**Понятие информационной безопасности**

Словосочетание *"информационная безопасность"* в разных контекстах может иметь различный смысл. В Доктрине информационной безопасности Российской Федерации термин *"информационная безопасность"* используется в широком смысле. Имеется в виду состояние защищенности национальных интересов в информационной сфере, определяемых совокупностью сбалансированных интересов личности, общества и государства.

В Законе РФ "Об участии в международном информационном обмене" *информационная безопасность* определяется аналогичным образом – как состояние защищенности информационной среды общества, обеспечивающее ее формирование, использование и развитие в интересах граждан, организаций, государства.

Под ***информационной безопасностью*** мы будем понимать защищенность информации и *поддерживающей инфраструктуры* от случайных или преднамеренных воздействий естественного или искусственного характера, которые могут нанести *неприемлемый ущерб* субъектам информационных отношений, в том числе владельцам и пользователям информации и *поддерживающей инфраструктуры*. (Чуть дальше мы поясним, что следует понимать под *поддерживающей инфраструктурой*.)

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

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

Угрозы *информационной безопасности* – это оборотная сторона использования информационных технологий.

Из этого положения можно вывести два важных следствия:

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

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

**Основные составляющие информационной безопасности**

*Информационная безопасность* – многогранная, можно даже сказать, многомерная область деятельности, в которой успех может принести только систематический, комплексный подход.

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

Поясним понятия доступности, целостности и конфиденциальности.

Доступность – это возможность за приемлемое время получить требуемую информационную услугу. Под целостностью подразумевается актуальность и непротиворечивость информации, ее защищенность от разрушения и несанкционированного изменения.

Наконец, конфиденциальность – это защита от несанкционированного доступа к информации.

Информационные системы создаются (приобретаются) для получения определенных информационных услуг. Если по тем или иным причинам предоставить эти услуги пользователям становится невозможно, это, очевидно, наносит ущерб всем *субъектам информационных отношений*. Поэтому, не противопоставляя доступность остальным аспектам, мы выделяем ее как важнейший элемент *информационной безопасности*.

Целостность можно подразделить на статическую (понимаемую как неизменность информационных объектов) и динамическую (относящуюся к корректному выполнению сложных действий (транзакций)). Средства контроля динамической целостности применяются, в частности, при анализе потока финансовых сообщений с целью выявления кражи, переупорядочения или дублирования отдельных сообщений.

Целостность оказывается важнейшим аспектом *ИБ* в тех случаях, когда информация служит "руководством к действию". Рецептура лекарств, предписанные медицинские процедуры, набор и характеристики комплектующих изделий, ход технологического процесса – все это примеры информации, нарушение целостности которой может оказаться в буквальном смысле смертельным. Неприятно и искажение официальной информации, будь то текст закона или страница Web-сервера какой-либо правительственной организации. Конфиденциальность – самый проработанный у нас в стране аспект *информационной безопасности*. К сожалению, практическая реализация мер по обеспечению конфиденциальности современных информационных систем наталкивается в России на серьезные трудности. Во-первых, сведения о технических каналах утечки информации являются закрытыми, так что большинство пользователей лишено возможности составить представление о потенциальных рисках. Во-вторых, на пути пользовательской криптографии как основного средства обеспечения конфиденциальности стоят многочисленные законодательные препоны и технические проблемы.

**Важность и сложность проблемы информационной безопасности**

При анализе проблематики, связанной с *информационной безопасностью*, необходимо учитывать специфику данного аспекта безопасности, состоящую в том, что *информационная безопасность* есть составная часть информационных технологий – области, развивающейся беспрецедентно высокими темпами. Здесь важны не столько отдельные решения (законы, учебные курсы, программно-технические изделия), находящиеся на современном уровне, сколько механизмы генерации новых решений, позволяющие жить в темпе технического прогресса.

К сожалению, современная технология программирования не позволяет создавать безошибочные программы, что не способствует быстрому развитию средств обеспечения *ИБ*. Следует исходить из того, что необходимо конструировать надежные системы (*информационной безопасности*) с привлечением ненадежных компонентов (программ). В принципе, это возможно, но требует соблюдения определенных архитектурных принципов и контроля состояния защищенности на всем протяжении **жизненного цикла ИС**.

В таких условиях системы *информационной безопасности* должны уметь противостоять разнообразным атакам, как внешним, так и внутренним, атакам автоматизированным и скоординированным. Иногда нападение длится доли секунды; порой прощупывание уязвимых мест ведется медленно и растягивается на часы, так что подозрительная активность практически незаметна. Целью злоумышленников может быть нарушение всех составляющих *ИБ* – доступности, целостности или конфиденциальности.

**Наиболее распространенные угрозы**

**Основные определения и критерии классификации угроз**

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

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

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

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

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

Для большинства *уязвимых* мест *окно опасности* существует сравнительно долго (несколько дней, иногда - недель), поскольку за это время должны произойти следующие события:

- должно стать известно о средствах использования пробела в защите;
- должны быть выпущены соответствующие заплаты;
- заплаты должны быть установлены в защищаемой ИС.

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

*Угрозы* можно классифицировать по нескольким критериям:

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

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

**Наиболее распространенные угрозы доступности**

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

Иногда такие ошибки и являются собственно *угрозами* (неправильно введенные данные или ошибка в программе, вызвавшая крах системы), иногда они создают *уязвимые* места, которыми могут воспользоваться *злоумышленники* (таковы обычно ошибки администрирования).

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

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

- *отказ пользователей*;
- *внутренний отказ* информационной системы;

- *отказ поддерживающей инфраструктуры*.

Обычно применительно к пользователям рассматриваются следующие *угрозы*:

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

Основными источниками *внутренних отказов* являются:

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

По отношению к поддерживающей инфраструктуре рекомендуется рассматривать следующие *угрозы*:

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

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

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

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

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

**Некоторые примеры угроз доступности**

Угрозы доступности могут выглядеть грубо - как повреждение или даже разрушение **оборудования** (в том числе носителей данных). Такое повреждение может вызываться естественными причинами (чаще всего - грозами). К сожалению, находящиеся в массовом использовании источники бесперебойного питания не защищают от мощных кратковременных импульсов, и случаи выгорания оборудования - не редкость.

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

Перейдем теперь к угрозам доступности. Речь пойдет о программных атаках на доступность.

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

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

По отношению к атаке "Papa Smurf" уязвимы сети, воспринимающие ping-пакеты с широковещательными адресами. Ответы на такие пакеты "съедают" полосу пропускания.

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

Для выведения систем из штатного режима эксплуатации могут использоваться уязвимые места в виде программных и аппаратных ошибок. Например, известная ошибка в процессоре Pentium I дает возможность локальному пользователю путем выполнения определенной команды "подвесить" компьютер, так что помогает только аппаратный RESET.

Программа "Teardrop" удаленно "подвешивает" компьютеры, эксплуатируя ошибку в сборке фрагментированных IP-пакетов.

**Вредоносное программное обеспечение**

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

Мы выделим следующие грани вредоносного ПО:

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

Часть, осуществляющую разрушительную функцию, будем называть "бомбой" (хотя, возможно, более удачными терминами были бы "заряд" или "боеголовка"). Вообще говоря, спектр вредоносных функций неограничен, поскольку "бомба", как и любая другая программа, может обладать сколь угодно сложной логикой, но обычно "бомбы" предназначаются для:

- внедрения другого вредоносного ПО;
- получения контроля над атакуемой системой;
- агрессивного потребления ресурсов;
- изменения или разрушения программ и/или данных.

По механизму распространения различают:

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

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

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

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

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

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

**Основные угрозы целостности**

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

В большинстве случаев виновниками оказывались штатные сотрудники организаций, отлично знакомые с режимом работы и мерами защиты. Это еще раз подтверждает опасность внутренних угроз, хотя говорят и пишут о них значительно меньше, чем о внешних.

Ранее мы проводили различие между статической и динамической целостностью. С целью нарушения статической целостности злоумышленник (как правило, штатный сотрудник) может:

- ввести неверные данные;
- изменить данные.

Иногда изменяются содержательные данные, иногда - служебная информация.

Угрозой целостности является не только фальсификация или изменение данных, но и отказ от совершенных действий. Если нет средств обеспечить "неотказуемость", компьютерные данные не могут рассматриваться в качестве доказательства.

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

Угрозами динамической целостности являются нарушение атомарности транзакций, переупорядочение, кража, дублирование данных или внесение дополнительных сообщений (сетевых пакетов и т.п.). Соответствующие действия в сетевой среде называются активным прослушиванием.

**Основные угрозы конфиденциальности**

Конфиденциальную информацию можно разделить на предметную и служебную. Служебная информация (например, пароли пользователей) не относится к определенной предметной области, в информационной системе она играет техническую роль, но ее раскрытие

особенно опасно, поскольку оно чревато получением несанкционированного доступа ко всей информации, в том числе предметной.

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

Многим людям приходится выступать в качестве пользователей не одной, а целого ряда систем (информационных сервисов). Если для доступа к таким системам используются многоразовые пароли или иная конфиденциальная информация, то наверняка эти данные будут храниться не только в голове, но и в записной книжке или на листках бумаги, которые пользователь часто оставляет на рабочем столе, а то и попросту теряет. И дело здесь не в неорганизованности людей, а в изначальной непригодности парольной схемы.

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

Кражи оборудования являются угрозой не только для резервных носителей, но и для компьютеров, особенно портативных.

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

**Законодательный уровень информационной безопасности**

Для защиты интересов субъектов информационных отношений необходимо сочетать меры следующих уровней:

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

Законодательный уровень является важнейшим для обеспечения информационной безопасности.

На законодательном уровне различаются две группы мер:

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

Самое важное (и, вероятно, самое трудное) на законодательном уровне - создать механизм, позволяющий согласовать процесс разработки законов с реалиями и прогрессом информационных технологий.

**Обзор российского законодательства в области информационной безопасности**

**Правовые акты общего назначения, затрагивающие вопросы информационной безопасности**

Основным законом Российской Федерации является Конституция.

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

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

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

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

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

Рассмотрим Уголовный кодекс Российской Федерации. Глава 28 - "Преступления в сфере компьютерной информации" - содержит три статьи:

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

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

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

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

**Закон "Об информации, информатизации и защите информации"**

Основополагающим среди российских законов, посвященных вопросам информационной безопасности, следует считать закон "Об информации, информатизации и защите информации" от 20 февраля 1995 года номер 24-ФЗ (принят Государственной Думой 25 января 1995 года). В нем даются основные определения и намечаются направления развития законодательства в данной области.

Процитируем некоторые из этих определений:

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

Мы, разумеется, не будем обсуждать качество данных в Законе определений. Обратим лишь внимание на гибкость определения конфиденциальной информации, которая не сводится к сведениям, составляющим государственную тайну, а также на понятие персональных данных, закладывающее основу защиты последних.

Закон выделяет следующие цели защиты информации:

- предотвращение утечки, хищения, утраты, искажения, подделки информации;
- предотвращение угроз безопасности личности, общества, государства;
- предотвращение несанкционированных действий по уничтожению, модификации, искажению, копированию, блокированию информации;
- предотвращение других форм незаконного вмешательства в информационные ресурсы и информационные системы, обеспечение правового режима документированной информации как объекта собственности;
- защита конституционных прав граждан на сохранение личной тайны и конфиденциальности персональных данных, имеющихся в информационных системах;
- сохранение государственной тайны, конфиденциальности документированной информации в соответствии с законодательством;
- обеспечение прав субъектов в информационных процессах и при разработке, производстве и применении информационных систем, технологий и средств их обеспечения.

Положение констатирует, что защита информации направлена на обеспечение интересов субъектов информационных отношений.

Режим защиты информации устанавливается:

- в отношении сведений, отнесенных к государственной тайне, - уполномоченными органами на основании Закона Российской Федерации "О государственной тайне";

- в отношении конфиденциальной документированной информации - собственником информационных ресурсов или уполномоченным лицом на основании настоящего Федерального закона;
- в отношении персональных данных - федеральным законом.

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

В качестве основного закон предлагает для защиты универсальные средства: лицензирование и сертификацию. Процитируем статью 19.

1. Информационные системы, базы и банки данных, предназначенные для информационного обслуживания граждан и организаций, подлежат сертификации в порядке, установленном Законом Российской Федерации "О сертификации продукции и услуг".
2. Информационные системы органов государственной власти Российской Федерации и органов государственной власти субъектов Российской Федерации, других государственных органов, организаций, которые обрабатывают документированную информацию с ограниченным доступом, а также средства защиты этих систем подлежат обязательной сертификации. Порядок сертификации определяется законодательством Российской Федерации.
3. Организации, выполняющие работы в области проектирования, производства средств защиты информации и обработки персональных данных, получают лицензии на этот вид деятельности. Порядок лицензирования определяется законодательством Российской Федерации.
4. Интересы потребителя информации при использовании импортной продукции в информационных системах защищаются таможенными органами Российской Федерации на основе международной системы сертификации.

**Другие законы и нормативные акты**

Обзор Законом "О лицензировании отдельных видов деятельности" от 8 августа 2001 года номер 128-ФЗ (Принят Государственной Думой 13 июля 2001 года). Начнем с основных определений.

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

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

Лицензирование - мероприятия, связанные с предоставлением лицензий, переоформлением документов, подтверждающих наличие лицензий, приостановлением и возобновлением действия лицензий, аннулированием лицензий и контролем лицензирующих органов за соблюдением лицензиатами при осуществлении лицензируемых видов деятельности соответствующих лицензионных требований и условий.

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

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

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

- распространение шифровальных (криптографических) средств;
- техническое обслуживание шифровальных (криптографических) средств;
- предоставление услуг в области шифрования информации;

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

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

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

Статья 17: "Сертификация информационных продуктов, информационных услуг, средств международного информационного обмена.

1. При ввозе информационных продуктов, информационных услуг в Российскую Федерацию импортер представляет сертификат, гарантирующий соответствие данных продуктов и услуг требованиям договора. В случае невозможности сертификации ввозимых на территорию Российской Федерации информационных продуктов, информационных услуг ответственность за использование данных продуктов и услуг лежит на импортере.
2. Средства международного информационного обмена, которые обрабатывают документированную информацию с ограниченным доступом, а также средства защиты этих средств подлежат обязательной сертификации.
3. Сертификация сетей связи производится в порядке, определяемом Федеральным законом "О связи"."

Закон "Об электронной цифровой подписи" номер 1-ФЗ (принят Государственной Думой 13 декабря 2001 года), развивающий и конкретизирующий приведенные выше положения закона "Об информации...".

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

Закон вводит следующие основные понятия:

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

Электронная цифровая подпись - реквизит электронного документа, предназначенный для защиты данного электронного документа от подделки, полученный в результате криптографического преобразования информации с использованием закрытого ключа электронной цифровой подписи и позволяющий идентифицировать владельца сертификата ключа подписи, а также установить отсутствие искажения информации в электронном документе.

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

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

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

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

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

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

Подтверждение подлинности электронной цифровой подписи в электронном документе - положительный результат проверки соответствующим сертифицированным средством электронной цифровой подписи с использованием сертификата ключа подписи принадлежности электронной цифровой подписи в электронном документе владельцу сертификата ключа подписи и отсутствия искажений в подписанном данной электронной цифровой подписью электронном документе.

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

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

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

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

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

Закон определяет сведения, которые должен содержать сертификат ключа подписи:

- уникальный регистрационный номер сертификата ключа подписи, даты начала и окончания срока действия сертификата ключа подписи, находящегося в реестре удостоверяющего центра;
- фамилия, имя и отчество владельца сертификата ключа подписи или псевдоним владельца. В случае использования псевдонима запись об этом вносится удостоверяющим центром в сертификат ключа подписи;
- открытый ключ электронной цифровой подписи;
- наименование средств электронной цифровой подписи, с которыми используется данный открытый ключ электронной цифровой подписи;
- наименование и местонахождение удостоверяющего центра, выдавшего сертификат ключа подписи;
- сведения об отношениях, при осуществлении которых электронный документ с электронной цифровой подписью будет иметь юридическое значение.

**Обзор зарубежного законодательства в области информационной безопасности**

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

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

Согласно Закону, все операторы федеральных ИС, содержащих конфиденциальную информацию, должны сформировать планы обеспечения ИБ. Обязательным является и периодическое обучение всего персонала таких ИС. НИСТ, в свою очередь, обязан проводить исследования природы и масштаба уязвимых мест, вырабатывать экономически оправданные меры защиты. Результаты исследований рассчитаны на применение не только в государственных системах, но и в частном секторе.

Закон обязывает НИСТ координировать свою деятельность с другими министерствами и ведомствами, включая Министерство обороны, Министерство энергетики, Агентство национальной безопасности (АНБ) и т.д., чтобы избежать дублирования и несовместимости.

Помимо регламентации дополнительных функций НИСТ, Закон предписывает создать при Министерстве торговли комиссию по информационной безопасности, которая должна:

- выявлять перспективные управленческие, технические, административные и физические меры, способствующие повышению ИБ;
- выдавать рекомендации Национальному институту стандартов и технологий, доводить их до сведения всех заинтересованных ведомств.

В 1997 году появилось продолжение описанного закона - законопроект "О совершенствовании информационной безопасности" (Computer Security Enhancement Act of

1997, H.R. 1903), направленный на усиление роли Национального института стандартов и технологий и упрощение операций с криптосредствами.

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

Для защиты федеральных ИС рекомендуется более широко применять технологические решения, основанные на разработках частного сектора. Кроме того, предлагается оценить возможности общедоступных зарубежных разработок.

Программа безопасности, предусматривающая экономически оправданные защитные меры и синхронизированная с жизненным циклом ИС, упоминается в законодательстве США неоднократно. Согласно пункту 3534 ("Обязанности федеральных ведомств") подглавы II ("Информационная безопасность") главы 35 ("Координация федеральной информационной политики") рубрики 44 ("Общественные издания и документы"), такая программа должна включать:

- периодическую оценку рисков с рассмотрением внутренних и внешних угроз целостности, конфиденциальности и доступности систем, а также данных, ассоциированных с критически важными операциями и ресурсами;
- правила и процедуры, позволяющие, опираясь на проведенный анализ рисков, экономически оправданным образом уменьшить риски до приемлемого уровня;
- обучение персонала с целью информирования о существующих рисках и об обязанностях, выполнение которых необходимо для их (рисков) нейтрализации;
- периодическую проверку и (пере)оценку эффективности правил и процедур;
- действия при внесении существенных изменений в систему;
- процедуры выявления нарушений информационной безопасности и реагирования на них; эти процедуры должны помочь уменьшить риски, избежать крупных потерь; организовать взаимодействие с правоохранительными органами.

Конечно, в законодательстве США имеются в достаточном количестве и положения ограничительной направленности, и директивы, защищающие интересы таких ведомств, как Министерство обороны, АНБ, ФБР, ЦРУ.

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

Как, вероятно, и во всех других законах аналогичной направленности, в данном случае устанавливается приоритет интересов национальной безопасности над сохранением тайны частной жизни. В остальном права личности защищены весьма тщательно. Например, если сотрудник фирмы обрабатывает персональные данные в интересах частных компаний, он дает подписку о неразглашении, которая действует и после перехода на другую работу.

Государственные учреждения, хранящие и обрабатывающие персональные данные, несут ответственность за нарушение тайны частной жизни "субъекта данных", как говорится в Законе. В материальном выражении ответственность ограничена верхним пределом в 250 тысяч немецких марок.

Из законодательства Великобритании упомянем семейство так называемых добровольных стандартов BS 7799, помогающих организациям на практике сформировать программы безопасности.

**Оценочные стандарты и технические спецификации. "Оранжевая книга" как оценочный стандарт**

**Основные понятия**

Обзор стандартов и спецификаций двух разных видов:

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

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

Исторически первым оценочным стандартом, получившим широкое распространение и оказавшим огромное влияние на базу стандартизации ИБ во многих странах, стал стандарт Министерства обороны США "Критерии оценки доверенных компьютерных систем".

Данный труд, называемый чаще всего по цвету обложки "Оранжевой книгой", был впервые опубликован в августе 1983 года. Уже одно его название требует комментария. Речь идет не о безопасных, а о доверенных системах, то есть системах, которым можно оказать определенную степень доверия.

"Оранжевая книга" поясняет понятие безопасной системы, которая "управляет, с помощью соответствующих средств, доступом к информации, так что только должным образом авторизованные лица или процессы, действующие от их имени, получают право читать, записывать, создавать и удалять информацию".

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

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

Обратим внимание, что в рассматриваемых Критериях и безопасность, и доверие оцениваются исключительно с точки зрения управления доступом к данным, что является одним из средств обеспечения конфиденциальности и целостности (статической). Вопросы доступности "Оранжевая книга" не затрагивает.

Степень доверия оценивается по двум основным критериям.

