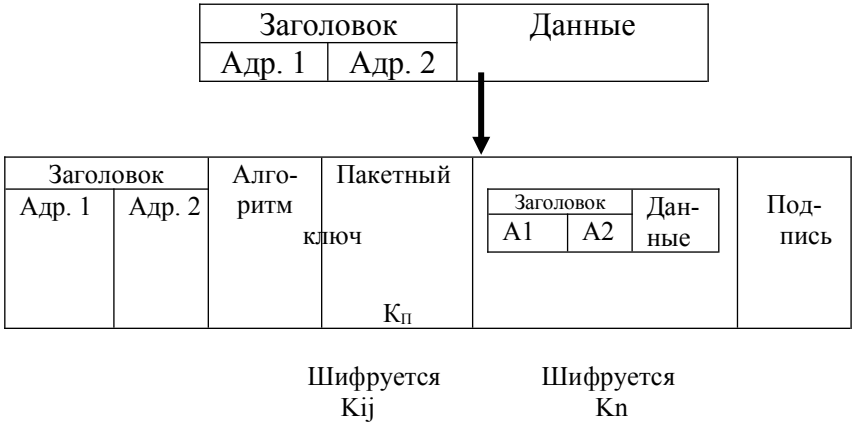


Рис. 38. Схема инкапсуляции IP-пакетов

SKIP-пакет внешне похож на обычный IP-пакет. В поле данных SKIP-пакета полностью размещается в зашифрованном виде исходный IP-пакет. В этом случае в новом заголовке вместо истинных адресов могут быть помещены некоторые другие адреса. Такая структура SKIP-пакета позволяет беспрепятственно направлять его любому хост-компьютеру в сети Internet, при этом межсетевая адресация осуществляется по обычному IP-заголовку в SKIP-пакете. Конечный получатель SKIP-пакета по заранее определенному разработчиками алгоритму расшифровывает криптограмму и формирует обычный TCP- или UDP-пакет, который и передает соответствующему модулю (TCP или UDP) ядра операционной системы.

Универсальный протокол защиты соединения SSL (Secure Socket Layer) функционирует на сеансовом уровне эталонной модели OSI. Протокол SSL, разработанный компанией Netscape, использует криптографию с открытым ключом. Этот протокол является действительно универсальным средством, позволяющим динамически защищать соединение при использовании любого прикладного протокола (FTP, TELNET, SMTP, DNS и т.д.). Протокол SSL поддерживают такие ведущие компании, как IBM, Digital Equipment Corporation, Microsoft Corporation, Motorola, Novell Inc., Sun Microsystems, MasterCard International Inc. и др.

Существует функционально законченный отечественный криптографический комплекс "Шифратор IP потоков", разработанный московским отделением Пензенского научно-исследовательского электротехнического института. Криптографический комплекс "Шифратор IP потоков" представляет собой распределенную систему криптографических шифраторов, средств управления криптографическими шифраторами, средств хранения, распространения и передачи криптографической информации, а также средств оперативного мониторинга и регистрации происходящих событий. Криптографический комплекс "Шифратор IP потоков" предназначен для выполнения следующих функций:

обеспечения конфиденциальности и целостности информации, передаваемой в сетях общего пользования (Internet), построенных на основе протоколов IP;

создания защищенных подсетей передачи конфиденциальной информации;

объединения локальных сетей в единую защищенную сеть;

закрытия доступа к ресурсам локальной сети или отдельным компьютерам из сети общего доступа;

организации единого центра управления защищенной подсетью.

Комплекс обеспечивает:

закрытие передаваемых данных на основе использования функций шифрования в соответствии с отечественным стандартом ГОСТ 28147-89;

контроль целостности передаваемой информации;

аутентификацию абонентов (узлов сети);

защиту доступа к локальной сети и сокрытие IP адресов подсети;

передачу контрольной информации в Центр управления ключевой системой защищенной IP сети;

поддержку протоколов маршрутизации PIP II, OSPF, BGP;

фильтрацию IP, I'CMP и TCP-соединений на этапе маршрутизации и при приеме/передаче в канал связи;

поддержку инкапсуляции IPX в IP (в соответствии с RFC-1234);

поддержку инкапсуляции IP в X.25 и Frame Relay;

защиту от НСД ресурсов самого шифратора.

Криптографический комплекс "Шифратор IP потоков" имеет модульную структуру и состоит из распределенной сети шифраторов IP потоков и единого центра управления ключевой системой.

Шифратор IP протоколов (ШИП) состоит из:

криптографического модуля, непосредственно встроенного в ядро операционной системы;

модуля поддержки клиентской части ключевой системы;

модуля записи протоколов работы криптографической системы;

модуля проверки целостности системы при загрузке.

ШИП содержит также плату с интерфейсом ISA, используемую для защиты от НСД при загрузке системы и для получения от сертифицированного физического датчика случайных чисел, необходимых для реализации процедуры шифрования.