1. Политика безопасности - набор законов, правил и норм поведения, определяющих, как организация обрабатывает, защищает и распространяет информацию. В частности, правила определяют, в каких случаях пользователь может оперировать конкретными наборами данных. Чем выше степень доверия системе, тем строже и многообразнее должна быть политика безопасности. В зависимости от сформулированной политики можно выбирать конкретные механизмы обеспечения безопасности. Политика безопасности - это активный аспект защиты, включающий в себя анализ возможных угроз и выбор мер противодействия.
2. Уровень гарантированности - мера доверия, которая может быть оказана архитектуре и реализации ИС. Доверие безопасности может проистекать как из анализа результатов тестирования, так и из проверки (формальной или нет) общего замысла и реализации системы в целом и отдельных ее компонентов. Уровень гарантированности показывает, насколько корректны механизмы, отвечающие за реализацию политики безопасности. Это пассивный аспект защиты.

Важным средством обеспечения безопасности является механизм подотчетности (протоколирования). Доверенная система должна фиксировать все события, касающиеся безопасности. Ведение протоколов должно дополняться аудитом, то есть анализом регистрационной информации.

Концепция доверенной вычислительной базы является центральной при оценке степени доверия безопасности. Доверенная вычислительная база - это совокупность защитных механизмов ИС (включая аппаратное и программное обеспечение), отвечающих за проведение в жизнь политики безопасности. Качество вычислительной базы определяется исключительно ее реализацией и корректностью исходных данных, которые вводит системный администратор. Вообще говоря, компоненты вне вычислительной базы могут не быть доверенными, однако это не должно влиять на безопасность системы в целом. В результате, для оценки доверия безопасности ИС достаточно рассмотреть только ее вычислительную базу, которая, как можно надеяться, достаточно компактна.

Основное назначение доверенной вычислительной базы - выполнять функции монитора обращений, то есть контролировать допустимость выполнения субъектами (активными сущностями ИС, действующими от имени пользователей) определенных операций над объектами (пассивными сущностями). Монитор проверяет каждое обращение пользователя к программам или данным на предмет согласованности с набором действий, допустимых для пользователя.

Монитор обращений должен обладать тремя качествами:

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

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

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

**Административный уровень информационной безопасности**

**Основные понятия**

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

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

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

*Политика безопасности* строится на основе *анализа рисков*, которые признаются реальными для информационной системы организации. Когда риски проанализированы и стратегия защиты определена, составляется программа обеспечения информационной безопасности. Под эту программу выделяются ресурсы, назначаются ответственные, определяется порядок контроля выполнения программы и т.п.

Под ***политикой безопасности*** понимают совокупность документированных решений, принимаемых руководством организации и направленных на защиту информации и ассоциированных с ней ресурсов.

Такая трактовка, конечно, гораздо шире, чем набор *правил разграничения доступа* (именно это означал термин "security policy" в "Оранжевой книге" и в построенных на ее основе нормативных документах других стран).

ИС организации и связанные с ней интересы субъектов - это сложная система, для рассмотрения которой необходимо применять объектно-ориентированный подход и понятие уровня детализации. Целесообразно выделить, по крайней мере, три таких уровня, что мы уже делали в примере и сделаем еще раз далее.

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

**Политика безопасности**

С практической точки зрения политику безопасности целесообразно рассматривать на **трех уровнях** детализации. К **верхнему** уровню можно отнести решения, затрагивающие организацию в целом. Они носят весьма общий характер и, как правило, исходят от руководства организации. Примерный список подобных решений может включать в себя следующие элементы:

- решение сформировать или пересмотреть комплексную программу обеспечения информационной безопасности, назначение ответственных за продвижение программы;
- формулировка целей, которые преследует организация в области информационной безопасности, определение общих направлений в достижении этих целей;
- обеспечение базы для соблюдения законов и правил;
- формулировка административных решений по тем вопросам реализации программы безопасности, которые должны рассматриваться на уровне организации в целом.

Для политики верхнего уровня цели организации в области информационной безопасности формулируются в терминах целостности, доступности и конфиденциальности. Если организация отвечает за поддержание критически важных баз данных, на первом плане может стоять уменьшение числа потерь, повреждений или искажений данных. Для организации, занимающейся продажей компьютерной техники, вероятно, важна актуальность информации о предоставляемых услугах и ценах и ее доступность максимальному числу потенциальных покупателей. Руководство режимного предприятия в первую очередь заботится о защите от несанкционированного доступа, то есть о конфиденциальности.

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

Политика верхнего уровня должна четко очерчивать сферу своего влияния. Возможно, это будут все компьютерные системы организации (или даже больше, если политика регламентирует некоторые аспекты использования сотрудниками своих домашних компьютеров). Возможна, однако, и такая ситуация, когда в сферу влияния включаются лишь наиболее важные системы.

В политике должны быть определены обязанности должностных лиц по выработке программы безопасности и проведению ее в жизнь. В этом смысле *политика безопасности* является основой подотчетности персонала.

Политика верхнего уровня имеет дело с тремя аспектами законопослушности и исполнительской дисциплины. Во-первых, организация должна соблюдать существующие законы. Во-вторых, следует контролировать действия лиц, ответственных за выработку программы безопасности. Наконец, необходимо обеспечить определенную степень исполнительности персонала, а для этого нужно выработать систему поощрений и наказаний.

Вообще говоря, на верхний уровень следует выносить минимум вопросов. Подобное вынесение целесообразно, когда оно сулит значительную экономию средств или когда иначе поступить просто невозможно.

Британский стандарт BS 7799:1995 рекомендует включать в документ, характеризующий политику безопасности организации, следующие разделы:

- вводный, подтверждающий озабоченность высшего руководства проблемами информационной безопасности;
- организационный, содержащий описание подразделений, комиссий, групп и т.д., отвечающих за работы в области информационной безопасности;
- классификационный, описывающий имеющиеся в организации материальные и информационные ресурсы и необходимый уровень их защиты;
- штатный, характеризующий меры безопасности, применяемые к персоналу (описание должностей с точки зрения информационной безопасности, организация обучения и переподготовки персонала, порядок реагирования на нарушения режима безопасности и т.п.);
- раздел, освещающий вопросы *физической защиты*;
- управляющий раздел, описывающий подход к управлению компьютерами и компьютерными сетями;
- раздел, описывающий *правила разграничения* доступа к производственной информации;
- раздел, характеризующий *порядок разработки* и сопровождения систем;
- раздел, описывающий меры, направленные на обеспечение *непрерывной работы* организации;
- юридический раздел, подтверждающий соответствие политики безопасности действующему законодательству.

К **среднему** уровню можно отнести вопросы, касающиеся отдельных аспектов информационной безопасности, но важные для различных эксплуатируемых организацией систем. Примеры таких вопросов - отношение к передовым (но, возможно, недостаточно проверенным) технологиям, доступ в Internet, использование домашних компьютеров, применение пользователями неофициального программного обеспечения и т.д.

Политика среднего уровня должна для каждого аспекта освещать следующие темы:

Описание аспекта. Например, если рассмотреть применение пользователями неофициального программного обеспечения, последнее можно определить как ПО, которое не было одобрено и/или закуплено на уровне организации.

Область применения. Следует определить, где, когда, как, по отношению к кому и чему применяется данная *политика безопасности*. Например, касается ли политика, связанная с использованием неофициального программного обеспечения, организаций-субподрядчиков? Затрагивает ли она сотрудников, пользующихся портативными и домашними компьютерами и вынужденных переносить информацию на производственные машины?

Позиция организации по данному аспекту. Продолжая пример с неофициальным программным обеспечением, можно представить себе позиции полного запрета, выработки процедуры приемки подобного ПО и т.п. Позиция может быть сформулирована и в гораздо более общем виде, как набор целей, которые преследует организация в данном аспекте. Вообще стиль документов, определяющих политику безопасности (как и их перечень), в разных организациях может сильно отличаться.

Роли и обязанности. В "политический" документ необходимо включить информацию о должностных лицах, ответственных за реализацию политики безопасности. Например, если для использования неофициального программного обеспечения сотрудникам требуется разрешение руководства, должно быть известно, у кого и как его можно получить. Если неофициальное программное обеспечение использовать нельзя, следует знать, кто следит за выполнением данного правила.

Законопослушность. Политика должна содержать общее описание запрещенных действий и наказаний за них.

Точки контакта. Должно быть известно, куда следует обращаться за разъяснениями, помощью и дополнительной информацией. Обычно "точкой контакта" служит определенное должностное лицо, а не конкретный человек, занимающий в данный момент данный пост.

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

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

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

**Программа безопасности**

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

Чтобы понять и реализовать какую-либо программу, ее нужно структурировать по уровням, обычно в соответствии со структурой организации. В простейшем и самом распространенном случае достаточно двух уровней - верхнего, или центрального, который охватывает всю организацию, и нижнего, или служебного, который относится к отдельным услугам или группам однородных сервисов.

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

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

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

*Контроль* деятельности в области безопасности имеет двустороннюю направленность. Во-первых, необходимо гарантировать, что действия организации не противоречат законам. При этом следует поддерживать контакты с внешними контролирующими организациями. Во-вторых, нужно постоянно отслеживать состояние безопасности внутри организации, реагировать на случаи нарушений и дорабатывать защитные меры с учетом изменения обстановки.

Следует подчеркнуть, что программа верхнего уровня должна занимать строго определенное место в деятельности организации, она должна официально приниматься и поддерживаться руководством, а также иметь определенный штат и бюджет.

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

**Синхронизация программы безопасности с жизненным циклом систем**

Если синхронизировать программу безопасности нижнего уровня с *жизненным циклом* защищаемого сервиса, можно добиться большего эффекта с меньшими затратами. Программисты знают, что добавить новую возможность к уже готовой системе на порядок сложнее, чем изначально спроектировать и реализовать ее. То же справедливо и для информационной безопасности.

В *жизненном цикле* информационного сервиса можно выделить следующие этапы:

*Инициация*. На данном этапе выявляется необходимость в приобретении нового сервиса, документируется его предполагаемое назначение.

*Закупка*. На данном этапе составляются спецификации, прорабатываются варианты приобретения, выполняется собственно *закупка*.

*Установка*. Сервис устанавливается, конфигурируется, тестируется и вводится в *эксплуатацию*.

*Эксплуатация*. На данном этапе сервис не только работает и администрируется, но и подвергается модификациям.

*Выведение из эксплуатации*. Происходит переход на новый сервис.

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

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

С точки зрения безопасности важнейшим действием здесь является оценка критичности как самого сервиса, так и информации, которая с его помощью будет обрабатываться. Требуется сформулировать ответы на следующие вопросы:

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

Результаты оценки критичности являются отправной точкой в составлении спецификаций. Кроме того, они определяют ту меру внимания, которую служба безопасности организации должна уделять новому сервису на последующих этапах его *жизненного цикла*.

Этап *закупки* - один из самых сложных. Нужно окончательно сформулировать требования к защитным средствам нового сервиса, к компании, которая может претендовать на роль поставщика, и к квалификации, которой должен обладать персонал, использующий или обслуживающий закупаемый продукт. Все эти сведения оформляются в виде спецификации, куда входят не только аппаратура и программы, но и документация, обслуживание, обучение персонала. Разумеется, особое внимание должно уделяться вопросам совместимости нового сервиса с существующей конфигурацией. Подчеркнем также, что нередко средства

безопасности являются необязательными компонентами коммерческих продуктов, и нужно проследить, чтобы соответствующие пункты не выпали из спецификации.

Когда продукт закуплен, его необходимо *установить*. Несмотря на кажущуюся простоту, *установка* является очень ответственным делом. Во-первых, новый продукт следует сконфигурировать. Как правило, коммерческие продукты поставляются с отключенными средствами безопасности; их необходимо включить и должным образом настроить. Для большой организации, где много пользователей и данных, начальная настройка может стать весьма трудоемким и ответственным делом.

Во-вторых, новый сервис нуждается в процедурных регуляторах. Следует позаботиться о чистоте и охране помещения, о документах, регламентирующих использование сервиса, о подготовке планов на случай экстренных ситуаций, об организации обучения пользователей и т.п.

После принятия перечисленных мер необходимо провести тестирование. Его полнота и комплексность могут служить гарантией безопасности *эксплуатации* в штатном режиме.

Период *эксплуатации* - самый длительный и сложный. С психологической точки зрения наибольшую опасность в это время представляют незначительные изменения в конфигурации сервиса, в поведении пользователей и администраторов. Если безопасность не поддерживать, она ослабевает. Пользователи не столь ревностно выполняют должностные инструкции, администраторы менее тщательно анализируют регистрационную информацию. То один, то другой пользователь получает дополнительные привилегии. Кажется, что в сущности ничего не изменилось; на самом же деле от былой безопасности не осталось и следа.

Для борьбы с эффектом медленных изменений приходится прибегать к периодическим проверкам безопасности сервиса. Разумеется, после значительных модификаций подобные проверки являются обязательными.

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

При *выведении данных из эксплуатации* их обычно переносят на другую систему, архивируют, выбрасывают или уничтожают. Если архивирование производится с намерением впоследствии прочитать данные в другом месте, следует позаботиться об аппаратно-программной совместимости средств чтения и записи. Информационные технологии развиваются очень быстро, и через несколько лет устройств, способных прочитать старый носитель, может просто не оказаться. Если данные архивируются в зашифрованном виде, необходимо сохранить ключ и средства расшифровки. При архивировании и хранении архивной информации нельзя забывать о поддержании конфиденциальности данных.

## Управление рисками

### Основные понятия

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

Вообще говоря, управление рисками, равно как и выработка собственной политики безопасности, актуально только для тех организаций, информационные системы которых и/или обрабатываемые данные можно считать нестандартными. Обычную организацию вполне устроит типовой набор защитных мер, выбранный на основе представления о типичных рисках или вообще без всякого анализа рисков (особенно это верно с формальной точки зрения, в свете проанализированного нами ранее российского законодательства в области ИБ).

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

С количественной точки зрения уровень риска является функцией вероятности реализации определенной угрозы (использующей некоторые уязвимые места), а также величины возможного ущерба.

Таким образом, суть мероприятий по управлению рисками состоит в том, чтобы оценить их размер, выработать эффективные и экономичные меры снижения рисков, а затем убедиться, что риски заключены в приемлемые рамки (и остаются таковыми). Следовательно, управление рисками включает в себя два вида деятельности, которые чередуются циклически:

- (пере)оценка (измерение) рисков;
- выбор эффективных и экономичных защитных средств (*нейтрализация рисков*).

По отношению к выявленным рискам возможны следующие действия:

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

Процесс управления рисками можно разделить на следующие этапы:

1. Выбор анализируемых объектов и уровня детализации их рассмотрения.
2. Выбор ***методологии оценки рисков***.
3. ***Идентификация активов***.
4. ***Анализ угроз*** и их последствий, ***выявление уязвимых мест*** в защите.
5. Оценка рисков.
6. Выбор защитных мер.
7. Реализация и проверка выбранных мер.
8. Оценка ***остаточного риска***.

Этапы 6 и 7 относятся к выбору защитных средств (нейтрализации рисков), остальные - к оценке рисков.

Уже перечисление этапов показывает, что управление рисками - процесс циклический. По существу, последний этап - это оператор конца цикла, предписывающий вернуться к началу. Риски нужно контролировать постоянно, периодически проводя их переоценку. Отметим, что добросовестно выполненная и тщательно документированная первая оценка может существенно упростить последующую деятельность.

Управление рисками, как и любую другую деятельность в области информационной безопасности, необходимо интегрировать в *жизненный цикл ИС*. Тогда эффект оказывается наибольшим, а затраты - минимальными. Ранее мы определили пять этапов жизненного цикла. Кратко опишем, что может дать управление рисками на каждом из них.

На этапе ***инициации*** известные риски следует учесть при выработке требований к системе вообще и средствам безопасности в частности.

На этапе ***закупки (разработки)*** знание рисков поможет выбрать соответствующие архитектурные решения, которые играют ключевую роль в обеспечении безопасности.

На этапе ***установки*** выявленные риски следует учитывать при конфигурировании, тестировании и проверке ранее сформулированных требований, а полный цикл управления рисками должен предшествовать внедрению системы в эксплуатацию.

На этапе ***эксплуатации*** управление рисками должно сопровождать все существенные изменения в системе.

При ***выведении системы из эксплуатации*** управление рисками помогает убедиться в том, что миграция данных происходит безопасным образом.

**Подготовительные этапы управления рисками**

В этом разделе будут описаны первые три этапа процесса управления рисками.

***Выбор анализируемых объектов*** и ***уровня детализации*** их рассмотрения - первый шаг в оценке рисков. Для небольшой организации допустимо рассматривать всю информационную инфраструктуру; однако если организация крупная, всеобъемлющая оценка может потребовать неприемлемых затрат времени и сил. В таком случае следует сосредоточиться на наиболее важных сервисах, заранее соглашаясь с приближенностью итоговой оценки. Если важных сервисов все еще много, выбираются те из них, риски для которых заведомо велики или неизвестны.

Вообще говоря, уязвимым является каждый компонент информационной системы - от сетевого кабеля до базы данных. Как правило, в сферу анализа невозможно включить каждый винтик и каждый байт. Приходится останавливаться на некотором уровне детализации, опять-таки отдавая себе отчет в приближенности оценки. Для новых систем предпочтителен детальный анализ; старая система, подвергшаяся небольшим модификациям, может быть проанализирована более поверхностно.

Очень важно выбрать разумную методологию оценки рисков. Целью оценки является получение ответа на два вопроса: приемлемы ли существующие риски, и если нет, то какие защитные средства стоит использовать. Значит, оценка должна быть количественной, допускающей сопоставление с заранее выбранными границами допустимости и расходами на реализацию новых регуляторов безопасности. Управление рисками - типичная оптимизационная задача, и существует довольно много программных продуктов, способных помочь в ее решении (иногда подобные продукты просто прилагаются к книгам по информационной безопасности). Принципиальная трудность, однако, состоит в неточности исходных данных. Можно, конечно, попытаться получить для всех анализируемых величин денежное выражение, высчитать все с точностью до копейки, но большого смысла в этом нет. Практичнее пользоваться условными единицами. В простейшем и вполне допустимом случае можно пользоваться трехбалльной шкалой.

При идентификации активов, то есть тех ресурсов и ценностей, которые организация пытается защитить, следует, конечно, учитывать не только компоненты информационной системы, но и поддерживающую инфраструктуру, персонал, а также нематериальные ценности, такие как репутация организации.

Одним из главных результатов процесса идентификации активов является получение детальной информационной структуры организации и способов ее (структуры) использования. Эти сведения целесообразно нанести на карту ИС в качестве граней соответствующих объектов.

Информационной основой сколько-нибудь крупной организации является сеть, поэтому в число аппаратных активов следует включить компьютеры (серверы, рабочие станции, ПК), периферийные устройства, внешние интерфейсы, кабельное хозяйство, активное сетевое оборудование (мосты, маршрутизаторы и т.п.). К программным активам, вероятно, будут отнесены операционные системы (сетевая, серверные и клиентские), прикладное программное обеспечение, инструментальные средства, средства управления сетью и отдельными системами. Важно зафиксировать, где (в каких узлах сети) хранится программное обеспечение, и из каких узлов оно используется. Третьим видом информационных активов являются данные, которые хранятся, обрабатываются и передаются по сети. Следует классифицировать данные по типам и степени конфиденциальности, выявить места их хранения и обработки, способы доступа к ним. Все это важно для оценки последствий нарушений информационной безопасности.

Управление рисками - процесс далеко не линейный. Практически все его этапы связаны между собой, и по завершении почти любого из них может возникнуть необходимость возврата к предыдущему. Так, при идентификации активов может оказаться, что выбранные границы

анализа следует расширить, а степень детализации - увеличить. Особенно труден первичный анализ, когда многократные возвраты к началу неизбежны.

**Основные этапы управления рисками**

Этапы, предшествующие анализу угроз, можно считать подготовительными, поскольку, строго говоря, они напрямую с рисками не связаны. Риск появляется там, где есть угрозы.

Краткий перечень наиболее распространенных угроз был рассмотрен нами ранее. К сожалению, на практике угроз гораздо больше, причем далеко не все из них носят компьютерный характер. Так, вполне реальной угрозой является наличие мышей и тараканов в занимаемых организацией помещениях. Первые могут повредить кабели, вторые вызвать короткое замыкание. Как правило, наличие той или иной угрозы является следствием пробелов в защите информационной системы, которые, в свою очередь, объясняются отсутствием некоторых сервисов безопасности или недостатками в реализующих их защитных механизмах. Опасность прогрызания кабелей возникает не просто там, где есть мыши, она связана с отсутствием или недостаточной прочностью защитной оболочки.

Первый шаг в анализе угроз - их идентификация. Рассматриваемые виды угроз следует выбирать исходя из соображений здравого смысла (исключив, например, землетрясения, однако не забывая о возможности захвата организации террористами), но в пределах выбранных видов провести максимально подробный анализ.

Целесообразно выявлять не только сами угрозы, но и **источники** их возникновения - это поможет в выборе дополнительных средств защиты. Например, нелегальный вход в систему может стать следствием воспроизведения начального диалога, подбора пароля или подключения к сети неавторизованного оборудования. Очевидно, для противодействия каждому из перечисленных способов нелегального входа нужны свои механизмы безопасности.

После ***идентификации угрозы*** необходимо оценить ***вероятность ее осуществления***. Допустимо использовать при этом трехбалльную шкалу (низкая (1), средняя (2) и высокая (3) вероятность).

Кроме вероятности осуществления, важен размер потенциального ущерба. Например, пожары бывают нечасто, но ущерб от каждого из них, как правило, велик. Тяжесть ущерба также можно оценить по трехбалльной шкале.

Оценивая размер ущерба, необходимо иметь в виду не только непосредственные расходы на замену оборудования или восстановление информации, но и более отдаленные, такие как подрыв репутации, ослабление позиций на рынке и т.п.

После того, как накоплены исходные данные и оценена степень неопределенности, можно переходить к обработке информации, то есть собственно к оценке рисков. Вполне допустимо применить такой простой метод, как умножение вероятности осуществления угрозы на ***предполагаемый ущерб***. Если для вероятности и ущерба использовать трехбалльную шкалу, то возможных произведений будет шесть: 1, 2, 3, 4, 6 и 9. Первые два результата можно отнести к низкому риску, третий и четвертый - к среднему, два последних - к высокому, после чего появляется возможность снова привести их к трехбалльной шкале. По этой шкале и следует оценивать приемлемость рисков. Правда, граничные случаи, когда вычисленная величина совпала с приемлемой, целесообразно рассматривать более тщательно из-за приближенного характера результата.

Если какие-либо риски оказались недопустимо высокими, необходимо их нейтрализовать, реализовав дополнительные меры защиты. Как правило, для ликвидации или нейтрализации уязвимого места, сделавшего угрозу реальной, существует несколько механизмов безопасности, различных по эффективности и стоимости. Например, если велика вероятность нелегального входа в систему, можно потребовать, чтобы пользователи выбирали длинные пароли (скажем, не менее восьми символов), задействовать программу генерации паролей или закупить интегрированную систему аутентификации на основе интеллектуальных карт. Если есть вероятность умышленного повреждения сервера баз данных, что может иметь

серьезные последствия, можно врезать замок в дверь серверной комнаты или поставить около каждого сервера по охраннику.

Оценивая стоимость мер защиты, приходится, разумеется, учитывать не только прямые расходы на закупку оборудования и/или программ, но и расходы на внедрение новинки и, в частности, обучение и переподготовку персонала. Эту стоимость также можно оценить по трехбалльной шкале и затем сопоставить ее с разностью между вычисленным и допустимым риском. Если по этому показателю новое средство оказывается экономически выгодным, его можно взять на заметку (подходящих средств, вероятно, будет несколько). Однако если средство окажется дорогим, его не следует сразу отбрасывать, памятуя о приближенности расчетов.

Выбирая подходящий способ защиты, целесообразно учитывать возможность ***экранирования*** одним механизмом обеспечения безопасности сразу нескольких прикладных сервисов.

Как и всякую иную деятельность, реализацию и проверку новых регуляторов безопасности следует предварительно планировать. В плане необходимо учесть наличие финансовых средств и сроки обучения персонала. Если речь идет о программно-техническом механизме защиты, нужно составить план тестирования (автономного и комплексного).

**Процедурный уровень информационной безопасности**

**Основные классы мер процедурного уровня**

Мы приступаем к рассмотрению мер безопасности, которые ориентированы на людей, а не на технические средства. Именно люди формируют режим информационной безопасности, и они же оказываются главной угрозой, поэтому "человеческий фактор" заслуживает особого внимания.

На *процедурном уровне* можно выделить следующие классы мер:

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

**Управление персоналом**

***Управление персоналом*** начинается с приема нового сотрудника на работу и даже раньше - с составления *описания должности*. Уже на данном этапе желательно подключить к работе специалиста по информационной безопасности для определения компьютерных привилегий, ассоциируемых с должностью. Существует два общих принципа, которые следует иметь в виду:

- *разделение обязанностей;*
- *минимизация привилегий.*

Принцип ***разделения обязанностей*** предписывает так распределять роли и ответственность, чтобы один человек не мог нарушить критически важный для организации процесс.

Принцип ***минимизации привилегий*** предписывает выделять пользователям только те права доступа, которые необходимы им для выполнения служебных обязанностей. Назначение этого принципа очевидно - уменьшить ущерб от случайных или умышленных некорректных действий.

Предварительное составление *описания должности* позволяет оценить ее критичность и спланировать процедуру проверки и отбора кандидатов. Когда кандидат определен, он, вероятно, должен пройти *обучение*; по крайней мере, его следует подробно ознакомить со служебными обязанностями, а также с нормами и процедурами информационной безопасности. Желательно, чтобы меры безопасности были им усвоены до вступления в должность и до заведения его системного счета с входным именем, паролем и привилегиями.

С момента заведения системного счета начинается его администрирование, а также протоколирование и анализ действий пользователя. Постепенно изменяется окружение, в котором работает пользователь, его служебные обязанности и т.п. Все это требует соответствующего изменения привилегий. Техническую сложность представляют временные перемещения пользователя, выполнение им обязанностей взамен сотрудника, ушедшего в отпуск, и иные обстоятельства, когда полномочия нужно сначала предоставить, а через некоторое время взять обратно. В такие периоды профиль активности пользователя резко меняется, что создает трудности при выявлении подозрительных ситуаций. Определенную аккуратность следует соблюдать и при выдаче новых постоянных полномочий, не забывая ликвидировать старые права доступа.

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

К управлению сотрудниками примыкает администрирование лиц, работающих по контракту (например, специалистов фирмы-поставщика, помогающих запустить новую систему). В соответствии с принципом *минимизации привилегий*, им нужно выделить ровно столько прав, сколько необходимо, и изъять эти права сразу по окончании контракта. Проблема, однако, состоит в том, что на начальном этапе внедрения "внешние" сотрудники будут администрировать "местных", а не наоборот. Здесь на первый план выходит квалификация персонала организации, его способность быстро *обучаться*, а также оперативное проведение учебных курсов. Важны и принципы выбора деловых партнеров.

Иногда внешние организации принимают на обслуживание и администрирование ответственные компоненты компьютерной системы, например, сетевое оборудование. Нередко администрирование выполняется в удаленном режиме. Вообще говоря, это создает в системе дополнительные уязвимые места, которые необходимо компенсировать усиленным контролем средств удаленного доступа или, опять-таки, *обучением* собственных сотрудников.

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

**Физическая защита**

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

Основной принцип ***физической защиты***, соблюдение которого следует постоянно контролировать, формулируется как "непрерывность защиты в пространстве и времени". Ранее мы рассматривали понятие окна опасности. Для *физической защиты* таких окон быть не должно.

Мы кратко рассмотрим следующие направления *физической защиты*:

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

Меры физического управления доступом позволяют контролировать и при необходимости ограничивать вход и выход сотрудников и посетителей. Контролироваться

может все здание организации, а также отдельные помещения, например, те, где расположены серверы, коммуникационная аппаратура и т.п.

При проектировании и реализации мер физического управления доступом целесообразно применять объектный подход. Во-первых, определяется периметр безопасности, ограничивающий контролируемую территорию. На этом уровне детализации важно продумать внешний интерфейс организации - порядок входа/выхода штатных сотрудников и посетителей, вноса/выноса техники. Все, что не входит во внешний интерфейс, должно быть инкапсулировано, то есть защищено от нелегальных проникновений.

Во-вторых, производится декомпозиция контролируемой территории, выделяются (под)объекты и связи (проходы) между ними. При такой, более глубокой детализации следует выделить среди подобъектов наиболее критичные с точки зрения безопасности и обеспечить им повышенное внимание. Декомпозиция должна быть семантически оправданной, обеспечивающей разграничение разнородных сущностей, таких как оборудование разных владельцев или персонал, работающий с данными разной степени критичности. Важно сделать так, чтобы посетители, по возможности, не имели непосредственного доступа к компьютерам или, в крайнем случае, позаботиться о том, чтобы от окон и дверей не просматривались экраны мониторов и принтеры. Необходимо, чтобы посетителей по внешнему виду можно было отличить от сотрудников.

Средства физического управления доступом известны давно. Это охрана, двери с замками, перегородки, телекамеры, датчики движения и многое другое. Для выбора оптимального (по критерию стоимость/эффективность) средства целесообразно провести анализ рисков (к этому мы еще вернемся). Кроме того, есть смысл периодически отслеживать появление технических новинок в данной области, стараясь максимально автоматизировать *физическую защиту*.

К поддерживающей инфраструктуре можно отнести системы электро-, водо- и теплоснабжения, кондиционеры и средства коммуникаций. В принципе, к ним применимы те же требования целостности и доступности, что и к информационным системам. Для обеспечения целостности нужно защищать оборудование от краж и повреждений. Для поддержания доступности следует выбирать оборудование с максимальным временем наработки на отказ, дублировать ответственные узлы и всегда иметь под рукой запчасти.

Отдельную проблему составляют аварии водопровода. Они происходят нечасто, но могут нанести огромный ущерб. При размещении компьютеров необходимо принять во внимание расположение водопроводных и канализационных труб и постараться держаться от них подальше. Сотрудники должны знать, куда следует обращаться при обнаружении протечек.

Перехват данных (о чем мы уже писали) может осуществляться самыми разными способами. Злоумышленник может подсматривать за экраном монитора, читать пакеты, передаваемые по сети, производить анализ побочных электромагнитных излучений и наводок (ПЭМИН) и т.д. Остается уповать на повсеместное использование криптографии (что, впрочем, сопряжено у нас в стране со множеством технических и законодательных проблем), стараться максимально расширить контролируемую территорию, разместившись в тихом особнячке, поодаль от других домов, пытаться держать под контролем линии связи (например, заключать их в надувную оболочку с обнаружением прокалывания), но самое разумное, вероятно, - постараться осознать, что для коммерческих систем обеспечение конфиденциальности является все-таки не главной задачей.

Мобильные и портативные компьютеры - заманчивый объект кражи. Их часто оставляют без присмотра, в автомобиле или на работе, и похитить такой компьютер совсем несложно. Настоятельно рекомендуется шифровать данные на жестких дисках таких компьютеров.

Вообще говоря, при выборе средств *физической защиты* следует производить анализ рисков. Так, принимая решение о закупке источника бесперебойного питания, необходимо учесть качество электропитания в здании, занимаемом организацией (впрочем, почти наверняка оно окажется плохим), характер и длительность сбоев электропитания, стоимость доступных

источников и возможные потери от аварий (поломка техники, приостановка работы организации и т.п.).

**Поддержание работоспособности**

Далее рассмотрим ряд рутинных мероприятий, направленных на ***поддержание работоспособности*** информационных систем. Именно здесь таится наибольшая опасность. Нечаянные ошибки системных администраторов и пользователей грозят повреждением аппаратуры, разрушением программ и данных; в лучшем случае они создают бреши в защите, которые делают возможной реализацию угроз.

Можно выделить следующие направления повседневной деятельности:

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

Поддержка пользователей подразумевает прежде всего консультирование и оказание помощи при решении разного рода проблем. Иногда в организациях создают для этой цели специальный "справочный стол", но чаще от пользователей отбивается системный администратор. Очень важно в потоке вопросов уметь выявлять проблемы, связанные с информационной безопасностью. Так, многие трудности пользователей, работающих на персональных компьютерах, могут быть следствием заражения вирусами. Целесообразно фиксировать вопросы пользователей, чтобы выявлять их типичные ошибки и выпускать памятки с рекомендациями для распространенных ситуаций.

Поддержка программного обеспечения - одно из важнейших средств обеспечения целостности информации. Прежде всего, необходимо следить за тем, какое программное обеспечение установлено на компьютерах. Если пользователи будут устанавливать программы по своему усмотрению, это может привести к заражению вирусами, а также появлению утилит, действующих в обход защитных средств.

Второй аспект поддержки программного обеспечения - контроль за отсутствием неавторизованного изменения программ и прав доступа к ним. Сюда же можно отнести поддержку эталонных копий программных систем. Обычно контроль достигается комбинированием средств физического и логического управления доступом, а также использованием утилит проверки и обеспечения целостности.

Конфигурационное управление позволяет контролировать и фиксировать изменения, вносимые в программную конфигурацию. Прежде всего, необходимо застраховаться от случайных или непродуманных модификаций, уметь как минимум возвращаться к прошлой, работающей, версии. Фиксация изменений позволит легко восстановить текущую версию после аварии.

Лучший способ уменьшить количество ошибок в рутинной работе - максимально автоматизировать ее.

Резервное копирование необходимо для восстановления программ и данных после аварий. И здесь целесообразно автоматизировать работу, как минимум, сформировав компьютерное расписание создания полных и инкрементальных копий, а как максимум - воспользовавшись соответствующими программными продуктами. Нужно также наладить размещение копий в безопасном месте, защищенном от несанкционированного доступа, пожаров, протечек, то есть от всего, что может привести к краже или повреждению носителей. Целесообразно иметь несколько экземпляров резервных копий и часть из них хранить вне территории организации, защищаясь, таким образом, от крупных аварий и аналогичных инцидентов.

Время от времени в тестовых целях следует проверять возможность восстановления информации с копий.

Управлять носителями необходимо для обеспечения *физической защиты* и учета дискет, лент, печатных выдач и т.п. Управление носителями должно обеспечивать конфиденциальность, целостность и доступность информации, хранящейся вне компьютерных систем. Под *физической защитой* здесь понимается не только отражение попыток несанкционированного доступа, но и предохранение от вредных влияний окружающей среды (жары, холода, влаги, магнетизма). Управление носителями должно охватывать весь жизненный цикл - от закупки до выведения из эксплуатации.

Документирование - неотъемлемая часть информационной безопасности. В виде документов оформляется почти все - от политики безопасности до журнала учета носителей. Важно, чтобы документация была актуальной, отражала именно текущее состояние дел, причем в непротиворечивом виде.

К хранению одних документов (содержащих, например, анализ уязвимых мест системы и угроз) применимы требования обеспечения конфиденциальности, к другим, таким как план восстановления после аварий - требования целостности и доступности (в критической ситуации план необходимо найти и прочитать).

Регламентные работы - очень серьезная угроза безопасности. Сотрудник, осуществляющий регламентные работы, получает исключительный доступ к системе, и на практике очень трудно проконтролировать, какие именно действия он совершает. Здесь на первый план выходит степень доверия к тем, кто выполняет работу.

**Реагирование на нарушения режима безопасности**

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

***Реакция на нарушения режима безопасности*** преследует три главные цели:

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

В организации должен быть человек, доступный 24 часа в сутки (лично, по телефону, пейджеру или электронной почте), который отвечает за реакцию на нарушения. Все должны знать координаты этого человека и обращаться к нему при первых признаках опасности. В общем, как при пожаре, нужно знать, куда звонить, и что делать до приезда пожарной команды.

Важность быстрой и скоординированной реакции можно продемонстрировать на следующем примере. Пусть локальная сеть предприятия состоит из двух сегментов, администрируемых разными людьми. Далее, пусть в один из сегментов был внесен вирус. Почти наверняка через несколько минут (или, в крайнем случае, несколько десятков минут) вирус распространится и на другой сегмент. Значит, меры нужно принять немедленно. "Вычищать" вирус необходимо одновременно в обоих сегментах; в противном случае сегмент, восстановленный первым, заразится от другого, а затем вирус вернется и во второй сегмент.

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

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

Чтобы предотвратить повторные нарушения, необходимо анализировать каждый инцидент, выявлять причины, накапливать статистику. Каковы источники вредоносного ПО? Какие пользователи имеют обыкновение выбирать слабые пароли? На подобные вопросы и должны дать ответ результаты анализа.

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

**Планирование восстановительных работ**

Ни одна организация не застрахована от серьезных аварий, вызванных естественными причинами, действиями злоумышленника, халатностью или некомпетентностью. В то же время, у каждой организации есть функции, которые руководство считает критически важными, они должны выполняться несмотря ни на что. ***Планирование восстановительных работ*** позволяет подготовиться к авариям, уменьшить ущерб от них и сохранить способность к функционированию хотя бы в минимальном объеме.

Отметим, что меры информационной безопасности можно разделить на три группы, в зависимости от того, направлены ли они на предупреждение, обнаружение или ликвидацию последствий атак. Большинство мер носит предупредительный характер. Оперативный анализ регистрационной информации и некоторые аспекты *реагирования на нарушения* (так называемый активный аудит) служат для обнаружения и отражения атак. *Планирование восстановительных работ*, очевидно, можно отнести к последней из трех перечисленных групп.

Процесс *планирования восстановительных работ* можно разделить на следующие этапы:

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

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

Идентифицируя ресурсы, необходимые для выполнения критически важных функций, следует помнить, что многие из них имеют некомпьютерный характер. На данном этапе желательно подключать к работе специалистов разного профиля, способных в совокупности охватить все аспекты проблемы. Критичные ресурсы обычно относятся к одной из следующих категорий:

- персонал;
- информационная инфраструктура;
- физическая инфраструктура.

Составляя списки ответственных специалистов, следует учитывать, что некоторые из них могут непосредственно пострадать от аварии (например, от пожара), кто-то может находиться в состоянии стресса, часть сотрудников, возможно, будет лишена возможности попасть на работу (например, в случае массовых беспорядков). Желательно иметь некоторый резерв специалистов или заранее определить каналы, по которым можно на время привлечь дополнительный персонал.

Информационная инфраструктура включает в себя следующие элементы:

- компьютеры;
- программы и данные;
- информационные сервисы внешних организаций;
- документацию.

Среди внешних информационных сервисов для коммерческих организаций, вероятно, важнее всего получить оперативную информацию и связь с государственными службами, курирующими данный сектор экономики.

Документация важна хотя бы потому, что не вся информация, с которой работает организация, представлена в электронном виде. Скорее всего, план восстановительных работ напечатан на бумаге.

К физической инфраструктуре относятся здания, инженерные коммуникации, средства связи, оргтехника и многое другое. Компьютерная техника не может работать в плохих условиях, без стабильного электропитания и т.п.

Анализируя критичные ресурсы, целесообразно учесть временной профиль их использования. Большинство ресурсов требуются постоянно, но в некоторых нужда может возникать только в определенные периоды (например, в конце месяца или года при составлении отчета).

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

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

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

Подготовка к реализации выбранной стратегии состоит в выработке плана действий в экстренных ситуациях и по их окончании, а также в обеспечении некоторой избыточности критичных ресурсов. Последнее возможно и без большого расхода средств, если заключить с одной или несколькими организациями соглашения о взаимной поддержке в случае аварий - те, кто не пострадал, предоставляют часть своих ресурсов во временное пользование менее удачливым партнерам.

Избыточность обеспечивается также мерами резервного копирования, хранением копий в нескольких местах, представлением информации в разных видах (на бумаге и в файлах) и т.д.

Имеет смысл заключить соглашение с поставщиками информационных услуг о первоочередном обслуживании в критических ситуациях или заключать соглашения с несколькими поставщиками. Правда, эти меры могут потребовать определенных расходов.

Проверка стратегии производится путем анализа подготовленного плана, принятых и намеченных мер

**Основные программно-технические меры**

**Основные понятия программно-технического уровня информационной безопасности**

Программно-технические меры, то есть меры, направленные на контроль компьютерных сущностей - оборудования, программ и/или данных, образуют последний и самый важный рубеж информационной безопасности.

Следует, однако, учитывать, что быстрое развитие информационных технологий не только предоставляет обороняющимся новые возможности, но и объективно затрудняет

обеспечение надежной защиты, если опираться исключительно на меры программно-технического уровня. Причин тому несколько:

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

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

Следуя объектно-ориентированному подходу, при рассмотрении информационной системы с единичным уровнем детализации мы увидим совокупность предоставляемых ею *информационных сервисов*. Назовем их ***основными***. Чтобы они могли функционировать и обладали требуемыми свойствами, необходимо несколько уровней *дополнительных (вспомогательных) сервисов* - от СУБД и мониторов транзакций до ядра операционной системы и оборудования.

К ***вспомогательным*** относятся сервисы безопасности (мы уже сталкивались с ними при рассмотрении стандартов и спецификаций в области ИБ); среди них нас в первую очередь будут интересовать универсальные, высокоуровневые, допускающие использование различными *основными* и *вспомогательными сервисами*. Далее мы рассмотрим следующие сервисы:

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

Совокупность перечисленных выше *сервисов безопасности* называется полным набором. Считается, что его, в принципе, достаточно для построения надежной защиты на программно-техническом уровне, правда, при соблюдении целого ряда дополнительных условий (отсутствие уязвимых мест, безопасное администрирование и т.д.).