Центр управления ключевой системой (ЦУКС) состоит из:

• автоматизированного рабочего места управления ключевой системой, работающего в среде X Windows;

• модуля серверной части ключевой системы;

• сервисной программы просмотра протоколов работы криптографического комплекса "Шифратор IP потоков".

Управление ключами выполняется при помощи ЦУКС и заключается в следующем:

• периодическая (плановая) смена парных ключей шифрования зарегистрированных узлов защищенной сети;

• формирование и рассылка по сети справочников соответствия, определяющих возможность абонентов работать друг с другом;

• сбор и хранение в базе данных информации о всех критичных событиях в сети, возникающих как при аутентификации абонентов, так и при передаче между ними зашифрованной информации.

В случае возникновения нештатных ситуаций, создающих угрозу нарушения защиты информации, администратор ЦУКС предпринимает действия, направленные на восстановление целостности системы защиты информации.

Схема организации виртуальной корпоративной сети с применением криптографического комплекса "Шифратор IP потоков" показана на рис. 39.

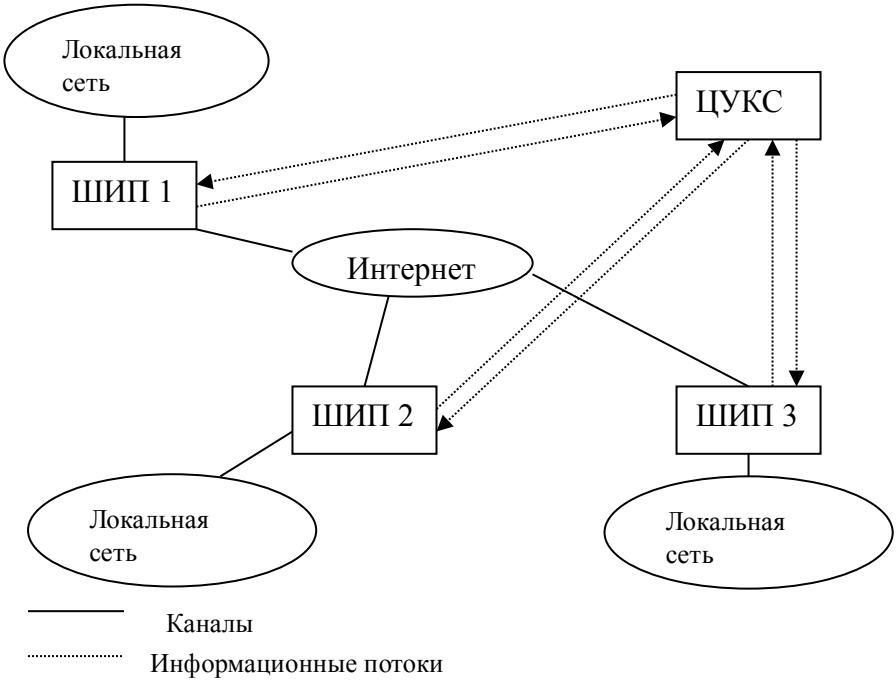


Рис. 39. Виртуальная корпоративная сеть с применением криптографического комплекса "Шифратор IP потоков"

При организации виртуальной корпоративной сети небольшого размера без жестких требований к времени оповещения абонентов о компрометации какого-либо абонента и без жестких требований к полноте собираемых протоколов об ошибках доступа возможно использование одного ЦУКС. При

организации виртуальной корпоративной сети среднего размера или с жесткими требованиями к времени оповещения абонентов о компрометации какого-либо абонента и к полноте собираемых протоколов об ошибках доступа следует использовать несколько ЦУКС. При этом желательно, чтобы ЦУКС имели независимые друг от друга каналы подключения к глобальной сети.

8. ЗАЩИТА КОМПЬЮТЕРНЫХ СИСТЕМ ОТ ВРЕДОНОСНЫХ ПРОГРАММ, НЕСАНКЦИОНИРОВАННОГО ИСПОЛЬЗОВАНИЯ И КОПИРОВАНИЯ

8.1. Вредоносные программы и их классификация

К вредоносным программам относятся компьютерные вирусы и программные закладки.

Компьютерным вирусом называют автономно функционирующую программу, обладающую одновременно тремя свойствами:

способностью к включению своего кода в тела других файлов и системных областей памяти компьютера;

последующему самостоятельному выполнению;

самостоятельному распространению в компьютерных системах.

Программной закладкой называют внешнюю или внутреннюю по отношению к атакуемой компьютерной системе программу, обладающую определенными разрушительными функциями по отношению к этой системе.