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

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

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

**Особенности современных информационных систем, существенные с точки зрения безопасности**

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

С точки зрения безопасности наиболее существенными представляются следующие аспекты современных ИС:

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

**Архитектурная безопасность**

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

Три принципа архитектурной безопасности:

- необходимость выработки и проведения в жизнь единой политики безопасности;

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

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

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

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

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

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

Принцип невозможности перехода в небезопасное состояние означает, что при любых обстоятельствах, в том числе нештатных, защитное средство либо полностью выполняет свои функции, либо полностью блокирует доступ.

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

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

Принцип эшелонированности обороны предписывает не полагаться на один защитный рубеж, каким бы надежным он ни казался. За средствами физической защиты должны следовать программно-технические средства, за *идентификацией* и *аутентификацией* - *управление доступом* и, как последний рубеж, - *протоколирование* и *аудит*. Эшелонированная оборона способна, по крайней мере, задержать злоумышленника, а благодаря наличию такого рубежа,

как *протоколирование* и *аудит*, его действия не останутся незамеченными. Принцип разнообразия защитных средств предполагает создание различных по своему характеру оборонительных рубежей, чтобы от потенциального злоумышленника требовалось овладение разнообразными и, по возможности, несовместимыми между собой навыками.

Очень важен принцип простоты и управляемости информационной системы в целом и защитных средств в особенности. Только для простого защитного средства можно формально или неформально доказать его корректность. Только в простой и управляемой системе можно проверить согласованность конфигурации различных компонентов и осуществлять централизованное администрирование. В этой связи важно отметить интегрирующую роль Web-сервиса, скрывающего разнообразие обслуживаемых объектов и предоставляющего единый, наглядный интерфейс. Соответственно, если объекты некоторого вида (например, таблицы базы данных) доступны через Web, необходимо заблокировать прямой доступ к ним, поскольку в противном случае система будет сложной и плохо управляемой.

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

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

Еще один важный архитектурный принцип - минимизация объема защитных средств, выносимых на клиентские системы. Причин тому несколько:

- для доступа в корпоративную сеть могут использоваться **потребительские устройства** с ограниченной функциональностью;
- конфигурацию клиентских систем трудно или невозможно контролировать.

К необходимому минимуму следует отнести реализацию *сервисов безопасности* на сетевом и транспортном уровнях и поддержку механизмов *аутентификации*, устойчивых к сетевым угрозам.

**Идентификация и аутентификация, управление доступом**

**Идентификация и аутентификация**

**Основные понятия**

*Идентификацию* и *аутентификацию* можно считать основой программно-технических средств безопасности, поскольку остальные сервисы рассчитаны на обслуживание именованных субъектов. *Идентификация* и *аутентификация* – это первая линия обороны, "проходная" информационного пространства организации.

***Идентификация*** позволяет субъекту (пользователю, процессу, действующему от имени определенного пользователя, или иному аппаратно-программному компоненту) назвать себя (сообщить свое имя). Посредством ***аутентификации*** вторая сторона убеждается, что субъект действительно тот, за кого он себя выдает. В качестве синонима слова "*аутентификация*" иногда используют словосочетание "проверка подлинности".

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

В сетевой среде, когда стороны *идентификации*/*аутентификации* территориально разнесены, у рассматриваемого сервиса есть два основных аспекта:

- что служит ***аутентификатором*** (то есть используется для подтверждения подлинности субъекта);
- как организован (и защищен) обмен данными *идентификации*/*аутентификации*.

Субъект может подтвердить свою подлинность, предъявив по крайней мере одну из следующих сущностей:

- нечто, что он знает (пароль, личный *идентификационный* номер, криптографический ключ и т.п.);
- нечто, чем он владеет (личную карточку или иное устройство аналогичного назначения);
- нечто, что есть часть его самого (голос, отпечатки пальцев и т.п., то есть свои биометрические характеристики).

В открытой сетевой среде между сторонами *идентификации*/*аутентификации* не существует доверенного маршрута; это значит, что в общем случае данные, переданные субъектом, могут не совпадать с данными, полученными и использованными для проверки подлинности. Необходимо обеспечить защиту от пассивного и активного прослушивания сети, то есть от ***перехвата***, ***изменения*** и/или ***воспроизведения*** данных. Передача паролей в открытом виде, очевидно, неудовлетворительна; не спасает положение и шифрование паролей, так как оно не защищает от *воспроизведения*. Нужны более сложные протоколы *аутентификации*.

Надежная *идентификация* затруднена не только из-за сетевых угроз, но и по целому ряду причин. Во-первых, почти все *аутентификационные* сущности можно узнать, украсть или подделать. Во-вторых, имеется противоречие между надежностью *аутентификации*, с одной стороны, и удобствами пользователя и системного администратора с другой. Так, из соображений безопасности необходимо с определенной частотой просить пользователя повторно вводить *аутентификационную* информацию (ведь на его место мог сесть другой человек), а это не только хлопотно, но и повышает вероятность того, что кто-то может подсмотреть за вводом данных. В-третьих, чем надежнее средство защиты, тем оно дороже.

Современные средства *идентификации*/*аутентификации* должны поддерживать концепцию *единого входа в сеть*. *Единый вход в сеть* – это, в первую очередь, требование удобства для пользователей. Если в корпоративной сети много информационных сервисов, допускающих независимое обращение, то многократная *идентификация*/*аутентификация* становится слишком обременительной. К сожалению, пока нельзя сказать, что *единый вход в сеть* стал нормой, доминирующие решения пока не сформировались.

Таким образом, необходимо искать компромисс между надежностью, доступностью по цене и удобством использования и администрирования средств *идентификации* и *аутентификации*.

Следует отметить, что сервис *идентификации* / *аутентификации* может стать объектом атак на доступность. Если система сконфигурирована так, что после определенного числа неудачных попыток устройство ввода идентификационной информации (такое, например, как терминал) блокируется, то злоумышленник может остановить работу легального пользователя буквально несколькими нажатиями клавиш.

**Парольная аутентификация**

Главное достоинство ***парольной аутентификации*** – простота и привычность. Пароли давно встроены в операционные системы и иные сервисы. При правильном использовании пароли могут обеспечить приемлемый для многих организаций уровень безопасности. Тем не менее, по совокупности характеристик их следует признать самым слабым средством проверки подлинности.

Чтобы пароль был запоминающимся, его зачастую делают простым. Однако простой пароль нетрудно угадать, особенно если знать пристрастия данного пользователя.

Иногда пароли с самого начала не хранятся в тайне, так как имеют стандартные значения, указанные в документации, и далеко не всегда после установки системы производится их смена.

Ввод пароля можно подсмотреть. Пароли нередко сообщают коллегам.

Пароль можно угадать "методом грубой силы", используя, скажем, словарь. Если файл паролей зашифрован, но доступен для чтения, его можно скачать к себе на компьютер и попытаться подобрать пароль, запрограммировав полный перебор (предполагается, что алгоритм шифрования известен).

Тем не менее, следующие меры позволяют значительно повысить надежность парольной защиты:

- ***наложение технических ограничений*** (пароль должен быть не слишком коротким, он должен содержать буквы, цифры, знаки пунктуации и т.п.);
- ***управление сроком действия паролей***, их периодическая смена;
- ограничение доступа к файлу паролей;
- ограничение числа неудачных попыток входа в систему (это затруднит применение "метода грубой силы");
- обучение пользователей;
- использование программных ***генераторов паролей*** (такая программа, основываясь на несложных правилах, может порождать только благозвучные и, следовательно, запоминающиеся пароли).
- Перечисленные меры целесообразно применять всегда, даже если наряду с паролями используются другие методы аутентификации.

**Одноразовые пароли**

Рассмотренные выше **пароли можно назвать многоразовыми**; их раскрытие позволяет злоумышленнику действовать от имени легального пользователя. Гораздо более сильным средством, устойчивым к пассивному прослушиванию сети, являются **одноразовые пароли**.

Наиболее известным программным *генератором одноразовых паролей* является система **S/KEY** компании Bellcore. Идея этой системы состоит в следующем. Пусть имеется **односторонняя функция** f (то есть функция, вычислить обратную которой за приемлемое время не представляется возможным). Эта функция известна и пользователю, и **серверу аутентификации**. Пусть, далее, имеется **секретный ключ** K, известный только пользователю.

На этапе начального администрирования пользователя функция f применяется к ключу K n раз, после чего результат сохраняется на сервере. После этого процедура проверки подлинности пользователя выглядит следующим образом:

- сервер присылает на пользовательскую систему число (n-1);
- пользователь применяет функцию f к секретному ключу K (n-1) раз и отправляет результат по сети на сервер аутентификации;
- сервер применяет функцию f к полученному от пользователя значению и сравнивает результат с ранее сохраненной величиной. В случае совпадения подлинность пользователя считается установленной, сервер запоминает новое значение (присланное пользователем) и уменьшает на единицу счетчик (n).

На самом деле реализация устроена чуть сложнее (кроме счетчика, сервер посылает затравочное значение, используемое функцией f), но для нас сейчас это не важно. Поскольку функция f необратима, *перехват* пароля, равно как и получение доступа к серверу *аутентификации*, не позволяют узнать секретный ключ K и предсказать следующий одноразовый пароль.

Система S/KEY имеет статус Internet-стандарта (RFC 1938).

Другой подход к надежной *аутентификации* состоит в *генерации нового пароля* через небольшой промежуток времени (например, каждые 60 секунд), для чего могут использоваться программы или специальные интеллектуальные карты (с практической точки зрения такие пароли можно считать одноразовыми). Серверу *аутентификации* должен быть известен

алгоритм *генерации паролей* и ассоциированные с ним параметры; кроме того, часы клиента и сервера должны быть синхронизированы.

**Сервер аутентификации Kerberos**

**Kerberos** – это программный продукт, разработанный в середине 1980-х годов в Массачусетском технологическом институте и претерпевший с тех пор ряд принципиальных изменений. Клиентские компоненты Kerberos присутствуют в большинстве современных операционных систем.

Kerberos предназначен для решения следующей задачи. Имеется открытая (незащищенная) сеть, в узлах которой сосредоточены **субъекты** – пользователи, а также клиентские и серверные программные системы. Каждый субъект обладает секретным ключом. Чтобы субъект C мог доказать свою подлинность субъекту S (без этого S не станет обслуживать C), он должен не только назвать себя, но и продемонстрировать знание секретного ключа. C не может просто послать S свой секретный ключ, во-первых, потому, что сеть открыта (доступна для пассивного и активного прослушивания), а, во-вторых, потому, что S не знает (и не должен знать) секретный ключ C. Требуется менее прямолинейный способ демонстрации знания секретного ключа.

Система Kerberos представляет собой **доверенную третью сторону** (то есть сторону, которой доверяют все), владеющую секретными ключами обслуживаемых субъектов и помогающую им в попарной проверке подлинности.

Чтобы с помощью Kerberos получить доступ к S (обычно это сервер), C (как правило – клиент) посылает Kerberos запрос, содержащий сведения о нем (клиенте) и о запрашиваемой услуге. В ответ Kerberos возвращает так называемый **билет**, зашифрованный секретным ключом сервера, и копию части информации из билета, зашифрованную секретным ключом клиента. Клиент должен расшифровать вторую порцию данных и переслать ее вместе с билетом серверу. Сервер, расшифровав билет, может сравнить его содержимое с дополнительной информацией, присланной клиентом. Совпадение свидетельствует о том, что клиент смог расшифровать предназначенные ему данные (ведь содержимое билета никому, кроме сервера и Kerberos, недоступно), то есть продемонстрировал знание секретного ключа. Значит, клиент – именно тот, за кого себя выдает. Подчеркнем, что секретные ключи в процессе проверки подлинности не передавались по сети (даже в зашифрованном виде) – они только использовались для шифрования. Как организован первоначальный обмен ключами между Kerberos и субъектами и как субъекты хранят свои секретные ключи – вопрос отдельный.

Проиллюстрируем описанную процедуру.

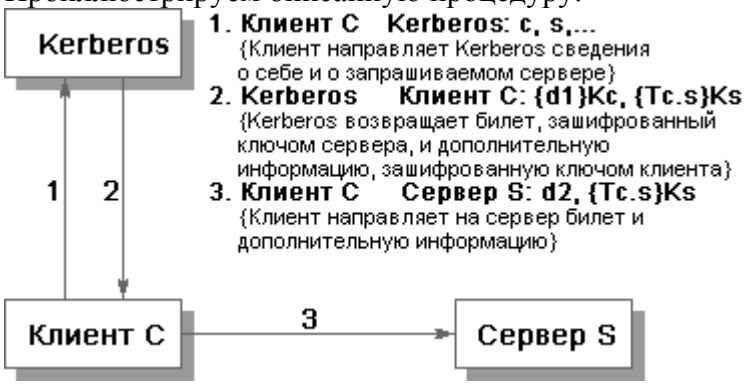


**Рис. 10.1.** Проверка сервером S подлинности клиента C.

Здесь c и s – сведения (например, имя), соответственно, о клиенте и сервере, d1 и d2 – дополнительная (по отношению к билету) информация, Tc.s – билет для клиента C на обслуживание у сервера S, Kc и Ks – секретные ключи клиента и сервера, {info}K – информация info, зашифрованная ключом K.

Приведенная схема – крайне упрощенная версия реальной процедуры проверки подлинности. Kerberos не только устойчив к сетевым угрозам, но и поддерживает концепцию *единого входа в сеть*.

**Идентификация/аутентификация с помощью биометрических данных**

**Биометрия** представляет собой совокупность автоматизированных методов *идентификации* и/или *аутентификации* людей на основе их физиологических и поведенческих характеристик. К числу физиологических характеристик принадлежат особенности **отпечатков пальцев**, **сетчатки** и **роговицы** глаз, **геометрия руки и лица** и т.п. К поведенческим характеристикам относятся **динамика подписи** (ручной), стиль **работы с клавиатурой**. На стыке физиологии и поведения находятся анализ особенностей **голоса** и **распознавание речи**.

В общем виде работа с биометрическими данными организована следующим образом. Сначала создается и поддерживается база данных характеристик потенциальных пользователей. Для этого биометрические характеристики пользователя снимаются, обрабатываются, и результат обработки (называемый **биометрическим шаблоном**) заносится в базу данных (исходные данные, такие как результат сканирования пальца или роговицы, обычно не хранятся).

В дальнейшем для *идентификации* (и одновременно *аутентификации*) пользователя процесс снятия и обработки повторяется, после чего производится поиск в базе данных шаблонов. В случае успешного поиска личность пользователя и ее подлинность считаются установленными. Для аутентификации достаточно произвести сравнение с одним биометрическим шаблоном, выбранным на основе предварительно введенных данных.

Обычно биометрию применяют вместе с другими *аутентификаторами*, такими, например, как интеллектуальные карты. Иногда биометрическая аутентификация является лишь первым рубежом защиты и служит для активизации интеллектуальных карт, хранящих криптографические секреты; в таком случае биометрический шаблон хранится на той же карте.

Но главная опасность состоит в том, что любая "пробоина" для биометрии оказывается фатальной. Пароли, при всей их ненадежности, в крайнем случае можно сменить. Утерянную аутентификационную карту можно аннулировать и завести новую. Палец же, глаз или голос сменить нельзя. Если биометрические данные окажутся скомпрометированы, придется как минимум производить существенную модернизацию всей системы.

**Управление доступом**

**Основные понятия**

С традиционной точки зрения средства управления доступом позволяют специфицировать и контролировать действия, которые субъекты (пользователи и процессы) могут выполнять над **объектами** (информацией и другими компьютерными ресурсами). В данном разделе речь идет о логическом управлении доступом, которое, в отличие от физического, реализуется программными средствами. Логическое управление доступом – это основной механизм многопользовательских систем, призванный обеспечить конфиденциальность и целостность объектов и, до некоторой степени, их доступность (путем запрещения обслуживания неавторизованных пользователей).

Рассмотрим формальную постановку задачи в традиционной трактовке. Имеется совокупность субъектов и набор объектов. Задача логического управления доступом состоит в том, чтобы для каждой пары "субъект-объект" определить множество допустимых операций (зависящее, быть может, от некоторых дополнительных условий) и контролировать выполнение установленного порядка.

Отношение "субъекты-объекты" можно представить в виде **матрицы доступа**, в строках которой перечислены субъекты, в столбцах – объекты, а в клетках, расположенных на пересечении строк и столбцов, записаны дополнительные условия (например, время и место действия) и разрешенные виды доступа. Фрагмент матрицы может выглядеть, например, так:

**Таблица 10.1. Фрагмент матрицы доступа**

| | **Файл** | **Программа** | **Линия связи** | **Реляционная таблица** |
|---|---|---|---|---|
| Пользователь 1 | orw с системной консоли | e | rw с 8:00 до 18:00 | |

| | | | | |
|---|---|---|---|---|
| Пользователь 2 | | | | a |

"o" – обозначает разрешение на передачу **прав доступа** другим пользователям,
"r" – чтение,
"w" – запись,
"e" – выполнение,
"a" – добавление информации

Тема логического управления доступом – одна из сложнейших в области информационной безопасности. Дело в том, что само понятие объекта (а тем более видов доступа) меняется от сервиса к сервису. Для операционной системы к объектам относятся файлы, устройства и процессы. Применительно к файлам и устройствам обычно рассматриваются права на чтение, запись, выполнение (для программных файлов), иногда на удаление и добавление. Отдельным правом может быть возможность передачи полномочий доступа другим субъектам (так называемое право владения). Процессы можно создавать и уничтожать. Современные операционные системы могут поддерживать и другие объекты.

Для систем управления реляционными базами данных объект – это база данных, таблица, представление, хранимая процедура. К таблицам применимы операции поиска, добавления, модификации и удаления данных, у других объектов иные виды доступа.

Разнообразие объектов и применимых к ним операций приводит к принципиальной децентрализации логического управления доступом. Каждый сервис должен сам решать, позволить ли конкретному субъекту ту или иную операцию. Теоретически это согласуется с современным объектно-ориентированным подходом, на практике же приводит к значительным трудностям. Главная проблема в том, что ко многим объектам можно получить доступ с помощью разных сервисов (возможно, при этом придется преодолеть некоторые технические трудности). Так, до реляционных таблиц можно добраться не только средствами СУБД, но и путем непосредственного чтения файлов или дисковых разделов, поддерживаемых операционной системой (разобравшись предварительно в структуре хранения объектов базы данных). В результате при задании матрицы доступа нужно принимать во внимание не только принцип распределения привилегий для каждого сервиса, но и существующие связи между сервисами (приходится заботиться о согласованности разных частей матрицы). Аналогичная трудность возникает при экспорте/импорте данных, когда информация о правах доступа, как правило, теряется (поскольку на новом сервисе она не имеет смысла). Следовательно, обмен данными между различными сервисами представляет особую опасность с точки зрения управления доступом, а при проектировании и реализации разнородной конфигурации необходимо позаботиться о согласованном распределении прав доступа субъектов к объектам и о минимизации числа способов экспорта/импорта данных.

При принятии решения о предоставлении доступа обычно анализируется следующая информация:

- идентификатор субъекта (идентификатор пользователя, сетевой адрес компьютера и т.п.). Подобные идентификаторы являются основой **произвольного (или дискреционного) управления доступом**;
- атрибуты субъекта (метка безопасности, группа пользователя и т.п.). Метки безопасности – основа **принудительного (мандатного) управления доступом**.

Матрицу доступа, ввиду ее разреженности (большинство клеток – пустые), неразумно хранить в виде двухмерного массива. Обычно ее хранят по столбцам, то есть для каждого объекта поддерживается список "допущенных" субъектов вместе с их правами. Элементами списков могут быть имена групп и шаблоны субъектов, что служит большим подспорьем администратору. Некоторые проблемы возникают только при удалении субъекта, когда приходится удалять его имя из всех списков доступа; впрочем, эта операция производится нечасто.

Списки доступа – исключительно гибкое средство. С их помощью легко выполнить требование о гранулярности прав с точностью до пользователя. Посредством списков несложно

добавить права или явным образом запретить доступ (например, чтобы наказать нескольких членов группы пользователей). Безусловно, списки являются лучшим средством произвольного управления доступом.

Подавляющее большинство операционных систем и систем управления базами данных реализуют именно произвольное управление доступом. Основное достоинство произвольного управления – гибкость. Вообще говоря, для каждой пары "субъект-объект" можно независимо задавать права доступа (особенно легко это делать, если используются **списки управления доступом**).

Удобной надстройкой над средствами логического управления доступом является **ограничивающий интерфейс**, когда пользователя лишают самой возможности попытаться совершить несанкционированные действия, включив в число видимых ему объектов только те, к которым он имеет доступ. Подобный подход обычно реализуют в рамках системы меню (пользователю показывают лишь допустимые варианты выбора) или посредством ограничивающих оболочек, таких как restricted shell в ОС Unix.

В заключение подчеркнем важность управления доступом не только на уровне операционной системы, но и в рамках других сервисов, входящих в состав современных приложений, а также, насколько это возможно, на "стыках" между сервисами. Здесь на первый план выходит существование единой политики безопасности организации, а также квалифицированное и согласованное системное администрирование.