Разрушительные функции могут быть следующими:

уничтожение или внесение изменений в функционирование программного обеспечения КС, уничтожение или изменение обрабатываемых в ней данных после выполнения некоторого условия или получения некоторого сообщения извне КС;

превышение полномочий пользователя с целью несанкционированного копирования конфиденциальной информации других пользователей КС или создание условий для такого копирования;

подмена отдельных функций подсистемы защиты КС или создание люков в ней для реализации угроз безопасности информации в КС (например, подмена средств шифрования путем эмуляции работы установленной в КС платы аппаратного шифрования);

перехват паролей пользователей КС с помощью имитации приглашения к его вводу или перехват всего ввода пользователя с клавиатуры;

перехват потока информации, передаваемой между объектами распределенной КС ;

распространение в распределенных КС с целью реализации той или иной угрозы безопасности информации .

Компьютерные вирусы классифицируются по следующим признакам.

1. По способу распространения в КС:

а) файловые вирусы, заражающие файлы одного или нескольких типов;

б) загрузочные вирусы, заражающие загрузочные сектора дисков;

в) комбинированные вирусы, способные заражать и файлы, и загрузочные сектора дисков.

2. По способу заражения других объектов КС:

а) резидентные вирусы, часть кода которых постоянно находится в оперативной памяти компьютера и заражает другие объекты КС;

б) нерезидентные вирусы, которые заражают другие объекты КС в момент открытия уже зараженных ими объектов.

3. По деструктивным возможностям:

а) безвредные вирусы, созданные в целях обучения, однако снижающие эффективность работы КС за счет потребле-

ния ее ресурсов (времени работы центрального процессора, оперативной и внешней памяти и др.);

б) неопасные вирусы, создающие различные звуковые и видеоэффекты;

в) опасные и очень опасные вирусы, вызывающие сбои в работе программного и (или) аппаратного обеспечения компьютера, потерю программ и данных, а потенциально – вывод из строя аппаратуры КС и нанесение вреда здоровью пользователей (с помощью, например, эффекта двадцать пятого кадра).

4. По особенностям реализуемого алгоритма:

а) вирусы-спутники, создающие для заражаемых файлов одноименные фалы с кодом вируса и переименовывающие исходные файлы (при открытии зараженного файла фактически открывается файл с кодом вируса, в котором после выполнения предусмотренных автором действий открывается исходный файл);

б) паразитические вирусы, которые обязательно изменяют содержимое заражаемых объектов;

в) вирусы-невидимки («стелс» - вирусы), в которых путем перехвата обращений операционной системы к зараженным объектам и возврата вместо них оригинальных незараженных данных скрывается факт присутствия вируса в КС (при собственном обращении к дисковой памяти вирусы-невидимки также используют нестандартные средства для обхода средств антивирусной защиты);

г) вирусы-призраки (полиморфные вирусы), каждая следующая копия которых в зараженных объектах отличается от предыдущих (не содержит одинаковых цепочек команд за счет применения шифрования на различных ключах базового кода вируса).

5. По наличию дополнительных возможностей:

а) по обработке атрибута «только чтение» заражаемых файлов;

б) сохранению времени последнего изменения зараженного файла;

в) обработке прерывания, вызванного неисправимой ошибкой устройства ввода-вывода (например, для подавления сообщения операционной системы об ошибке при попытке заражения объекта на защищенном от записи устройстве или объекта, доступ к которому по записи для текущего пользователя запрещен; вывод подобного сообщения может обнаружить присутствие вируса в КС);

г) распространению в КС не только при открытии уже зараженного объекта, но и при выполнении любой операции с ним.

Для отнесения конкретной программы к разряду программных закладок достаточно, чтобы данная программа обладала хотя бы одним из следующих свойств:

скрытие признаков своего присутствия в КС;

реализация самодублирования, перенос своего кода в не занимаемые ранее области оперативной или внешней памяти;

искажение кода других программ в оперативной памяти компьютера;

сохранение данных, размещенных в оперативной памяти, в других ее областях;

искажение, блокировка, подмена сохраняемых (передаваемых) данных, полученных в результате работы других программ или уже находящихся во внешней памяти.

В соответствии с методами внедрения программных закладок в КС и возможным местам их размещения в системе **закладки могут быть разделены на следующие группы**:

программные закладки, ассоциированные с BIOS;

закладки, ассоциированные с программами начальной загрузки и загрузки операционной системы;

закладки, ассоциированные с драйверами операционной системы и другими системными модулями;

закладки, ассоциированные с прикладным программным обеспечением общего назначения (например, архиваторами);

программные файлы, содержащие только код закладки и внедряемые с помощью пакетных командных файлов;

закладки, маскируемые под прикладное программное обеспечение общего назначения;

закладки, маскируемые под игровое и образовательное программное обеспечение (для облегчения их первоначального внедрения в КС).

8.2. Методы обнаружения и удаления вирусов

Прежде всего существуют организационные меры обнаружения и удаления компьютерных вирусов.

Основными каналами распространения компьютерных вирусов являются следующие:

электронная почта, сообщения которой могут содержать зараженные присоединенные файлы;

телеконференции и электронные доски объявлений в сети Интернет;

свободное и условно свободное программное обеспечение, размещенное на общедоступных узлах сети Интернет и случайно или намеренно зараженное вирусами;

размещенные на общедоступных узлах сети Интернет информационные ресурсы, содержащие ссылки на зараженные файлы с элементами управления Active–X;

локальные компьютерные сети организаций, создающие удобную среду для заражения вирусами объектов на других рабочих станциях и серверах;

обмен зараженными файлами на дискетах или дисках между пользователями КС;

использование нелицензионных дисков с программным обеспечением и другими информационными ресурсами.

Для предупреждения вирусного заражения локальной сети организации или компьютера отдельного пользователя необходимо максимально перекрыть возможность проникновения вирусов с использованием перечисленных каналов.

В частности могут использоваться **следующие профилактические меры**:

физическое или логическое (для отдельных учетных записей) отключение накопителей на гибких магнитных дисках и компакт-дисках;

разграничение прав отдельных пользователей и групп на доступ к папкам и файлам операционной системы и других пользователей;

ограничение времени работы в КС привилегированных пользователей (для выполнения действий в КС, не требующих дополнительных полномочий, администраторы должны использовать вторую учетную запись с обычными привилегиями);

использование, как правило, только лицензионного программного обеспечения, приобретенного у официальных представителей фирм-правообладателей;

выделение не подсоединенного к локальной сети компьютера для тестирования полученного из ненадежных источников программного обеспечения и т.д.

В качестве профилактической меры предупреждения заражения файлов пользователей макровирусами в программах пакета MS Office предусмотрена встроенная защита от потенциально опасных макросов. Эта защита устанавливается (в пакете MS Office XP) с помощью команды меню Сервис / Параметры / Безопасность и кнопки «Защита от макросов» или команды меню Сервис / Макрос / Безопасность.

Возможет выбор одного из трех уровней защиты:

высокая безопасность, при установке которой будет разрешено выполнение только макросов, снабженных ЭЦП и полученных из надежных источников, список которых содержится на вкладке «Надежные источники» окна выбора уровня безопасности (макросы без ЭЦП будут автоматически отключаться);

средняя безопасность, при выборе которой при открытии содержащего макросы документа решение об отключении этих макросов будет приниматься самим пользователем;

низкая безопасность, при установке которой все макросы в открываемом документе будут выполняться (корпорация Microsoft рекомендует устанавливать данный уровень безопасности только при наличии антивирусных программ на компьютере пользователя и полной уверенности в безопасности открываемых документов).

Для получения подписи под макросами документа MS Office необходимо открыть окно системы программирования Microsoft Visual Basic с помощью команды меню любой из программ этого пакета Сервис / Макрос / Макросы, выбрать имя макроса и нажать кнопку «Изменить». В окне системы программирования затем выполняется команда Tools | Digital Signature и в появившемся окне цифровой подписи выбирается сертификат открытого ключа ЭЦП, который в дальнейшем будет использован для проверки подписи.

В качестве дополнительной меры защиты можно отменить автоматическое выполнение макросов, полученных из надежных источников. Если снять флажок «Доверять всем установленным надстройкам и шаблонам», то вывод предупреждения о наличии макросов в открываемых документах будет производиться и для макросов, находящихся в уже установленных на компьютере пользователя шаблонах и надстройках MS Office.

Установка защиты от потенциально опасных макросов не позволяет отделить макросы, расширяющие функциональность приложений MS Office, от макросов, содержащих вирусы. Кроме того, некоторые из макровирусов, получив однажды управление, могут понизить уровень безопасности до самого низкого и тем самым блокировать встроенную защиту от макросов.

Для защиты от несанкционированного изменения файла общих шаблонов normal.dot, что требуется для распростране-

ния в КС многих макровирусов, доступ к этому файлу может быть защищен с помощью специального пароля. Для его установки необходимо:

1) открыть окно системы программирования Visual Basic for Applications с помощью команды меню программы пакета MS Office Сервис / Макрос / Редактор Visual Basic;

2) в окне структуры проекта выделить узел Normal и выполнить команду его контекстного меню Normal Properties;

3) открыть вкладку Protectoin, установить флажок Lock project for viewing и внести в поле Password (с подтверждением в поле Confirm password) пароль для доступа к файлу общих шаблонов.

Для снижения риска заражения вирусами при просмотре информационных ресурсов сети Интернет могут быть использованы свойства обозревателя Microsoft Internet Explorer (вкладка «Безопасность» окна свойств). Узлам зоны Интернет можно назначить разные уровни безопасности.