**Ролевое управление доступом**

При большом количестве пользователей традиционные подсистемы управления доступом становятся крайне сложными для администрирования. Число связей в них пропорционально произведению количества пользователей на количество объектов. Необходимы решения в объектно-ориентированном стиле, способные эту сложность понизить.

Таким решением является **ролевое управление доступом** (РУД). Суть его в том, что между пользователями и их привилегиями появляются промежуточные сущности – роли. Для каждого пользователя одновременно могут быть активными несколько ролей, каждая из которых дает ему определенные права (см. рис).


**Рис.** Пользователи, объекты и роли.

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

Ролевое управление доступом оперирует следующими основными понятиями:

- **пользователь** (человек, интеллектуальный автономный агент и т.п.);
- **сеанс работы пользователя**;
- **роль** (обычно определяется в соответствии с организационной структурой);

- **объект** (сущность, доступ к которой разграничивается; например, файл ОС или таблица СУБД);
- **операция** (зависит от объекта; для файлов ОС – чтение, запись, выполнение и т.п.; для таблиц СУБД – вставка, удаление и т.п., для прикладных объектов операции могут быть более сложными);
- право доступа (разрешение выполнять определенные операции над определенными объектами).

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

Между ролями может быть определено отношение частичного порядка, называемое наследованием. Если роль r2 является наследницей r1, то все права r1 приписываются r2, а все пользователи r2 приписываются r1. Очевидно, что **наследование ролей** соответствует наследованию классов в объектно-ориентированном программировании, только правам доступа соответствуют методы классов, а пользователям – объекты (экземпляры) классов.

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

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


**Рис.** Фрагмент иерархии ролей.

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

**Статическое разделение обязанностей** налагает ограничения на **приписывание пользователей ролям**. В простейшем случае членство в некоторой роли запрещает приписывание пользователя определенному множеству других ролей. В общем случае данное ограничение задается как пара "множество ролей – число" (где множество состоит, по крайней мере, из двух ролей, а число должно быть больше 1), так что никакой пользователь не может быть приписан указанному (или большему) числу ролей из заданного множества. Например, может существовать пять бухгалтерских ролей, но политика безопасности допускает членство не более чем в двух таких ролях (здесь число=3).

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

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

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

- **Административные функции** (создание и сопровождение ролей и других атрибутов ролевого доступа): создать/удалить роль/пользователя, приписать пользователя/право роли или ликвидировать существующую ассоциацию, создать/удалить отношение наследования между существующими ролями, создать новую роль и сделать ее наследницей/предшественницей существующей роли, создать/удалить ограничения для статического/динамического разделения обязанностей.
- **Вспомогательные функции** (обслуживание сеансов работы пользователей): открыть сеанс работы пользователя с активацией подразумеваемого набора ролей; активировать новую роль, деактивировать роль; проверить правомерность доступа.
- **Информационные функции** (получение сведений о текущей конфигурации с учетом отношения наследования). Здесь проводится разделение на обязательные и необязательные функции. К числу первых принадлежат получение списка пользователей, приписанных роли, и списка ролей, которым приписан пользователь.

Все остальные функции отнесены к разряду необязательных. Это получение информации о правах, приписанных роли, о правах заданного пользователя (которыми он обладает как член множества ролей), об активных в данный момент сеанса ролях и правах, об операциях, которые роль/пользователь правомочны совершить над заданным объектом, о статическом/динамическом разделении обязанностей.

Можно надеяться, что предлагаемый стандарт поможет сформировать единую терминологию и, что более важно, позволит оценивать РУД-продукты с единых позиций, по единой шкале.

**Протоколирование и аудит, шифрование, контроль целостности**

**Протоколирование и аудит**

**Основные понятия**

Под ***протоколированием*** понимается сбор и накопление информации о *событиях*, происходящих в информационной системе. У каждого *сервиса* свой набор возможных событий, но в любом случае их можно разделить на внешние (вызванные действиями других сервисов), внутренние (вызванные действиями самого сервиса) и клиентские (вызванные действиями пользователей и администраторов).

***Аудит*** – это анализ накопленной информации, проводимый оперативно, в реальном времени или периодически (например, раз в день). Оперативный аудит с автоматическим реагированием на выявленные нештатные ситуации называется активным.

Реализация протоколирования и аудита решает следующие задачи:

- обеспечение *подотчетности* пользователей и администраторов;
- обеспечение возможности реконструкции последовательности событий;
- обнаружение попыток нарушений информационной безопасности;
- предоставление информации для выявления и анализа проблем.

При протоколировании события рекомендуется записывать, по крайней мере, следующую информацию:

- дата и время события;

- уникальный идентификатор пользователя – инициатора действия;
- тип события;
- результат действия (успех или неудача);
- источник запроса (например, имя терминала);
- имена затронутых объектов (например, открываемых или удаляемых файлов);
- описание изменений, внесенных в базы данных защиты (например, новая метка безопасности объекта).

Характерная особенность протоколирования и аудита – зависимость от других средств безопасности. Идентификация и аутентификация служат отправной точкой подотчетности пользователей, логическое управление доступом защищает конфиденциальность и целостность *регистрационной информации*. Возможно, для защиты привлекаются и криптографические методы.

Возвращаясь к целям протоколирования и аудита, отметим, что обеспечение подотчетности важно в первую очередь как сдерживающее средство.

Реконструкция последовательности событий позволяет выявить слабости в защите сервисов, найти виновника вторжения, оценить масштабы причиненного ущерба и вернуться к нормальной работе.

Обнаружение попыток нарушений информационной безопасности – функция активного аудита, о котором пойдет речь в следующем разделе. Обычный аудит позволяет выявить подобные попытки с опозданием, но и это оказывается полезным.

Выявление и анализ проблем могут помочь улучшить такой параметр безопасности, как доступность. Обнаружив узкие места, можно попытаться переконфигурировать или перенастроить систему, снова измерить производительность и т.д.

Непросто осуществить организацию согласованного протоколирования и аудита в распределенной разнородной системе. Во-первых, некоторые компоненты, важные для безопасности (например, маршрутизаторы), могут не обладать своими ресурсами протоколирования; в таком случае их нужно экранировать другими сервисами, которые возьмут протоколирование на себя. Во-вторых, необходимо увязывать между собой события в разных сервисах.

**Активный аудит**

**Основные понятия**

Под ***подозрительной активностью*** понимается поведение пользователя или компонента информационной системы, являющееся ***злоумышленным*** (в соответствии с заранее определенной политикой безопасности) или нетипичным (согласно принятым критериям).

Задача активного аудита – оперативно выявлять подозрительную активность и предоставлять средства для *автоматического реагирования* на нее.

Активность, не соответствующую политике безопасности, целесообразно разделить на *атаки*, направленные на незаконное получение полномочий, и на действия, выполняемые в рамках имеющихся полномочий, но нарушающие политику безопасности.

Атаки нарушают любую осмысленную политику безопасности. Иными словами, активность атакующего является разрушительной независимо от политики. Следовательно, для описания и выявления атак можно применять универсальные методы, инвариантные относительно политики безопасности, такие как сигнатуры и их обнаружение во входном потоке событий с помощью аппарата *экспертных систем*.

***Сигнатура атаки*** – это совокупность условий, при выполнении которых атака считается имеющей место, что вызывает заранее определенную реакцию. Простейший пример сигнатуры – "зафиксированы три последовательные неудачные попытки входа в систему с одного терминала", пример ассоциированной реакции – блокирование терминала до прояснения ситуации.

Действия, выполняемые в рамках имеющихся полномочий, но нарушающие политику безопасности, мы будем называть злоупотреблением полномочиями. *Злоупотребления полномочиями* возможны из-за неадекватности средств разграничения доступа выбранной политике безопасности. Простейшим примером злоупотреблений является неэтичное поведение суперпользователя, просматривающего личные файлы других пользователей. Анализируя регистрационную информацию, можно обнаружить подобные события и сообщить о них *администратору безопасности*, хотя для этого необходимы соответствующие средства выражения политики безопасности.

Нетипичное *поведение* выявляется статистическими методами. В простейшем случае применяют систему *порогов*, превышение которых является подозрительным. В более развитых системах производится сопоставление долговременных характеристик работы (называемых *долгосрочным профилем*) с *краткосрочными профилями*.

Применительно к средствам активного аудита различают *ошибки первого* и *второго рода*: *пропуск атак* и *ложные тревоги*, соответственно. Нежелательность ошибок первого рода очевидна; ошибки второго рода не менее неприятны, поскольку отвлекают администратора безопасности от действительно важных дел, косвенно способствуя пропуску атак.

Достоинства сигнатурного метода – высокая производительность, малое число ошибок второго рода, обоснованность решений. Основной недостаток – неумение обнаруживать неизвестные атаки и вариации известных атак.

Основные достоинства статистического подхода – универсальность и обоснованность решений, потенциальная способность обнаруживать неизвестные атаки, то есть минимизация числа ошибок первого рода. Минусы заключаются в относительно высокой доле ошибок второго рода, плохой работе в случае, когда неправомерное поведение является типичным, когда типичное поведение плавно меняется от легального к неправомерному, а также в случаях, когда типичного поведения нет (как показывает статистика, таких пользователей примерно 5-10%).

Средства активного аудита могут располагаться на всех линиях обороны информационной системы. На границе контролируемой зоны они могут обнаруживать подозрительную активность в точках подключения к внешним сетям (не только попытки нелегального проникновения, но и действия по "прощупыванию" сервисов безопасности). В корпоративной сети, в рамках информационных сервисов и сервисов безопасности, активный аудит в состоянии обнаружить и пресечь подозрительную активность внешних и внутренних пользователей, выявить проблемы в работе сервисов, вызванные как нарушениями безопасности, так и аппаратно-программными ошибками. Важно отметить, что активный аудит, в принципе, способен обеспечить защиту от атак на доступность.

**Функциональные компоненты и архитектура**

В составе средств активного аудита можно выделить следующие функциональные компоненты:

- компоненты генерации регистрационной информации. Они находятся на стыке между средствами активного аудита и контролируемыми объектами;
- компоненты хранения сгенерированной регистрационной информации;
- компоненты извлечения регистрационной информации (*сенсоры*). Обычно различают сетевые и хостовые сенсоры, имея в виду под первыми выделенные компьютеры, сетевые карты которых установлены в режим прослушивания, а под вторыми – программы, читающие регистрационные журналы операционной системы. На наш взгляд, с развитием коммутационных технологий это различие постепенно стирается, так как сетевые сенсоры приходится устанавливать в активном сетевом оборудовании и, по сути, они становятся частью сетевой ОС;
- компоненты просмотра регистрационной информации. Могут помочь при принятии решения о реагировании на подозрительную активность;

- компоненты *анализа* информации, поступившей от сенсоров. В соответствии с данным выше определением средств активного аудита, выделяют пороговый анализатор, анализатор нарушений политики безопасности, экспертную систему, выявляющую сигнатуры атак, а также статистический анализатор, обнаруживающий нетипичное поведение;
- компоненты хранения информации, участвующей в анализе. Такое хранение необходимо, например, для выявления атак, протяженных во времени;
- компоненты принятия решений и реагирования ("*решатели*"). "Решатель" может получать информацию не только от локальных, но и от внешних анализаторов, проводя так называемый корреляционный анализ распределенных событий;
- компоненты хранения информации о контролируемых объектах. Здесь могут храниться как пассивные данные, так и методы, необходимые, например, для извлечения из объекта регистрационной информации или для реагирования;
- компоненты, играющие роль организующей оболочки для менеджеров активного аудита, называемые мониторами и объединяющие анализаторы, "решатели", хранилище описаний объектов и *интерфейсные* компоненты. В число последних входят компоненты интерфейса с другими мониторами, как равноправными, так и входящими в иерархию. Такие интерфейсы необходимы, например, для выявления распределенных, широкомасштабных атак;
- компоненты интерфейса с администратором безопасности.

Средства активного аудита строятся в архитектуре *менеджер*/*агент*. Основными агентскими компонентами являются сенсоры. Анализ, принятие решений – функции менеджеров. Очевидно, между менеджерами и агентами должны быть сформированы доверенные каналы.

Подчеркнем важность интерфейсных компонентов. Они полезны как с внутренней для средств активного аудита точки зрения (обеспечивают расширяемость, подключение компонентов различных производителей), так и с внешней точки зрения. Между менеджерами (между компонентами анализа и "решателями") могут существовать горизонтальные связи, необходимые для анализа распределенной активности. Возможно также формирование иерархий средств активного аудита с вынесением на верхние уровни информации о наиболее масштабной и опасной активности.

Обратим также внимание на архитектурную общность средств активного аудита и управления, являющуюся следствием общности выполняемых функций. Продуманные интерфейсные компоненты могут существенно облегчить совместную работу этих средств.

**Шифрование**

Криптография необходима для реализации, по крайней мере, трех сервисов безопасности:

- шифрование;
- контроль целостности;
- аутентификация (этот сервис был рассмотрен нами ранее).

***Шифрование*** – наиболее мощное средство обеспечения конфиденциальности. Во многих отношениях оно занимает центральное место среди программно-технических регуляторов безопасности, являясь основой реализации многих из них, и в то же время последним (а подчас и единственным) защитным рубежом. Например, для портативных компьютеров только шифрование позволяет обеспечить конфиденциальность данных даже в случае кражи.

В большинстве случаев и шифрование, и контроль целостности играют глубоко инфраструктурную роль, оставаясь прозрачными и для приложений, и для пользователей. Типичное место этих сервисов безопасности – на сетевом и транспортном уровнях реализации стека сетевых протоколов.

Различают два основных метода *шифрования*: *симметричный* и *асимметричный*. В первом из них один и тот же ключ (хранящийся в секрете) используется и для зашифрования, и для *расшифрования* данных. Разработаны весьма эффективные (быстрые и надежные) методы симметричного шифрования. Существует и национальный стандарт на подобные методы – ГОСТ 28147-89 "Системы обработки информации. Защита криптографическая. Алгоритм криптографического преобразования".

Рис. иллюстрирует использование симметричного шифрования. Для определенности мы будем вести речь о защите сообщений, хотя события могут развиваться не только в пространстве, но и во времени, когда зашифровываются и расшифровываются никуда не перемещающиеся файлы.

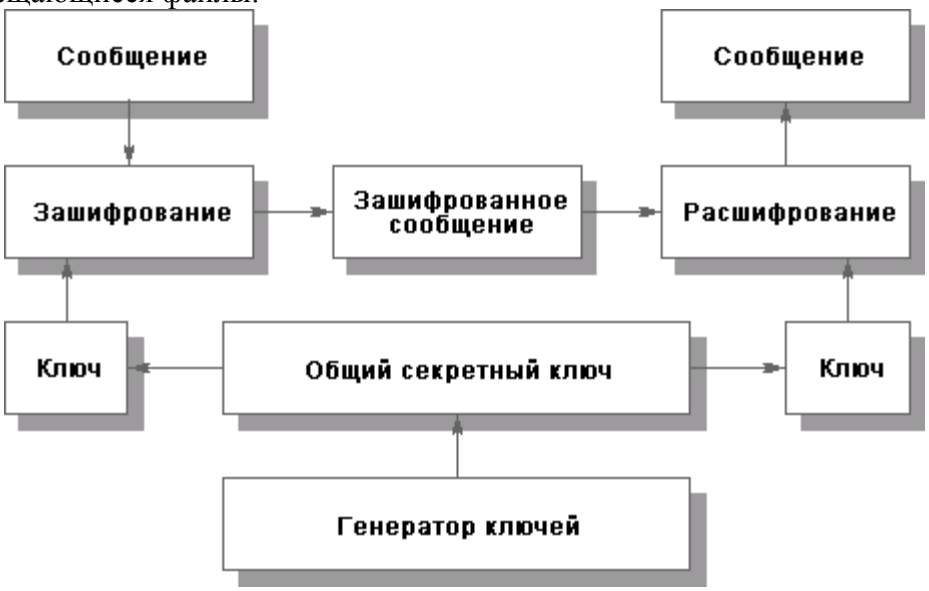


**Рис.** Использование симметричного метода шифрования.

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

В асимметричных методах используются два ключа. Один из них, несекретный (он может публиковаться вместе с другими открытыми сведениями о пользователе), применяется для шифрования, другой (секретный, известный только получателю) – для расшифрования. Самым популярным из асимметричных является метод RSA (Райвест, Шамир, Адлеман), основанный на операциях с большими (скажем, 100-значными) простыми числами и их произведениями.

Проиллюстрируем использование асимметричного шифрования (см. рис.).

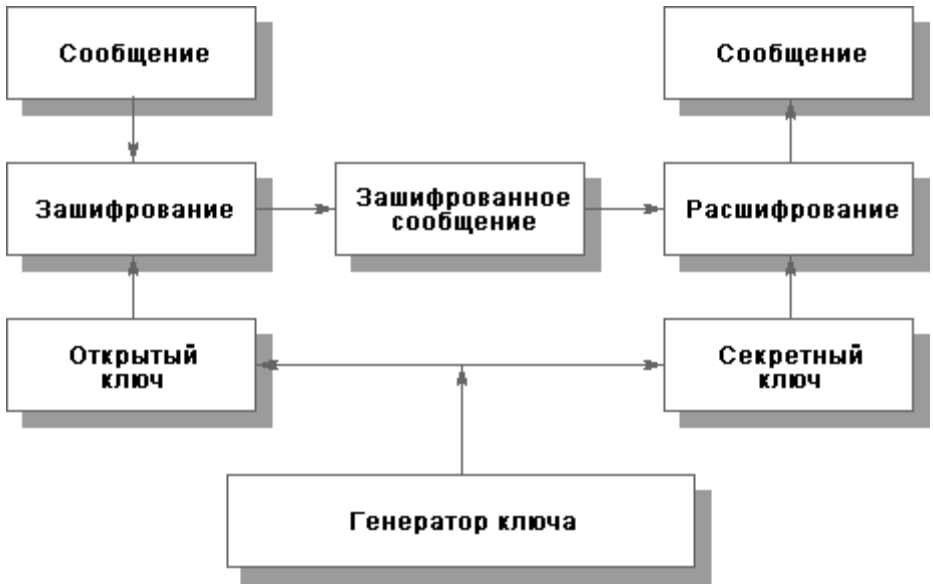


**Рис.** Использование асимметричного метода шифрования.

Существенным недостатком асимметричных методов шифрования является их низкое быстродействие, поэтому данные методы приходится сочетать с симметричными (асимметричные методы на 3 – 4 порядка медленнее). Так, для решения задачи эффективного шифрования с передачей секретного ключа, использованного отправителем, сообщение сначала симметрично зашифровывают случайным ключом, затем этот ключ зашифровывают *открытым* асимметричным *ключом* получателя, после чего сообщение и ключ отправляются по сети.

Рис. иллюстрирует эффективное шифрование, реализованное путем сочетания симметричного и асимметричного методов.

На рис. показано расшифрование эффективно зашифрованного сообщения.

Отметим, что асимметричные методы позволили решить важную задачу совместной выработки секретных ключей (это существенно, если стороны не доверяют друг другу), обслуживающих сеанс взаимодействия, при изначальном отсутствии общих секретов. Для этого используется алгоритм Диффи-Хелмана.

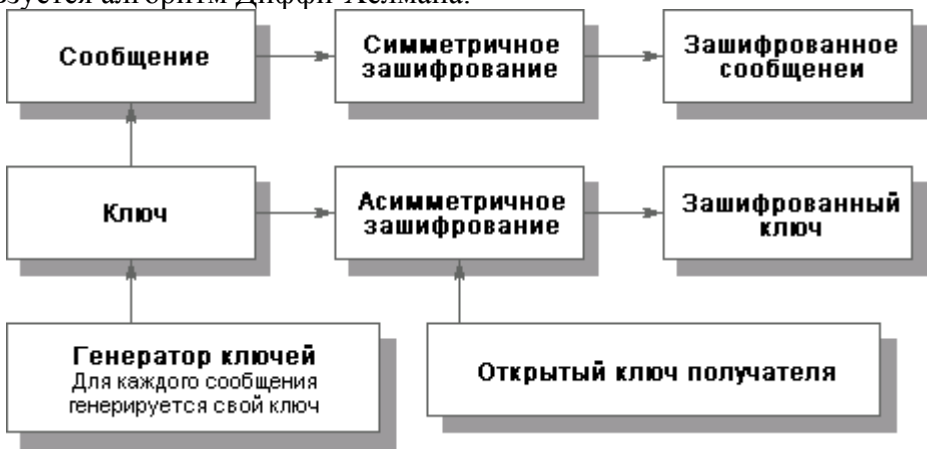


**Рис.** Эффективное шифрование сообщения.

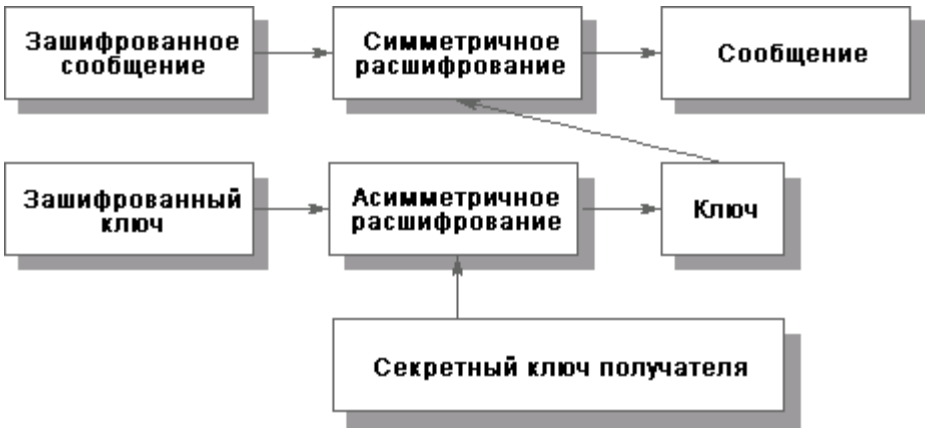


**Рис.** Расшифрование эффективно зашифрованного сообщения.

Определенное распространение получила разновидность симметричного шифрования, основанная на использовании *составных ключей*. Идея состоит в том, что секретный ключ делится на две части, хранящиеся отдельно. Каждая часть сама по себе не позволяет выполнить расшифрование. Если у правоохранительных органов появляются подозрения относительно лица, использующего некоторый ключ, они могут в установленном порядке получить половинки ключа и дальше действовать обычным для симметричного расшифрования образом.