К программно-аппаратным методам защиты от заражения загрузочными вирусами можно отнести защиту, устанавливаемую с помощью программы BIOS Setup (параметр Anti-Virus Protection или аналогичный функции Advanced BIOS Features). Включение этой защиты (задание значения Enable указанному параметру) обеспечит выдачу предупреждающего сообщения при попытке записи в загрузочные сектора дисковой памяти. К недостаткам подобной защиты от заражения вирусами относится то, что она может быть отключена кодом вируса прямым редактированием содержимого энергозависимой CMOS-памяти, хранящей настройки, которые были установлены программой BOIS Setup.

Другим способом программно-аппаратной защиты от заражения компьютерными вирусами может быть использование специального контроллера, вставляющегося в один из разъемов для расширений аппаратного обеспечения компьютера, и драйвера для управления работой контроллера. Поскольку контроллер подключается к системной шине компью-

тера, он получает полный контроль всех обращений к его дисковой памяти. С помощью драйвера контроллера могут быть указаны недоступные для изменения области дисковой памяти (загрузочные сектора, области установленного на компьютере системного и прикладного программного обеспечения и т.п.). В этом случае заражение указанных областей любыми вирусами будет невозможно. К недостаткам подобной защиты относится то, что в указанные области дисковой памяти будет невозможна и легальная запись данных.

Обязательным средством антивирусной защиты является использование специальных программ для обнаружения и удаления вирусов в различных объектах КС. Рассмотрим методы обнаружения компьютерных вирусов.

1. **Просмотр (сканирование) проверяемых объектов** (системных областей дисковой и оперативной памяти, а также файлов заданных типов) в поиске сигнатур (уникальных последовательностей байтов) известных вирусов. Соответствующие программные средства называют сканерами, а при наличии дополнительной функции удаления обнаруженных вирусов – полифагами. Обычно сканеры запускаются при загрузке операционной системы или после обнаружения признаков вирусного заражения другими средствами.

К недостаткам программ-сканеров относятся:

необходимость постоянного обновления баз данных сигнатур известных вирусов, которые используются при поиске;

неспособность обнаружения новых компьютерных вирусов;

недостаточная способность обнаружения сложных полиморфных вирусов.

2. Обнаружение изменений в объектах КС путем сравнения их вычисленных при проверке хеш-значений с эталонными (или проверки ЭЦП для этих объектов). При вычислении хеш-значений объектов могут учитываться и характеристики (атрибуты) проверяемых файлов. Подобные

программные средства называют ревизорами, или инспекторами. Потенциально они могут обнаружить и новые вирусы. Однако не все изменения проверяемых объектов вызываются вирусным заражением: обновление отдельных компонентов операционной системы, легальное изменение файлов документов и пакетных командных файлов и т.п. Программы-ревизоры не могут помочь при записи на жесткий диск компьютера пользователя уже зараженного файла, но могут обнаружить заражение вирусом новых объектов. Обычно программы-ревизоры выполняются при загрузке операционной системы.

3. Эвристический анализ – проверка системных областей памяти и файлов с целью обнаружения фрагментов исполнимого кода, характерного для компьютерных вирусов (например, установка резидентной части кода вируса). Потенциально эвристические анализаторы способны обнаружить (с определенной вероятностью) любые новые разновидности компьютерных вирусов. В Российской Федерации наиболее известным программным средством такого рода является программа DrWeb И. Данилова.

4. **Постоянное присутствие в оперативной памяти компьютера** с целью контроля всех подозрительных действий других программ – попыток изменения загрузочных секторов дисков, установки резидентного модуля и т.п. Подобные программы получили название **мониторов**. Мониторы также автоматически проверяют на наличие известных вирусов все устанавливаемые дискеты и компакт-диски, открываемые файлы и т.п. Обычно мониторы используют общую со сканерами базу сигнатур вирусов и загружаются в оперативную память в процессе загрузки операционной системы.

5. Вакцинирование – присоединение к защищаемому файлу специального модуля контроля, следящего за целостностью данного файла с помощью вычисления его хеш-значения и сравнения с эталоном. После заражения файла вирусом его целостность будет нарушена. Однако вирусы-невидимки способны обнаруживать присоединенный код программы-

вакцины и обходить реализуемую им проверку. Кроме того, данный метод плохо применим для защиты файлов документов MS Office.

Большинство современных комплексов антивирусных программ включают в свой состав сканеры (с дополнительной функцией избыточного сканирования или эвристического анализа), мониторы и, реже, инспекторы.

Сложные разновидности даже известных вирусов не всегда могут быть удалены, а зараженные ими файлы восстановлены. Поэтому для обязательной подготовки к возможному заражению объектов КС вирусами необходимо:

подготовить защищенную от записи системную дискету или загрузочный компакт-диск, записав на них последние версии антивирусных программ и баз сигнатур известных вирусов:

постоянно обновлять версии установленного в КС антивирусного программного обеспечения;

регулярно проверять объекты КС всеми имеющимися антивирусными программами (в том числе и инспекторами);

обязательно проверять на наличие вирусов все входящие сообщения электронной почты и присоединенные к ним файлы;

регулярно выполнять резервное копирование наиболее важных системных и прикладных файлов;

отключить максимально возможное число каналов распространения вирусов.

8.3. Методы защиты от программных закладок

Одним из возможных методов защиты от программных закладок является использование принципа минимальных полномочий, в соответствии с которым каждому субъекту (процессу или пользователю) всегда предоставляются в системе минимальные права.

Для обнаружения присутствия в системе программной закладки могут применяться следующие способы:

просмотр списка активных процессов с помощью диспетчера задач операционной системы;

просмотр состояния IP-портов с помощью системной программы netstat;

просмотр разделов реестра для обнаружения дополнительно установленных программ, которые автоматически выполняются при загрузке операционной системы;

просмотр файла аудита для поиска попыток доступа неизвестных процессов к критичным объектам КС или объектам с конфиденциальной информацией;

контроль обращений процессов к объектам файловой системы, разделам реестра и используемым сетевыми программами портам (например, с помощью известных программ FileMon, RegMon и PortMon) и др.

Некоторые антивирусные программы (сканеры и мониторы) могут обнаруживать инсталляторы закладок и сами закладки.

Наиболее эффективным методом защиты от программных закладок является использование организационных мер, к которым можно отнести следующие:

минимизация времени работы в КС с полномочиями администратора;

создание специальной учетной записи пользователя КС для выхода в сеть Интернет с минимальными полномочиями (запуск обозревателя и сохранение файлов в специальной папке);

аккуратное использование почтовых и офисных программ привилегированными пользователями (например, запрет доступа администратора к отдельным папкам и файлам).

Помимо организационных мер для выявления программных закладок эффективными могут оказаться методы семантического анализа исполнимого кода системных и прикладных модулей с целью поиска небезопасных для КС участков и

(или) недокументированных функций. Для этого должны применяться дисассемблирование кода и эмуляции его выполнения с помощью специальных отладчиков.

Эффективным методом защиты от вредоносных программ является создание изолированной программной среды, обладающей следующими свойствами:

на компьютере с проверенной BIOS установлена проверенная операционная система;

достоверно установлена целостность модулей операционной системы и BIOS для данного сеанса работы пользователя;

исключен запуск любых программ в данной программно-аппаратной среде, кроме проверенных;

исключен запуск проверенных программ вне проверенной среды их выполнения (т.е. в обход контролируемых проверенной средой событий).

Но в соответствии с комплексным подходом к обеспечению информационной безопасности универсальных приемов, сохраняющих постоянную эффективность, быть не может. Требуется тщательный анализ новой информации о типах программных закладок и способах их внедрения в КС для выбора адекватных методов защиты.

8.4. Принципы построения систем защиты от копирования

Под **системой защиты от несанкционированного использования и копирования** понимается комплекс программных или программно-аппаратных средств, предназначенных для усложнения или запрещения нелегального распространения, использования и (или) изменения программных продуктов и иных информационных ресурсов.

Термин «нелегальное» означает производимое без согласия правообладателя. Нелегальное изменение информационного ресурса может потребоваться нарушителю для того, что-

бы измененный им продукт не попадал под действие законодательства о защите авторских прав.

Под **надежностью системы защиты от несанкционированного копирования** понимается ее способность противостоять попыткам изучения алгоритма ее работы и обхода реализованных в ней методов защиты.

Можно выделить **следующие принципы создания и использования систем защиты от копирования**.

1. Учет условий распространения программных продуктов:

распространение дистрибутивных файлов на магнитных носителях через сеть торговых агентов или через сеть Интернет с последующей установкой самим пользователем. В этом случае пользователь может попытаться скопировать дистрибутивные магнитные диски, исследовать алгоритм работы системы защиты при помощи специальных программных средств (отладчиков и дисассемблеров), попытаться нарушить условия лицензионного соглашения и установить продукт на большем числе компьютеров;

установка программного продукта официальным представителем правообладателя. В данном случае пользователь может попытаться нарушить условия лицензионного соглашения или исследовать алгоритм работы системы защиты;

приобретение и использование программного продукта лицами или организациями, не заинтересованными в его нелегальном распространении среди их коммерческих конкурентов. В этом случае возможны только попытки несанкционированного использования продукта другими лицами;

приобретение программного продукта только для снятия с него системы защиты.

4. Учет возможностей пользователей программного продукта по снятию с него системы защиты (наличие достаточных материальных ресурсов, возможность привлечения необходимых специалистов и т.п.).

5. Учет свойств распространяемого программного продукта (предполагаемого тиража, оптовой и розничной цены, частоты обновления, специализированности и сложности продукта, уровня послепродажного сервиса для легальных пользователей, возможности применения правовых санкций к нарушителю и др.).