Порядок работы с составными ключами – хороший пример следования принципу разделения обязанностей. Он позволяет сочетать права на разного рода тайны (персональную, коммерческую) с возможностью эффективно следить за нарушителями закона, хотя, конечно, здесь очень много тонкостей и технического, и юридического плана.

Многие криптографические алгоритмы в качестве одного из параметров требуют псевдослучайное значение, в случае предсказуемости которого в алгоритме появляется уязвимость (подобное уязвимое место было обнаружено в некоторых вариантах Web-навигаторов). Генерация *псевдослучайных последовательностей* – важный аспект криптографии, на котором мы, однако, останавливаться не будем.

**Контроль целостности**

Криптографические методы позволяют надежно контролировать целостность как отдельных порций данных, так и их наборов (таких как поток сообщений); определять подлинность источника данных; гарантировать невозможность отказаться от совершенных действий ("неотказуемость").

В основе криптографического контроля целостности лежат два понятия:

- хэш-функция;
- *электронная цифровая подпись (ЭЦП)*.

***Хэш-функция*** – это труднообратимое преобразование данных (*односторонняя функция*), реализуемое, как правило, средствами симметричного шифрования со связыванием блоков. Результат шифрования последнего блока (зависящий от всех предыдущих) и служит результатом хэш-функции.

Пусть имеются данные, целостность которых нужно проверить, хэш-функция и ранее вычисленный результат ее применения к исходным данным (так называемый *дайджест*). Обозначим хэш-функцию через h, исходные данные – через T, проверяемые данные – через T'. Контроль целостности данных сводится к проверке равенства h(T') = h(T). Если оно выполнено, считается, что T' = T. Совпадение дайджестов для различных данных называется коллизией. В принципе, коллизии, конечно, возможны, поскольку мощность множества дайджестов меньше, чем мощность множества хэшируемых данных, однако то, что h есть функция односторонняя, означает, что за приемлемое время специально организовать коллизию невозможно.

Рассмотрим теперь применение асимметричного шифрования для *выработки* и *проверки электронной цифровой подписи*. Пусть E(T) обозначает результат зашифрования текста T с помощью открытого ключа, а D(T) – результат расшифрования текста T (как правило,

шифрованного) с помощью секретного ключа. Чтобы асимметричный метод мог применяться для реализации ЭЦП, необходимо выполнение тождества

E(D(T)) = D(E(T)) = T

На рис. 11.5 показана процедура выработки электронной цифровой подписи, состоящая в шифровании преобразованием D дайджеста h(T).


**Рис.** Выработка электронной цифровой подписи.

Проверка ЭЦП может быть реализована так, как показано на рис. 11.6.

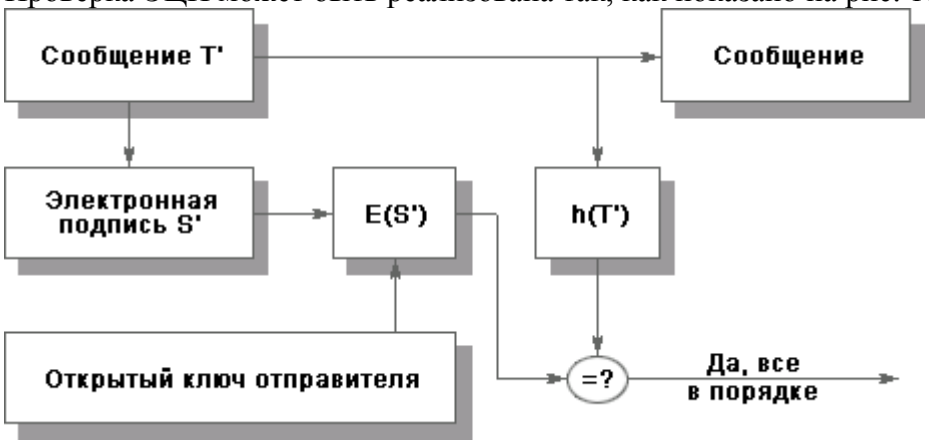


**Рис.** Проверка электронной цифровой подписи.

Из равенства

E(S') = h(T')

следует, что S' = D(h(T')) (для доказательства достаточно применить к обеим частям преобразование D и вычеркнуть в левой части тождественное преобразование D(E())). Таким образом, электронная цифровая подпись защищает целостность сообщения и удостоверяет личность отправителя, то есть защищает целостность источника данных и служит основой неотказуемости.

Для контроля целостности последовательности сообщений (то есть для защиты от кражи, дублирования и переупорядочения сообщений) применяют временные штампы и нумерацию элементов последовательности, при этом штампы и номера включают в подписываемый текст.

**Цифровые сертификаты**

При использовании асимметричных методов шифрования (и, в частности, электронной цифровой подписи) необходимо иметь гарантию подлинности пары (имя пользователя, открытый ключ пользователя). Для решения этой задачи в спецификациях X.509 вводятся понятия *цифрового сертификата* и *удостоверяющего центра*.

***Удостоверяющий центр*** – это компонент *глобальной службы каталогов*, отвечающий за управление криптографическими ключами пользователей. Открытые ключи и другая информация о пользователях хранится удостоверяющими центрами в виде цифровых сертификатов, имеющих следующую структуру:

- порядковый номер сертификата;
- идентификатор алгоритма электронной подписи;
- имя удостоверяющего центра;
- срок годности;
- имя владельца сертификата (имя пользователя, которому принадлежит сертификат);
- открытые ключи владельца сертификата (ключей может быть несколько);
- идентификаторы алгоритмов, ассоциированных с открытыми ключами владельца сертификата;
- электронная подпись, сгенерированная с использованием секретного ключа удостоверяющего центра (подписывается результат хэширования всей информации, хранящейся в сертификате).

Цифровые сертификаты обладают следующими свойствами:

- любой пользователь, знающий открытый ключ удостоверяющего центра, может узнать открытые ключи других клиентов центра и проверить целостность сертификата;
- никто, кроме удостоверяющего центра, не может модифицировать информацию о пользователе без нарушения целостности сертификата.

В спецификациях X.509 не описывается конкретная процедура генерации криптографических ключей и управления ими, однако даются некоторые общие рекомендации. В частности, оговаривается, что пары ключей могут порождаться любым из следующих способов:

- ключи может генерировать сам пользователь. В таком случае секретный ключ не попадает в руки третьих лиц, однако нужно решать задачу безопасной связи с удостоверяющим центром;
- ключи генерирует доверенное лицо. В таком случае приходится решать задачи безопасной доставки секретного ключа владельцу и предоставления доверенных данных для создания сертификата;
- ключи генерируются удостоверяющим центром. В таком случае остается только задача безопасной передачи ключей владельцу.

Цифровые сертификаты в формате X.509 версии 3 стали не только формальным, но и фактическим стандартом, поддерживаемым многочисленными удостоверяющими центрами

**Экранирование, анализ защищенности**

**Экранирование**

**Основные понятия**

Формальная постановка задачи *экранирования*, состоит в следующем. Пусть имеется два множества информационных систем. ***Экран*** – это средство *разграничения доступа* клиентов из одного множества к серверам из другого множества. *Экран* осуществляет свои функции, контролируя все информационные потоки между двумя множествами систем (рис. 12.1). Контроль потоков состоит в их *фильтрации*, возможно, с выполнением некоторых преобразований.

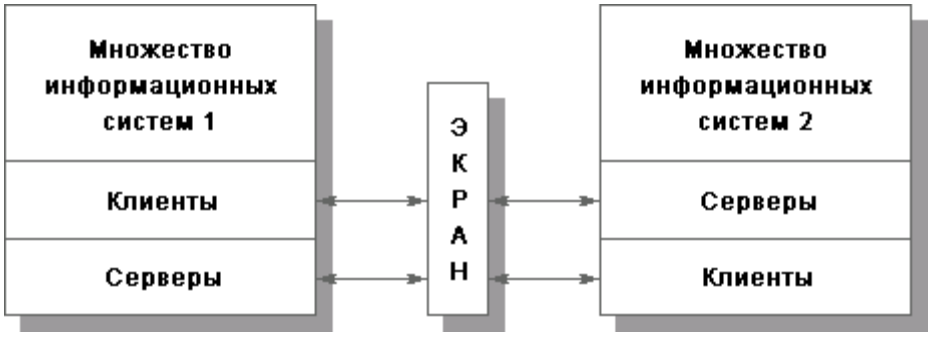


**Рис.** Экран как средство разграничения доступа.

На следующем уровне детализации *экран* (полупроницаемую мембрану) удобно представлять как последовательность *фильтров*. Каждый из *фильтров*, проанализировав данные, может задержать (не пропустить) их, а может и сразу "перебросить" за *экран*. Кроме того, допускается преобразование данных, передача порции данных на следующий *фильтр* для продолжения анализа или обработка данных от имени адресата и возврат результата отправителю (рис.).


**Рис.** Экран как последовательность фильтров.

Помимо функций *разграничения доступа*, *экраны* осуществляют *протоколирование* обмена информацией.

Обычно *экран* не является симметричным, для него определены понятия "внутри" и "снаружи". При этом задача *экранирования* формулируется как защита внутренней области от потенциально враждебной внешней. Так, *межсетевые экраны (МЭ)* чаще всего устанавливают для защиты корпоративной сети организации, имеющей выход в Internet

*Экранирование* помогает поддерживать *доступность* сервисов внутренней области, уменьшая или вообще ликвидируя нагрузку, вызванную внешней активностью. Уменьшается уязвимость внутренних сервисов безопасности, поскольку первоначально злоумышленник должен преодолеть *экран*, где защитные механизмы сконфигурированы особенно тщательно. Кроме того, экранирующая система, в отличие от универсальной, может быть устроена более простым и, следовательно, более безопасным образом.

*Экранирование* дает возможность контролировать также информационные потоки, направленные во внешнюю область, что способствует поддержанию режима *конфиденциальности* в ИС организации.

Подчеркнем, что *экранирование* может использоваться как сервис безопасности не только в сетевой, но и в любой другой среде, где происходит обмен сообщениями. Важнейший пример подобной среды – объектно-ориентированные программные системы, когда для активизации методов объектов выполняется (по крайней мере, в концептуальном плане) передача сообщений. Весьма вероятно, что в будущих объектно-ориентированных средах *экранирование* станет одним из важнейших инструментов *разграничения доступа* к объектам.

*Экранирование* может быть частичным, защищающим определенные информационные сервисы.

***Ограничивающий интерфейс*** также можно рассматривать как разновидность *экранирования*. На невидимый объект трудно нападать, особенно с помощью фиксированного набора средств. В этом смысле Web-интерфейс обладает естественной защитой, особенно в том случае, когда гипертекстовые документы формируются динамически. Каждый пользователь видит лишь то, что ему положено видеть. Можно провести аналогию между динамически формируемыми гипертекстовыми документами и представлениями в реляционных базах данных, с той существенной оговоркой, что в случае Web возможности существенно шире.

Экранирующая роль *Web-сервиса* наглядно проявляется и тогда, когда этот сервис осуществляет посреднические (точнее, интегрирующие) функции при доступе к другим ресурсам, например таблицам базы данных. Здесь не только контролируются потоки запросов, но и скрывается реальная организация данных.

**Архитектурные аспекты**

Бороться с угрозами, присущими сетевой среде, средствами универсальных операционных систем не представляется возможным. Универсальная ОС – это огромная программа, наверняка содержащая, помимо явных ошибок, некоторые особенности, которые могут быть использованы для нелегального получения привилегий. Современная технология программирования не позволяет сделать столь большие программы безопасными. Кроме того, администратор, имеющий дело со сложной системой, далеко не всегда в состоянии учесть все последствия производимых изменений. Наконец, в универсальной многопользовательской системе бреши в безопасности постоянно создаются самими пользователями (слабые и/или редко изменяемые пароли, неудачно установленные права доступа, оставленный без присмотра терминал и т.п.). Единственный перспективный путь связан с разработкой специализированных сервисов безопасности, которые в силу своей простоты допускают формальную или неформальную верификацию. *Межсетевой экран* как раз и является таким средством, допускающим дальнейшую декомпозицию, связанную с обслуживанием различных сетевых протоколов.

*Межсетевой экран* располагается между защищаемой (внутренней) сетью и внешней средой (внешними сетями или другими сегментами корпоративной сети). В первом случае говорят о внешнем *МЭ*, во втором – о внутреннем. В зависимости от точки зрения, внешний *межсетевой экран* можно считать первой или последней (но никак не единственной) линией обороны. Первой – если смотреть на мир глазами внешнего злоумышленника. Последней – если стремиться к защищенности всех компонентов корпоративной сети и пресечению неправомерных действий внутренних пользователей.

*Межсетевой экран* – идеальное место для встраивания средств активного аудита. С одной стороны, и на первом, и на последнем защитном рубеже выявление подозрительной активности по-своему важно. С другой стороны, *МЭ* способен реализовать сколь угодно мощную реакцию на подозрительную активность, вплоть до разрыва связи с внешней средой. Правда, нужно отдавать себе отчет в том, что соединение двух сервисов безопасности в принципе может создать брешь, способствующую атакам на *доступность*.

На *межсетевой экран* целесообразно возложить идентификацию/аутентификацию внешних пользователей, нуждающихся в доступе к корпоративным ресурсам (с поддержкой концепции единого входа в сеть).

В силу принципов **эшелонированности обороны** для защиты внешних подключений обычно используется двухкомпонентное *экранирование* (см. рис.). Первичная *фильтрация* (например, блокирование пакетов управляющего протокола SNMP, опасного атаками на доступность, или пакетов с определенными IP-адресами, включенными в "черный список") осуществляется **граничным маршрутизатором** (см. также следующий раздел), за которым располагается так называемая **демилитаризованная зона** (сеть с умеренным доверием безопасности, куда выносятся внешние информационные сервисы организации – Web, электронная почта и т.п.) и основной *МЭ*, защищающий внутреннюю часть корпоративной сети.

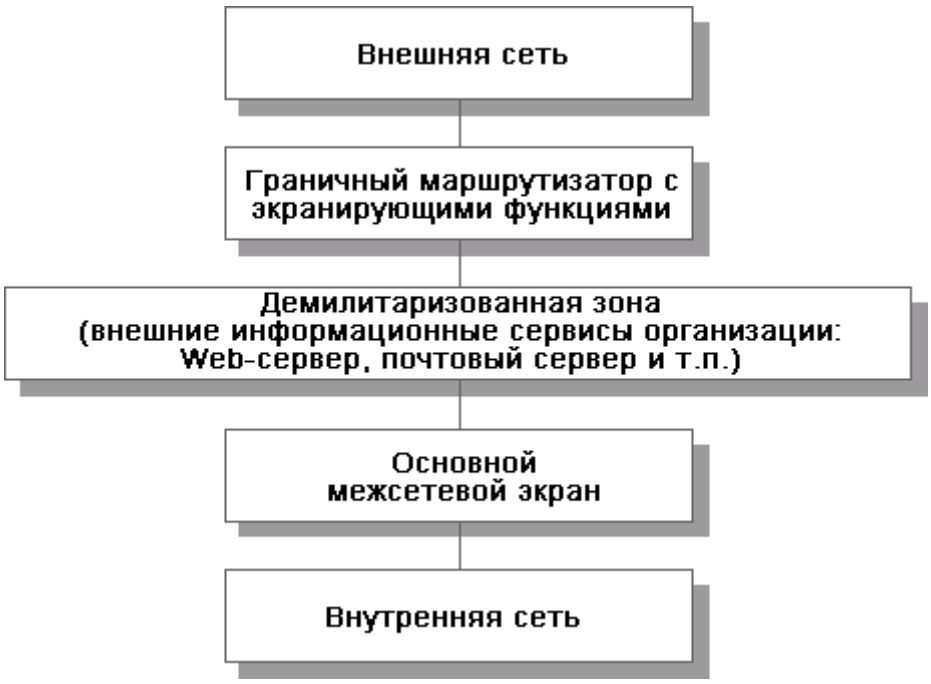


**Рис.** Двухкомпонентное экранирование с демилитаризованной зоной.

Теоретически *межсетевой экран* (особенно внутренний) **должен быть многопротокольным**, однако на практике доминирование семейства протоколов TCP/IP столь велико, что поддержка других протоколов представляется излишеством, вредным для безопасности (чем сложнее сервис, тем он более уязвим).

Вообще говоря, и внешний, и внутренний *межсетевой экран* может стать узким местом, поскольку объем сетевого трафика имеет тенденцию быстрого роста. Один из подходов к решению этой проблемы предполагает разбиение *МЭ* на несколько аппаратных частей и организацию специализированных **серверов-посредников**. Основной *межсетевой экран* может проводить грубую классификацию входящего трафика по видам и передоверять *фильтрацию* соответствующим посредникам (например, посреднику, анализирующему HTTP-трафик). Исходящий трафик сначала обрабатывается сервером-посредником, который может выполнять и функционально полезные действия, такие как кэширование страниц внешних *Web-серверов*, что снижает нагрузку на сеть вообще и основной *МЭ* в частности.

Ситуации, когда корпоративная сеть содержит лишь один внешний канал, являются скорее исключением, чем правилом. Напротив, типична ситуация, при которой корпоративная сеть состоит из нескольких территориально разнесенных сегментов, каждый из которых подключен к Internet. В этом случае каждое подключение должно защищаться своим *экраном*. Точнее говоря, можно считать, что корпоративный внешний *межсетевой экран* является составным, и требуется решать задачу согласованного администрирования (управления и аудита) всех компонентов.

Противоположностью составным корпоративным *МЭ* (или их компонентами) являются персональные *межсетевые экраны* и **персональные экранирующие устройства**. Первые являются программными продуктами, которые устанавливаются на персональные компьютеры и защищают только их. Вторые реализуются на отдельных устройствах и защищают небольшую локальную сеть, такую как сеть домашнего офиса.

**Классификация межсетевых экранов**

При рассмотрении любого вопроса, касающегося сетевых технологий, основой служит семиуровневая эталонная модель ISO/OSI. *Межсетевые экраны* также целесообразно классифицировать по уровню *фильтрации* – канальному, сетевому, транспортному или прикладному. Соответственно, можно говорить об **экранирующих концентраторах** (мостах, коммутаторах) (уровень 2), **маршрутизаторах** (уровень 3), о транспортном *экранировании*

(уровень 4) и о прикладных *экранах* (уровень 7). Существуют также комплексные *экраны*, анализирующие информацию на нескольких уровнях.

*Фильтрация* информационных потоков осуществляется *межсетевыми экранами* на основе **набора правил**, являющихся выражением сетевых аспектов политики безопасности организации. В этих правилах, помимо информации, содержащейся в фильтруемых потоках, могут фигурировать данные, полученные из окружения, например, текущее время, количество активных соединений, **порт**, через который поступил сетевой запрос, и т.д. Таким образом, в *межсетевых экранах* используется очень мощный логический подход к *разграничению доступа*.

Возможности *межсетевого экрана* непосредственно определяются тем, какая информация может использоваться в правилах *фильтрации* и какова может быть мощность наборов правил. Вообще говоря, чем выше уровень в модели ISO/OSI, на котором функционирует *МЭ*, тем более содержательная информация ему доступна и, следовательно, тем тоньше и надежнее он может быть сконфигурирован.

Экранирующие маршрутизаторы (и концентраторы) имеют дело с отдельными пакетами данных, поэтому иногда их называют **пакетными фильтрами**. Решения о том, пропустить или задержать данные, принимаются для каждого пакета независимо, на основании анализа адресов и других полей заголовков сетевого (канального) и, быть может, транспортного уровней. Еще один важный компонент анализируемой информации – порт, через который поступил пакет.