6. Оценка возможных потерь при снятии защиты и нелегальном использовании.

7. Учет особенностей уровня знаний и квалификации лиц, снимающих систему защиты.

8. Постоянное обновление использованных в системе защиты средств.

При добавлении к программному продукту системы его защиты от копирования возможен выбор уже имеющейся системы, что минимизирует издержки на установку системы защиты. Однако имеющаяся система защиты от копирования будет более легко сниматься с программного продукта в силу ее известности, а также может оказаться несовместимой с защищаемой программой и имеющимся у пользователя программно-аппаратным обеспечением. Поэтому более целесообразной является разработка специализированной системы защиты от копирования конкретного программного продукта, что, однако, более заметно увеличит затраты на его производство.

Существуют следующие основные **компоненты системы защиты программных продуктов от несанкционированного копирования**:

модуль проверки ключевой информации (некопируемой метки на дистрибутивном диске, уникального набора характеристик компьютера, идентифицирующей информации для легального пользователя). Данный модуль может быть добавлен к исполнимому коду защищаемой программ по технологии компьютерного вируса, в виде отдельного программного модуля или в виде отдельной функции проверки внутри защищаемой программы;

модуль защиты от изучения алгоритма работы системы защиты;

модуль согласования с работой функций защищаемой программы в случае ее санкционированного использования;

модуль ответной реакции в случае попытки несанкционированного использования.

8.5. Методы защиты от копирования

Основным методом защиты инсталляционных дисков, содержащих дистрибутивные файлы, которые используются для установки программы на компьютере пользователя, является нанесение на инсталляционный диск некопируемой метки.

Некопируемая метка – совокупность информационных характеристик магнитного носителя, существенно изменяющаяся при его копировании. Возможно нанесение магнитной или физической некопируемой метки.

К основным способам нанесения магнитной некопируемой метки относятся:

вынос метки за пределы стандартного поля копирования информации на магнитном носителе;

нестандартная разметка одной или нескольких дорожек диска;

привязка к временным характеристикам чтения и записи информации с магнитного носителя;

комбинирование нескольких способов.

Программная реализация данных способов возможна с помощью различных функций, встроенных в операционную систему.

Способ получения физической некопируемой метки основан на повреждении (например, лазерным лучом) небольшой части поверхности диска.

Другим **методом защиты от копирования** является **настройка устанавливаемого программного обеспечения на характеристики компьютера**.

Настройка устанавливаемого программного обеспечения на характеристики компьютера пользователя заключается в определении максимально полного набора таких характеристик, их хешировании, получении электронной цифровой подписи с помощью секретного ключа пользователя и записи ЭЦП в реестр операционной системы в раздел с настройками текущего пользователя.

Предлагается на следующие характеристики компьютера выполнять настройки устанавливаемого программного обеспечения:

имя компьютера;

имя пользователя;

версия операционной системы;

параметры центрального процессора;

параметры оперативной памяти;

тип используемой клавиатуры;

параметры используемой мыши;

ширина и высота экрана монитора;

информация о дисковых устройствах компьютера;

параметры диска, на котором выполняется установка программного обеспечения (емкость, тип файловой системы, серийный номер, метка тома) и др.

Для получения значений указанных характеристик могут использоваться функции из набора Windows API.

Методы настройки устанавливаемого программного продукта на характеристики компьютера имеют существенный недостаток, связанный с тем, что возможно изменение части характеристик компьютера, на котором был установлен продукт. В этом случае могут потребоваться удаление и повторная установка защищенной программы.

Если при ее инсталляции с защищенного дистрибутивного диска изменялось значение счетчика возможных установок,

то потребуется также применение специальной программы деинсталляции защищенного программного продукта.

К наиболее надежным методам защиты от несанкционированного копирования программных продуктов относится использование специальных электронных ключей, присоединенных к компьютеру пользователя при помощи USB-порта. Подобная программно-аппаратная система защиты работает следующим образом:

приложение настраивается правообладателем на характеристики электронного ключа при помощи специального программного обеспечения;

при работе защищаемой программы она обменивается с электронным ключом аутентифицирующей информацией, подтверждающей подлинность ключа;

при отсутствии ключа или наличия у него иных характеристик защищаемое приложение не может быть использовано.

Недостатком данного метода защиты от копирования является увеличение стоимости для пользователя защищенной программы, что может оказаться неприемлемым для отдельных классов программных средств (например, игровых программ, учебных программ и т.п.).

Применение даже всей совокупности рассмотренных методов и средств не позволит создать совершенную систему защиты от несанкционированного использования и копирования программ. К сожалению, любая подобная система будет снята, если нарушитель обладает неограниченными временными и материальными ресурсами.

ЗАКЛЮЧЕНИЕ

За последние годы в области информационной безопасности накоплен огромный материал. Несмотря на интенсивное развитие компьютерных средств и информационных технологий, уязвимость современных информационных систем не уменьшается. Без знаний и квалифицированного применения

новых технологий, стандартов, протоколов, методов и средств защиты компьютерной информации практически невозможно квалифицированно работать на компьютере.

Поэтому ознакомление студентов специальности 230101 «Вычислительные машины, комплексы, системы и сети» с основами защиты компьютерной информации, методами и средствами защиты данных в вычислительных системах является важной частью их профессионального обучения.

Разобраться во всем многообразии проблем информационной безопасности, упорядочить представления о методах и средствах защиты информации поможет данное пособие.

В учебном пособии были изложены следующие вопросы:

основы компьютерной безопасности, основные понятия в области защиты информации;

классификация методов и средств защиты информации в компьютерных системах, при этом были рассмотрены правовые, административные, программно-аппаратные методы и средства защиты;

криптографические методы и средства защиты информации, представлены основные алгоритмы шифрования данных;

основные модели разграничения доступа к данным, реализованные в операционных системах;

алгоритмы аутентификации пользователей;

методы и средства защиты информации в компьютерных сетях;

методы и средства защиты компьютерных систем от вредоносных программ, несанкционированного использования и копирования.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Анин Б.Ю. Защита компьютерной информации / Б.Ю. Анин. СПб.: БХВ-Петербург, 2000.

2. Барсуков В.С. Современные технологии безопасности / В.С. Барсуков, В.В. Водолазкий. М.: Нолидж, 2000.

3. Баричев С.Г. Основы современной криптографии / С.Г. Баричев, В.В. Гончаров, Р.Е. Серов. М.: Горячая линия – Телеком, 2001.

4. Галатенко В.А. Основы информационной безопасности / В.А. Галатенко. М.: Интернет-университет информационных технологий, 2006.

5. Государственная техническая комиссия при Президенте Российской Федерации. Сборник руководящих документов по защите информации от несанкционированного доступа. – М.: СИП РИА, 1997.

6. Зима В.М. Безопасность глобальных сетевых технологий / В.М. Зима, А.А. Молдовян, Н.А. Молдовян. СПб.: БХВ-Петербург, 2003.

7. Крысин А. Информационная безопасность. Практическое руководство / А.Крысин. М.: СПАРРК, Век+, 2003.

8. Малюк А.А. Введение в защиту информации в автоматизированных системах / А.А. Малюк, С.В. Пазизин, Н.С. Погожин. М.: Горячая линия – Телеком, 2001.

9. Мамаев М. Технологии защиты информации в Интернете: специальный справочник / М. Мамаев, С. Петренко. СПб.: Питер-пресс, 2001.

10. Мельников В.В. Безопасность информации в автоматизированных системах / В.В. Мельников. М.: Финансы и статистика, 2003.

11. Программирование алгоритмов защиты информации / А.В. Домашев, В.О. Попов, Д.И. Правиков и др. М.: Нолидж, 2000.

12. Проскурин В.Г. Программно-аппаратные средства обеспечения информационной безопасности. Защита в опера-

ционных системах / В.Г. Проскурин, С.В. Крутов, И.В. Мацкевич. М.: Радио и связь, 2000.

13. Романец Ю.В. Защита информации в компьютерных системах и сетях / Ю.В. Романец, П.А. Тимофеев, В.Ф. Шаныгин. М.: Радио и связь, 1999.

14. Смит Р.Э. Аутентификация: от паролей до открытых ключей / Р.Э. Смит. М.: Издательский дом «Вильямс», 2002.

15. Степанов Е.А. Информационная безопасность и защита информации: учеб. пособие / Е.А. Степанов, И.К. Корнеев. М.: Инфра-М, 2000.

16. Столингс В. Основы защиты сетей. Приложения и стандарты / В. Столингс. М.: Издательский дом «Вильямс», 2002.

17. Теоретические основы компьютерной безопасности / П.Н. Девянин, О.О. Михальский, Д.И. Правиков, А.Ю. Щербаков. М.: Радио и связь, 2000.

18. Хорев П.Б. Методы и средства защиты информации в компьютерных системах: учеб. пособие / П.Б. Хорев. М.: Издательский центр «Академия», 2005.

19. Щербаков А.Ю. Прикладная криптография. Использование и синтез криптографических интерфейсов / А.Ю. Щербаков, А.В. Домашев. М.: Издательско-торговый дом «Русская редакция», 2003.

ОГЛАВЛЕНИЕ

Учебное издание

Сергеева Татьяна Ивановна
Сергеев Михаил Юрьевич

МЕТОДЫ И СРЕДСТВА ЗАЩИТЫ КОМПЬЮТЕРНОЙ ИНФОРМАЦИИ

В авторской редакции

Подписано в печать 28.02.2011.
Объем данных 1,3 Мб

ГОУВПО «Воронежский государственный
технический университет»
394026 Воронеж, Московский просп., 14

230