Экранирующие концентраторы являются средством не столько *разграничения доступа*, сколько оптимизации работы локальной сети за счет организации так называемых виртуальных локальных сетей. Последние можно считать важным результатом применения внутреннего межсетевого *экранирования*.

Современные маршрутизаторы позволяют связывать с каждым портом несколько десятков правил и **фильтровать пакеты как на входе, так и на выходе**. В принципе, в качестве пакетного фильтра может использоваться и универсальный компьютер, снабженный несколькими сетевыми картами.

Основные достоинства экранирующих маршрутизаторов – доступная цена (на границе сетей маршрутизатор нужен практически всегда, вопрос лишь в том, как задействовать его экранирующие возможности) и **прозрачность** для более высоких уровней модели OSI. Основной недостаток – ограниченность анализируемой информации и, как следствие, относительная слабость обеспечиваемой защиты.

Транспортное *экранирование* позволяет контролировать процесс установления виртуальных соединений и передачу информации по ним. С точки зрения реализации экранирующий транспорт представляет собой довольно простую, а значит, надежную программу.

По сравнению с пакетными фильтрами, транспортное *экранирование* обладает большей информацией, поэтому соответствующий *МЭ* может осуществлять более тонкий контроль за виртуальными соединениями (например, он способен отслеживать количество передаваемой информации и разрывать соединения после превышения определенного порога, препятствуя тем самым несанкционированному экспорту информации). Аналогично, возможно накопление более содержательной регистрационной информации. Главный недостаток – сужение области применения, поскольку вне контроля остаются датаграммные протоколы. Обычно транспортное *экранирование* применяют в сочетании с другими подходами, как важный дополнительный элемент.

*Межсетевой экран*, функционирующий на прикладном уровне, способен обеспечить наиболее надежную защиту. Как правило, подобный *МЭ* представляет собой универсальный компьютер, на котором функционируют **экранирующие агенты**, интерпретирующие протоколы прикладного уровня (HTTP, FTP, SMTP, telnet и т.д.) в той степени, которая необходима для обеспечения безопасности.

При использовании прикладных *МЭ*, помимо *фильтрации*, реализуется еще один важнейший аспект *экранирования*. Субъекты из внешней сети видят только шлюзовой компьютер; соответственно, им доступна только та информация о внутренней сети, которую он считает нужным экспортировать. Прикладной *МЭ* на самом деле экранирует, то есть заслоняет, внутреннюю сеть от внешнего мира. В то же время, субъектам внутренней сети кажется, что они напрямую общаются с объектами внешнего мира. Недостаток прикладных *МЭ* – отсутствие полной прозрачности, требующее специальных действий для поддержки каждого прикладного протокола.

Если организация располагает исходными текстами прикладного *МЭ* и в состоянии эти тексты модифицировать, перед ней открываются чрезвычайно широкие возможности по настройке *экрана* с учетом собственных нужд. Дело в том, что при разработке систем **клиент/сервер в многозвенной архитектуре** появляются специфические прикладные протоколы, нуждающиеся в защите не меньше стандартных. Подход, основанный на использовании экранирующих агентов, позволяет построить такую защиту, не снижая безопасности и эффективности других приложений и не усложняя структуру связей в *межсетевом экране*.

Комплексные *межсетевые экраны*, охватывающие уровни от сетевого до прикладного, соединяют в себе лучшие свойства "одноуровневых" *МЭ* разных видов. Защитные функции выполняются комплексными *МЭ* прозрачным для приложений образом, не требуя внесения каких-либо изменений ни в существующее программное обеспечение, ни в действия, ставшие для пользователей привычными.

Комплексность *МЭ* может достигаться разными способами: "снизу вверх", от сетевого уровня через накопление контекста к прикладному уровню, или "сверху вниз", посредством дополнения прикладного *МЭ* механизмами транспортного и сетевого уровней.

Помимо выразительных возможностей и допустимого количества правил, качество *межсетевого экрана* определяется еще двумя очень важными характеристиками – **простотой использования** и **собственной защищенностью**. В плане простоты использования первостепенное значение имеют наглядный интерфейс при определении правил *фильтрации* и возможность **централизованного администрирования** составных конфигураций. В свою очередь, в последнем аспекте хотелось бы выделить средства централизованной загрузки правил *фильтрации* и **проверки набора правил на непротиворечивость**. Важен и централизованный сбор и анализ регистрационной информации, а также получение сигналов о попытках выполнения действий, запрещенных политикой безопасности.

Собственная защищенность *межсетевого экрана* обеспечивается теми же средствами, что и защищенность универсальных систем. Имеется в виду физическая защита, идентификация и аутентификация, *разграничение доступа*, контроль целостности, протоколирование и аудит. При выполнении централизованного администрирования следует также позаботиться о защите информации от пассивного и активного прослушивания сети, то есть обеспечить ее (информации) целостность и *конфиденциальность*. Крайне важно оперативное наложение заплат, ликвидирующих выявленные уязвимые места *МЭ*.

Хотелось бы подчеркнуть, что природа *экранирования* как сервиса безопасности очень глубока. Помимо блокирования потоков данных, нарушающих политику безопасности, *межсетевой экран* может скрывать информацию о защищаемой сети, тем самым затрудняя действия потенциальных злоумышленников. Мощным методом сокрытия информации является **трансляция** "внутренних" сетевых **адресов**, которая попутно решает проблему расширения адресного пространства, выделенного организации.

Отметим также следующие дополнительные возможности *межсетевых экранов*:

- контроль информационного наполнения (антивирусный **контроль "на лету"**, верификация Java-апплетов, выявление ключевых слов в электронных сообщениях и т.п.);
- выполнение функций **ПО промежуточного слоя**.

Особенно важным представляется последний из перечисленных аспектов. ПО промежуточного слоя, как и традиционные *межсетевые экраны* прикладного уровня, скрывает информацию о предоставляемых услугах. За счет этого оно может выполнять такие функции, как **маршрутизация запросов** и **балансировка нагрузки**. Представляется вполне естественным, чтобы эти возможности были реализованы в рамках *межсетевого экрана*. Это существенно упрощает действия по обеспечению высокой *доступности* экспортируемых сервисов и позволяет осуществлять переключение на резервные мощности прозрачным для внешних пользователей образом. В результате к услугам, традиционно предоставляемым *межсетевыми экранами*, добавляется поддержка высокой *доступности* сетевых сервисов.

**Анализ защищенности**

Сервис **анализа защищенности** предназначен для выявления уязвимых мест с целью их оперативной ликвидации. Сам по себе этот сервис ни от чего не защищает, но помогает обнаружить (и устранить) пробелы в защите раньше, чем их сможет использовать злоумышленник. В первую очередь, имеются в виду не архитектурные (их ликвидировать сложно), а "оперативные" бреши, появившиеся в результате ошибок администрирования или из-за невнимания к обновлению версий программного обеспечения.

Системы анализа защищенности (называемые также **сканерами защищенности**), как и рассмотренные выше средства активного аудита, основаны на накоплении и использовании знаний. В данном случае имеются в виду знания о пробелах в защите: о том, как их искать, насколько они серьезны и как их устранять.

Соответственно, ядром таких систем является **база уязвимых мест**, которая определяет доступный диапазон возможностей и требует практически постоянной актуализации.

Наиболее эффективными являются **сетевые сканеры** (очевидно, в силу доминирования семейства протоколов TCP/IP), а также антивирусные средства. **Антивирусную защиту** мы причисляем к средствам анализа защищенности, не считая ее отдельным сервисом безопасности.

Сканеры могут выявлять уязвимые места как путем пассивного анализа, то есть изучения конфигурационных файлов, задействованных портов и т.п., так и путем имитации действий атакующего. Некоторые найденные уязвимые места могут устраняться автоматически (например, лечение зараженных файлов), о других сообщается администратору.

Системы анализа защищенности снабжены традиционным "технологическим сахаром": **автообнаружением** компонентов анализируемой ИС и графическим интерфейсом (помогающим, в частности, эффективно работать с протоколом сканирования).

**Обеспечение высокой доступности**

**Доступность**

**Основные понятия**

Информационная система предоставляет своим пользователям определенный набор услуг (сервисов). Говорят, что обеспечен нужный уровень доступности этих сервисов, если следующие показатели находятся в заданных пределах:

- **Эффективность услуг**. Эффективность услуги определяется в терминах максимального времени обслуживания запроса, количества поддерживаемых пользователей и т.п. Требуется, чтобы эффективность не опускалась ниже заранее установленного порога.
- *Время недоступности*. Если эффективность информационной услуги не удовлетворяет наложенным ограничениям, услуга считается недоступной. Требуется, чтобы максимальная продолжительность периода недоступности и суммарное *время недоступности* за некоторой период (месяц, год) не превышали заранее заданных пределов.

В сущности, требуется, чтобы информационная система почти всегда работала с нужной эффективностью. Для некоторых критически важных систем (например, систем управления)

*время недоступности* должно быть нулевым, без всяких "почти". В таком случае говорят о вероятности возникновения ситуации недоступности и требуют, чтобы эта вероятность не превышала заданной величины. Для решения данной задачи создавались и создаются специальные ***отказоустойчивые системы***, стоимость которых, как правило, весьма высока.

К подавляющему большинству коммерческих систем предъявляются менее жесткие требования, однако современная деловая жизнь и здесь накладывает достаточно суровые ограничения, когда число обслуживаемых пользователей может измеряться тысячами, время ответа не должно превышать нескольких секунд, а *время недоступности* – нескольких часов в год.

Задачу обеспечения *высокой доступности* необходимо решать для современных конфигураций, построенных в технологии клиент/сервер. Это означает, что в защите нуждается вся цепочка – от пользователей (возможно, удаленных) до критически важных серверов (в том числе серверов безопасности).

Основные угрозы доступности были рассмотрены нами ранее.

В соответствии с ГОСТ 27.002, под ***отказом*** понимается событие, которое заключается в нарушении работоспособности изделия. В контексте данной работы изделие – это информационная система или ее компонент.

В простейшем случае можно считать, что *отказы* любого компонента составного изделия ведут к общему *отказу*, а распределение *отказов* во времени представляет собой простой пуассоновский поток событий. В таком случае вводят понятие ***интенсивности отказов*** и ***среднего времени наработки на отказ***, которые связаны между собой соотношением

$$T_i = \frac{1}{\lambda_i}$$

где **i** – номер компонента,

$\lambda_i$ – *интенсивность отказов*,

$T_i$ – *среднее время наработки на отказ*.

*Интенсивности отказов* независимых компонентов складываются:

$$\lambda = \lambda_1 + \ldots + \lambda_n$$

а *среднее время наработки на отказ* для составного изделия задается соотношением

$$T = \frac{1}{\lambda}$$

Уже эти выкладки показывают, что если существует компонент, *интенсивность отказов* которого много больше, чем у остальных, то именно он определяет *среднее время наработки на отказ* всей информационной системы. Это является теоретическим обоснованием принципа первоочередного укрепления самого *слабого звена*.

Пуассоновская модель позволяет обосновать еще одно очень важное положение, состоящее в том, что эмпирический подход к построению систем *высокой доступности* не может быть реализован за приемлемое время. При традиционном цикле тестирования/отладки программной системы по оптимистическим оценкам каждое исправление ошибки приводит к экспоненциальному убыванию (примерно на половину десятичного порядка) *интенсивности отказов*. Отсюда следует, что для того, чтобы на опыте убедиться в достижении необходимого уровня доступности, независимо от применяемой технологии тестирования и отладки, придется потратить время, практически равное *среднему времени наработки на отказ*. Например, для достижения *среднего времени наработки на отказ* $10^5$ часов потребуется более $10^{4,5}$ часов, что составляет более трех лет. Значит, нужны иные методы построения систем *высокой доступности*, методы, эффективность которых доказана аналитически или практически за более чем пятьдесят лет развития вычислительной техники и программирования.

Пуассоновская модель применима в тех случаях, когда информационная система содержит одиночные точки *отказа*, то есть компоненты, выход которых из строя ведет к *отказу* всей системы. Для исследования систем с резервированием применяется иной формализм.

В соответствии с постановкой задачи будем считать, что существует количественная мера эффективности предоставляемых изделием информационных услуг. В таком случае вводятся понятия ***показателей эффективности*** отдельных элементов и эффективности функционирования всей сложной системы.

В качестве меры доступности можно принять вероятность приемлемости эффективности услуг, предоставляемых информационной системой, на всем протяжении рассматриваемого отрезка времени. Чем большим запасом эффективности располагает система, тем выше ее доступность.

При наличии *избыточности* в конфигурации системы вероятность того, что в рассматриваемый промежуток времени *эффективность информационных сервисов* не опустится ниже допустимого предела, зависит не только от вероятности *отказа* компонентов, но и от времени, в течение которого они остаются неработоспособными, поскольку при этом суммарная эффективность падает, и каждый следующий *отказ* может стать фатальным. Чтобы максимально увеличить доступность системы, необходимо минимизировать время неработоспособности каждого компонента. Кроме того, следует учитывать, что, вообще говоря, ремонтные работы могут потребовать понижения эффективности или даже временного отключения работоспособных компонентов; такого рода влияние также необходимо минимизировать.

Несколько терминологических замечаний. Обычно в литературе по теории надежности вместо доступности говорят о ***готовности*** (в том числе о высокой *готовности*). Мы предпочли термин "доступность", чтобы подчеркнуть, что *информационный сервис* должен быть не просто "готов" сам по себе, но доступен для своих пользователей в условиях, когда ситуации недоступности могут вызываться причинами, на первый взгляд не имеющими прямого отношения к сервису (пример – отсутствие консультационного обслуживания).

Далее, вместо *времени недоступности* обычно говорят о ***коэффициенте готовности***. Нам хотелось обратить внимание на два показателя – длительность однократного простоя и суммарную продолжительность простоев, поэтому мы предпочли термин "*время недоступности*" как более емкий.

**Основы мер обеспечения высокой доступности**

Основой мер повышения доступности является применение структурированного подхода, нашедшего воплощение в объектно-ориентированной методологии. *Структуризация* необходима по отношению ко всем аспектам и составным частям информационной системы – от архитектуры до административных баз данных, на всех этапах ее жизненного цикла – от инициации до выведения из эксплуатации. *Структуризация*, важная сама по себе, является одновременно необходимым условием практической реализуемости прочих мер повышения доступности.

При разработке мер обеспечения *высокой доступности информационных сервисов* рекомендуется руководствоваться следующими архитектурными принципами, рассматривавшимися ранее:

- апробированность всех процессов и составных частей информационной системы;
- унификация процессов и составных частей;
- управляемость процессов, контроль состояния частей;
- автоматизация процессов;
- модульность архитектуры;
- ориентация на простоту решений.

Доступность системы в общем случае достигается за счет применения трех групп мер, направленных на повышение:

- ***безотказности*** (под этим понимается минимизация вероятности возникновения какого-либо *отказа*; это элемент *пассивной безопасности*, который дальше рассматриваться не будет);

- ***отказоустойчивости*** (способности к *нейтрализации отказов*, *"живучести"*, то есть способности сохранять требуемую эффективность, несмотря на *отказы* отдельных компонентов);
- ***обслуживаемости*** (под *обслуживаемостью* понимается минимизация времени простоя отказавших компонентов, а также отрицательного влияния ремонтных работ на *эффективность информационных сервисов*, то есть быстрое и *безопасное восстановление* после *отказов*).

Главное при разработке и реализации мер обеспечения *высокой доступности – полнота* и *систематичность*. В этой связи представляется целесообразным составить (и поддерживать в актуальном состоянии) ***карту информационной системы*** организации (на что мы уже обращали внимание), в которой фигурировали бы все объекты ИС, их состояние, связи между ними, процессы, ассоциируемые с объектами и связями. С помощью подобной *карты* удобно формулировать намечаемые меры, контролировать их исполнение, анализировать состояние ИС.

**Отказоустойчивость и зона риска**

Информационную систему можно представить в виде графа сервисов, ребра в котором соответствуют отношению "сервис А непосредственно использует сервис В".

Пусть в результате осуществления некоторой атаки (источником которой может быть как человек, так и явление природы) выводится из строя подмножество сервисов S1 (то есть эти сервисы в результате нанесенных повреждений становятся неработоспособными). Назовем S1 **зоной поражения**.

В **зону риска** S мы будем включать все сервисы, эффективность которых при осуществлении атаки падает ниже допустимого предела. Очевидно, S1 – подмножество S. S строго включает S1, когда имеются сервисы, непосредственно не затронутые атакой, но критически зависящие от пораженных, то есть неспособные переключиться на использование эквивалентных услуг либо в силу отсутствия таковых, либо в силу невозможности доступа к ним. Например, зона поражения может сводиться к одному порту концентратора, обслуживающему критичный сервер, а зона риска охватывает все рабочие места пользователей сервера.

Чтобы система не содержала **одиночных точек** *отказа*, то есть оставалась "живучей" при реализации любой из рассматриваемых угроз, ни одна зона риска не должна включать в себя предоставляемые услуги. *Нейтрализацию отказов* нужно выполнять внутри системы, незаметно для пользователей, за счет размещения достаточного количества избыточных ресурсов.

С другой стороны, естественно соизмерять усилия по обеспечению "*живучести*" с рассматриваемыми угрозами. Когда рассматривается набор угроз, соответствующие им зоны поражения могут оказаться вложенными, так что "*живучесть*" по отношению к более серьезной угрозе автоматически влечет за собой и "*живучесть*" в более легких случаях. Следует учитывать, однако, что обычно стоимость переключения на резервные ресурсы растет вместе с увеличением объема этих ресурсов. Значит, для наиболее вероятных угроз целесообразно минимизировать зону риска, даже если предусмотрена нейтрализация объемлющей угрозы. Нет смысла переключаться на резервный вычислительный центр только потому, что у одного из серверов вышел из строя блок питания.

Зону риска можно трактовать не только как совокупность ресурсов, но и как часть пространства, затрагиваемую при реализации угрозы. В таком случае, как правило, чем больше расстояние дублирующего ресурса от границ зоны риска, тем выше стоимость его поддержания, поскольку увеличивается протяженность линий связи, время переброски персонала и т.п.

Введем еще одно понятие. Назовем **зоной нейтрализации** угрозы совокупность ресурсов, вовлеченных в *нейтрализацию отказа*, возникшего вследствие реализации угрозы. Имеются в виду ресурсы, режим работы которых в случае *отказа* изменяется. Очевидно, зона

риска является подмножеством зоны нейтрализации. Чем меньше разность между ними, тем экономичнее данный механизм нейтрализации.

Все, что находится вне зоны нейтрализации, *отказа* "не чувствует" и может трактовать внутренность этой зоны как безотказную. Таким образом, в иерархически организованной системе грань между "*живучестью*" и *обслуживаемостью*, с одной стороны, и *безотказностью*, с другой стороны, относительна. Целесообразно конструировать целостную информационную систему из компонентов, которые на верхнем уровне можно считать безотказными, а вопросы "*живучести*" и *обслуживаемости* решать в пределах каждого компонента.

**Обеспечение отказоустойчивости**

Основным средством повышения "*живучести*" является внесение *избыточности* в конфигурацию аппаратных и программных средств, поддерживающей инфраструктуры и персонала, **резервирование** технических средств и **тиражирование** информационных ресурсов (программ и данных).

**Меры** по обеспечению *отказоустойчивости* можно разделить на **локальные** и **распределенные**. Локальные меры направлены на достижение "*живучести*" отдельных компьютерных систем или их аппаратных и программных компонентов (в первую очередь с целью нейтрализации внутренних *отказов* ИС). Типичные примеры подобных мер – использование кластерных конфигураций в качестве платформы критичных серверов или "горячее" резервирование активного сетевого оборудования с автоматическим переключением на резерв.

Если в число рассматриваемых рисков входят серьезные аварии поддерживающей инфраструктуры, приводящие к выходу из строя производственной площадки организации, следует предусмотреть распределенные меры обеспечения *живучести*, такие как создание или аренда резервного вычислительного центра. При этом, помимо дублирования и/или тиражирования ресурсов, необходимо предусмотреть средства автоматического или быстрого ручного переконфигурирования компонентов ИС, чтобы обеспечить переключение с основной площадки на резервную.

Аппаратура – относительно статичная составляющая, однако было бы ошибкой полностью отказывать ей в динамичности. В большинстве организаций информационные системы находятся в постоянном развитии, поэтому на протяжении всего жизненного цикла ИС следует соотносить все изменения с необходимостью обеспечения "*живучести*", не забывать "тиражировать" новые и модифицированные компоненты.

Программы и данные более динамичны, чем аппаратура, и резервироваться они могут постоянно, при каждом изменении, после завершения некоторой логически замкнутой группы изменений или по истечении определенного времени.

Резервирование программ и данных может выполняться многими способами – за счет зеркалирования дисков, резервного копирования и восстановления, репликации баз данных и т.п. Будем использовать для всех перечисленных способов термин "тиражирование".

Выделим следующие классы тиражирования:

- **Симметричное/асимметричное**. Тиражирование называется симметричным, если все серверы, предоставляющие данный сервис, могут изменять принадлежащую им информацию и передавать изменения другим серверам. В противном случае тиражирование называется асимметричным.
- **Синхронное/асинхронное**. Тиражирование называется синхронным, если изменение передается всем экземплярам сервиса в рамках одной распределенной транзакции. В противном случае тиражирование называется асинхронным.
- **Осуществляемое средствами сервиса**, хранящего информацию/**внешними средствами**.

Рассмотрим, какие способы тиражирования предпочтительнее.

Безусловно, следует предпочесть стандартные средства тиражирования, встроенные в сервис.

Асимметричное тиражирование теоретически проще симметричного, поэтому целесообразно выбрать асимметрию.

Труднее всего выбрать между синхронным и асинхронным тиражированием. Синхронное идейно проще, но его реализация может быть тяжеловесной и сложной, хотя это внутренняя сложность сервиса, невидимая для приложений. Асинхронное тиражирование устойчивее к *отказам* в сети, оно меньше влияет на работу основного сервиса.

Чем надежнее связь между серверами, вовлеченными в процесс тиражирования, чем меньше время, отводимое на переключение с основного сервера на резервный, чем жестче требования к актуальности информации, тем более предпочтительным оказывается синхронное тиражирование.

С другой стороны, недостатки асинхронного тиражирования могут компенсироваться процедурными и программными мерами, направленными на контроль целостности информации в распределенной ИС. Сервисы, входящие в состав ИС, в состоянии обеспечить ведение и хранение журналов транзакций, с помощью которых можно выявлять операции, утерянные при переключении на резервный сервер. Даже в условиях неустойчивой связи с удаленными филиалами организации подобная проверка в фоновом режиме займет не более нескольких часов, поэтому асинхронное тиражирование может использоваться практически в любой ИС.

Асинхронное тиражирование может производиться на сервер, работающий в режиме "горячего" резерва, возможно, даже обслуживающего часть пользовательских запросов, или на сервер, работающий в режиме "теплого" резерва, когда изменения периодически "накатываются", но сам резервный сервер запросов не обслуживает.

Достоинство "теплого" резервирования в том, что его можно реализовать, оказывая меньшее влияние на основной сервер. Это влияние вообще может быть сведено к нулю, если асинхронное тиражирование осуществляется путем передачи инкрементальных копий с основного сервера (резервное копирование необходимо выполнять в любом случае).

Основной недостаток "теплого" резерва состоит в длительном времени включения, что может быть неприемлемо для "тяжелых" серверов, таких как кластерная конфигурация сервера СУБД. Здесь необходимо проводить измерения в условиях, близких к реальным.

Второй недостаток "теплого" резерва вытекает из опасности малых изменений. Может оказаться, что в самый нужный момент срочный перевод резерва в штатный режим невозможен.

Учитывая приведенные соображения, следует в первую очередь рассматривать возможность **"горячего" резервирования**, либо тщательно контролировать использование **"теплого" резерва** и регулярно (не реже одного раза в неделю) проводить пробные переключения резерва в "горячий" режим.

**Программное обеспечение промежуточного слоя**

С помощью **программного обеспечения промежуточного слоя (ПО ПС)** можно для произвольных прикладных сервисов добиться высокой "*живучести*" с полностью прозрачным для пользователей переключением на резервные мощности.

Перечислим основные достоинства ПО ПС, существенные для обеспечения *высокой доступности*.

- ПО ПС уменьшает сложность создания распределенных систем. Подобное ПО берет на себя часть функций, которые в локальном случае выполняют операционные системы;
- ПО ПС берет на себя **маршрутизацию запросов**, позволяя тем самым обеспечить "*живучесть*" прозрачным для пользователей образом;
- ПО ПС осуществляет **балансировку загрузки** вычислительных мощностей, что также способствует повышению доступности данных;

- ПО ПС в состоянии осуществлять **тиражирование** любой информации, а не только содержимого баз данных. Следовательно, любое приложение можно сделать устойчивым к *отказам* серверов;
- ПО ПС в состоянии **отслеживать состояние приложений** и при необходимости тиражировать и перезапускать программы, что гарантирует "*живучесть*" программных систем;
- ПО ПС дает возможность прозрачным для пользователей образом выполнять **переконфигурирование** (и, в частности, **наращивание**) серверных компонентов, что позволяет масштабировать систему, сохраняя инвестиции в прикладные системы. Стабильность прикладных систем – важный фактор повышения доступности данных.

Ранее мы упоминали о достоинствах использования ПО ПС в рамках межсетевых экранов, которые в таком случае становятся элементом обеспечения *отказоустойчивости* предоставляемых *информационных сервисов*.

**Обеспечение обслуживаемости**

Меры по обеспечению *обслуживаемости* направлены на снижение сроков диагностирования и устранения *отказов* и их последствий.

Для обеспечения *обслуживаемости* рекомендуется соблюдать следующие архитектурные принципы:

- ориентация на построение информационной системы из унифицированных компонентов с целью упрощения замены отказавших частей;
- ориентация на решения **модульной** структуры с возможностью **автоматического обнаружения** *отказов*, **динамического переконфигурирования** аппаратных и программных средств и **замены** *отказавших* **компонентов в "горячем" режиме**.

Динамическое переконфигурирование преследует две основные цели:

- **изоляция отказавших компонентов**;
- **сохранение работоспособности сервисов**.

Изолированные компоненты образуют зону поражения реализованной угрозы. Чем меньше соответствующая зона риска, тем выше *обслуживаемость* сервисов. Так, при *отказах* блоков питания, вентиляторов и/или дисков в современных серверах зона риска ограничивается отказавшим компонентом; при *отказах* процессорных модулей весь сервер может потребовать перезагрузки (что способно вызвать дальнейшее расширение зоны риска). Очевидно, в идеальном случае зоны поражения и риска совпадают, и современные серверы и активное сетевое оборудование, а также программное обеспечение ведущих производителей весьма близки к этому идеалу.

Возможность программирования реакции на *отказ* также повышает *обслуживаемость* систем. Каждая организация может выбрать свою стратегию реагирования на *отказы* тех или иных аппаратных и программных компонентов и автоматизировать эту реакцию. Так, в простейшем случае возможна отправка сообщения системному администратору, чтобы ускорить начало ремонтных работ; в более сложном случае может быть реализована процедура "мягкого" выключения (переключения) сервиса, чтобы упростить обслуживание.

Возможность удаленного выполнения административных действий – важное направление повышения *обслуживаемости*, поскольку при этом ускоряется начало восстановительных мероприятий, а в идеале все работы (обычно связанные с обслуживанием программных компонентов) выполняются в удаленном режиме, без перемещения квалифицированного персонала, то есть с высоким качеством и в кратчайшие сроки. Для современных систем возможность удаленного администрирования – стандартное свойство, но важно позаботиться о его практической реализуемости в условиях разнородности конфигураций (в первую очередь клиентских). Централизованное распространение и конфигурирование программного обеспечения, управление компонентами информационной

системы и диагностирование – надежный фундамент технических мер повышения *обслуживаемости*.

Существенный аспект повышения *обслуживаемости* – организация консультационной службы для пользователей (**обслуживаемость пользователей**), внедрение программных систем для работы этой службы, обеспечение достаточной пропускной способности каналов связи с пользователями, в том числе в режиме пиковых нагрузок.

**Туннелирование и управление**

**Туннелирование**

***Туннелирование*** следует рассматривать как самостоятельный сервис безопасности. Его суть состоит в том, чтобы "упаковать" передаваемую порцию данных, вместе со служебными полями, в новый "конверт". В качестве синонимов термина "*туннелирование*" могут использоваться "***конвертование***" и "***обертывание***".

*Туннелирование* может применяться для нескольких целей:

- передачи через сеть пакетов, принадлежащих протоколу, который в данной сети не поддерживается (например, передача пакетов IPv6 через старые сети, поддерживающие только IPv4);
- обеспечения слабой формы *конфиденциальности* (в первую очередь *конфиденциальности трафика*) за счет сокрытия истинных адресов и другой служебной информации;
- обеспечения *конфиденциальности* и целостности передаваемых данных при использовании вместе с криптографическими сервисами.

*Туннелирование* может применяться как на сетевом, так и на прикладном уровнях. Например, стандартизовано *туннелирование* для IP и двойное *конвертование* для почты X.400.

На рис. показан пример обертывания пакетов IPv6 в формат IPv4.

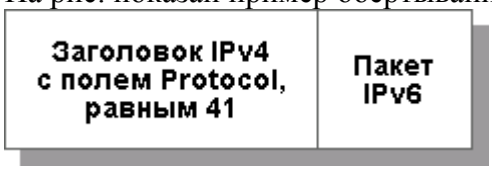


**Рис.** Обертывание пакетов IPv6 в формат IPv4 с целью их туннелирования через сети IPv4.

Комбинация *туннелирования* и шифрования (наряду с необходимой криптографической инфраструктурой) на выделенных шлюзах и экранирования на маршрутизаторах поставщиков сетевых услуг (для разделения пространств "своих" и "чужих" сетевых адресов в духе виртуальных локальных сетей) позволяет реализовать такое важное в современных условиях защитное средство, как ***виртуальные частные сети***. Подобные сети, наложенные обычно поверх Internet, существенно дешевле и гораздо безопаснее, чем собственные сети организации, построенные на выделенных каналах. Коммуникации на всем их протяжении физически защитить невозможно, поэтому лучше изначально исходить из предположения об их уязвимости и соответственно обеспечивать защиту. Современные протоколы, направленные на поддержку классов обслуживания, помогут гарантировать для *виртуальных частных сетей* заданную пропускную способность, величину задержек и т.п., ликвидируя тем самым единственное на сегодня реальное преимущество сетей собственных.

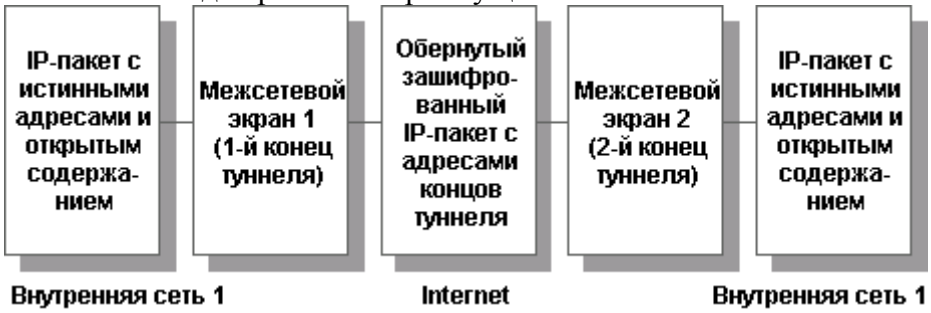


**Рис.** Межсетевые экраны как точки реализации сервиса виртуальных частных сетей.

Концами *туннелей*, реализующих *виртуальные частные сети*, целесообразно сделать *межсетевые экраны*, обслуживающие подключение организаций к внешним сетям (см. рис.). В таком случае *туннелирование* и шифрование станут дополнительными преобразованиями, выполняемыми в процессе фильтрации сетевого трафика наряду с трансляцией адресов.

Концами *туннелей*, помимо корпоративных *межсетевых экранов*, могут быть мобильные компьютеры сотрудников (точнее, их персональные МЭ).

**Управление**

**Основные понятия**

Управление можно отнести к числу инфраструктурных сервисов, обеспечивающих нормальную работу компонентов и средств безопасности. Сложность современных систем такова, что без правильно организованного управления они постепенно деградируют как в плане эффективности, так и в плане защищенности.

Возможен и другой взгляд на управление – как на интегрирующую оболочку информационных сервисов и сервисов безопасности (в том числе средств обеспечения высокой доступности), обеспечивающую их нормальное, согласованное функционирование под контролем администратора ИС.

Согласно стандарту X.700, управление подразделяется на:

- **мониторинг** компонентов;
- **контроль** (то есть выдачу и реализацию управляющих воздействий);
- **координацию** работы компонентов системы.

Системы управления должны:

- позволять администраторам планировать, организовывать, контролировать и учитывать использование информационных сервисов;
- давать возможность отвечать на изменение требований;
- обеспечивать предсказуемое поведение информационных сервисов;
- обеспечивать защиту информации.

Иными словами, управление должно обладать достаточно богатой функциональностью, быть результативным, гибким и информационно безопасным.

В X.700 выделяется пять функциональных областей управления:

- **управление конфигурацией** (установка параметров для нормального функционирования, запуск и остановка компонентов, сбор информации о текущем состоянии системы, прием извещений о существенных изменениях в условиях функционирования, изменение конфигурации системы);
- **управление отказами** (выявление отказов, их изоляция и восстановление работоспособности системы);
- **управление производительностью** (сбор и анализ статистической информации, определение производительности системы в штатных и нештатных условиях, изменение режима работы системы);
- **управление безопасностью** (реализация политики безопасности путем создания, удаления и изменения сервисов и механизмов безопасности, распространения соответствующей информации и реагирования на инциденты);
- **управление учетной информацией** (т.е. взимание платы за пользование ресурсами).

В стандартах семейства X.700 описывается модель управления, способная обеспечить достижение поставленных целей. Вводится понятие управляемого объекта как совокупности характеристик компонента системы, важных с точки зрения управления. К таким характеристикам относятся:

- **атрибуты объекта**;
- **допустимые операции**;
- **извещения**, которые объект может генерировать;
- связи с другими управляемыми объектами.

Согласно рекомендациям X.701, системы управления распределенными ИС строятся в архитектуре **менеджер/агент**. Агент (как программная модель управляемого объекта) выполняет управляющие действия и порождает (при возникновении определенных событий) извещения от его имени. В свою очередь, менеджер выдает агентам команды на управляющие воздействия и получает извещения.

Иерархия взаимодействующих менеджеров и агентов может иметь несколько уровней. При этом элементы промежуточных уровней играют двоякую роль: по отношению к вышестоящим элементам они являются агентами, а к нижестоящим – менеджерами. **Многоуровневая архитектура** менеджер/агент – ключ к распределенному, масштабируемому управлению большими системами.

Логически связанной с многоуровневой архитектурой является концепция **доверенного** (или **делегированного**) **управления**. При доверенном управлении менеджер промежуточного уровня может управлять объектами, использующими собственные протоколы, в то время как "наверху" опираются исключительно на стандартные средства.

Обязательным элементом при любом числе архитектурных уровней является **управляющая консоль**.

С точки зрения изучения возможностей систем управления следует учитывать разделение, введенное в X.701. Управление подразделяется на следующие аспекты:

- информационный (атрибуты, операции и извещения управляемых объектов);
- функциональный (управляющие действия и необходимая для них информация);
- коммуникационный (обмен управляющей информацией);
- организационный (разбиение на области управления).

Ключевую роль играет **модель управляющей информации**. Она описывается рекомендациями X.720. Модель является объектно-ориентированной с поддержкой инкапсуляции и наследования. Дополнительно вводится понятие **пакета** как совокупности атрибутов, операций, извещений и соответствующего поведения.

Класс объектов определяется позицией в **дереве наследования**, набором включенных пакетов и внешним интерфейсом, то есть видимыми снаружи атрибутами, операциями, извещениями и демонстрируемым поведением.

К числу концептуально важных можно отнести понятие **"проактивного"**, то есть **упреждающего управления**. Упреждающее управление основано на предсказании поведения системы на основе текущих данных и ранее накопленной информации. Простейший пример подобного управления – выдача сигнала о возможных проблемах с диском после серии программно-нейтрализуемых ошибок чтения/записи. В более сложном случае определенный характер рабочей нагрузки и действий пользователей может предшествовать резкому замедлению работы системы; адекватным управляющим воздействием могло бы стать понижение приоритетов некоторых заданий и извещение администратора о приближении кризиса.

**Возможности типичных систем**

Развитые системы управления имеют, если можно так выразиться, двухмерную настраиваемость – на нужды конкретных организаций и на изменения в информационных технологиях. Системы управления живут (по крайней мере, должны жить) долго. За это время в различных предметных областях администрирования (например, в области **резервного копирования**) наверняка появятся решения, превосходящие изначально заложенные в управляющий комплект. Последний должен уметь эволюционировать, причем разные его компоненты могут делать это с разной скоростью. Никакая жесткая, монолитная система такого не выдержит.

Единственный выход – наличие **каркаса**, с которого можно снимать старое и "навешивать" новое, не теряя эффективности управления.

Каркас как самостоятельный продукт необходим для достижения по крайней мере следующих целей:

- сглаживание разнородности управляемых информационных систем, предоставление унифицированных программных интерфейсов для быстрой разработки управляющих приложений;
- создание инфраструктуры управления, обеспечивающей наличие таких свойств, как поддержка распределенных конфигураций, масштабируемость, информационная безопасность и т.д.;
- предоставление функционально полезных универсальных сервисов, таких как планирование заданий, генерация отчетов и т.п.

Вопрос о том, что, помимо каркаса, должно входить в систему управления, является достаточно сложным. Во-первых, многие системы управления имеют мэйнфреймовое прошлое и попросту унаследовали некоторую функциональность, которая перестала быть необходимой. Во-вторых, для ряда функциональных задач появились отдельные, высококачественные решения, превосходящие аналогичные по назначению "штатные" компоненты. Видимо, с развитием объектного подхода, многоплатформенности важнейших сервисов и их взаимной совместимости, системы управления действительно превратятся в каркас. Пока же на их долю остается достаточно важных областей, а именно:

- управление **безопасностью**;
- управление **загрузкой**;
- управление **событиями**;
- управление **хранением данных**;
- управление **проблемными ситуациями**;
- генерация **отчетов**.

На уровне инфраструктуры присутствует решение еще одной важнейшей функциональной задачи – обеспечение **автоматического обнаружения** управляемых объектов, выявление их характеристик и связей между ними.

Отметим, что управление безопасностью в совокупности с соответствующим программным интерфейсом позволяет реализовать платформно-независимое **разграничение доступа** к объектам произвольной природы и (что очень важно) вынести функции безопасности из прикладных систем. Чтобы выяснить, разрешен ли доступ текущей политикой, приложению достаточно обратиться к **менеджеру безопасности** системы управления.

Менеджер безопасности осуществляет **идентификацию/аутентификацию** пользователей, контроль доступа к ресурсам и протоколирование неудачных попыток доступа. Можно считать, что менеджер безопасности встраивается в ядро операционных систем контролируемых элементов ИС, перехватывает соответствующие обращения и осуществляет свои проверки перед проверками, выполняемыми ОС, так что он создает еще один защитный рубеж, не отменяя, а дополняя защиту, реализуемую средствами ОС.

Развитые системы управления располагают централизованной базой, в которой хранится информация о контролируемой ИС и, в частности, некоторое представление о **политике безопасности**. Можно считать, что при каждой попытке доступа выполняется просмотр сохраненных в базе правил, в результате которого выясняется наличие у пользователя необходимых прав. Тем самым для проведения единой политики безопасности в рамках корпоративной информационной системы закладывается прочный технологический фундамент.

Хранение параметров безопасности в базе данных дает администраторам еще одно важное преимущество – возможность выполнения разнообразных запросов. Можно получить список ресурсов, доступных данному пользователю, список пользователей, имеющих доступ к данному ресурсу и т.п.

Одним из элементов обеспечения **высокой доступности** данных является подсистема автоматического управления хранением данных, выполняющая резервное копирование данных,

а также автоматическое отслеживание их перемещения между основными и резервными носителями.

Для обеспечения высокой доступности информационных сервисов используется управление загрузкой, которое можно подразделить на управление прохождением заданий и **контроль производительности**.

Контроль производительности – понятие многогранное. Сюда входят и оценка быстродействия компьютеров, и анализ пропускной способности сетей, и отслеживание числа одновременно поддерживаемых пользователей, и время реакции, и накопление и анализ статистики использования ресурсов. Обычно в распределенной системе соответствующие данные доступны "в принципе", они поставляются точечными средствами управления, но проблема получения целостной картины, как текущей, так и перспективной, остается весьма сложной. Решить ее способна только система управления корпоративного уровня.

Средства контроля производительности целесообразно разбить на две категории:

- выявление случаев неадекватного функционирования компонентов информационной системы и автоматическое реагирование на эти события;
- анализ тенденций изменения производительности системы и долгосрочное планирование.

Для функционирования обеих категорий средств необходимо выбрать отслеживаемые параметры и допустимые границы для них, выход за которые означает "неадекватность функционирования". После этого задача сводится к выявлению нетипичного поведения компонентов ИС, для чего могут применяться статистические методы.

Управление событиями (точнее, сообщениями о событиях) – это базовый механизм, позволяющий контролировать состояние информационных систем в реальном времени. Системы управления позволяют классифицировать события и назначать для некоторых из них специальные процедуры обработки. Тем самым реализуется важный принцип автоматического реагирования.