Министерство образования и науки Российской Федерации

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ ГИДРОМЕТЕОРОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ

Т.М. ТАТАРНИКОВА

ЗАЩИЩЁННЫЕ КОРПОРАТИВНЫЕ СЕТИ

Раздел: «Задачи по защите информации»

Учебное пособие

РГГМУ

Санкт-Петербург
2012

УДК 681.3.657.1

Татарникова Т.М. Защищённые корпоративные сети. Раздел: «Задачи по защите информации». Учебное пособие. - СПб., изд. РГГМУ, 2012.-114 с.

ISBN 978-5-86813-304-6

Рецензент В.В. Фомин, д-р техн. наук, проф. РГГУ им. А.И. Герцена

Учебное пособие содержит описание угроз информационным и вычислительным ресурсам корпоративных сетей, характеристику методов, средств и технологий противодуйствия сетевым атакам, а также краткие выводы по главам для закрепления пройденного материала.

Предназначено для подготовки специалистов по специальности “Информационная безопасность”.

Tatarnikova, T.M. Protection of information in corporate computer networks. The manual. - St. Petersburg, RSHU Publishers, 2012. – 114 p.

The book contains a description of threats to information and computing resources of corporate networks, as well as characterization of methods, tools and technologies to counter network attacks. It also includes brief conclusions for the chapters of the book to consolidate the material covered.

The manual is designed for training experts for the direction “Information Security”.

ISBN 978-5-86813-304-6

ВВЕДЕНИЕ

Учебное пособие написано в соответствии с программой дисциплины «Защищенные корпоративные сети», раздел «Задачи по защите корпоративных сетей».

Информация как результат обработки, передачи и хранения определяет действия людей, которые с ней работают и сложность технического и программного обеспечения, созданного человеком для защиты информации, так как последствия потери, подлога или хищения данных, хранящихся в вычислительных системах, а также нарушения работоспособности самих вычислительных средств могут быть очень высоки.

Обеспечение безопасности данных в корпоративных вычислительных сетях также подчиняется общей концепции информационной безопасности. Это концепция гласит, что составляющими информационной безопасности являются три задачи:

– обеспечение целостности информации;

– обеспечение доступности информации;

– обеспечение конфиденциальности.

Целостность информации условно подразделяется на статическую и динамическую. *Статическая* целостность информации предполагает неизменность информационных объектов от их исходного состояния, определяемого автором или источником информации.

Динамическая целостность информации включает вопросы корректного выполнения сложных действий с информационными потоками, например, анализ потока сообщений для выявления некорректных, контроль правильности передачи сообщений, подтверждение отдельных сообщений и др.

Целостность является важнейшим аспектом информационной безопасности в тех случаях, когда информация используется для управления различными процессами, например, техническими, социальными и т.д. Так, ошибка в управляющей программе приведет к остановке управляемой системы, неправильная трактовка закона может привести к его нарушениям, точно так же неточный перевод инструкции по применению лекарственного препарата может нанести вред здоровью. Все эти примеры иллюстрируют нарушение целостности информации, что может привести к катастрофическим последствиям. Именно поэтому целостность информации выделяется в качестве одной из базовых составляющих информационной безопасности.

Целостность – гарантия того, что информация сейчас существует в её исходном виде, то есть при ее хранении или передаче не было произведено несанкционированных изменений.

Конфиденциальность – самый проработанный у нас в стране аспект информационной безопасности. К сожалению, практическая реализация мер по обеспечению конфиденциальности современных информационных систем в России связана с серьезными трудностями. 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 и технические проблемы.

Конфиденциальная информация есть практически во всех организациях. Это может быть технология производства, программный продукт, анкетные данные сотрудников и др. Применительно к вычислительным системам в обязательном порядке конфиденциальными данными являются пароли для доступа к системе.

Конфиденциальность – гарантия доступности конкретной информации только тому кругу лиц, для кого она предназначена.

Доступность – это гарантия получения требуемой информации или информационной услуги пользователем за определенное время.

Нарушение каждой из трех категорий приводит к нарушению информационной безопасности в целом. Так, нарушение доступности приводит к отказу в доступе к информации, нарушение целостности - к фальсификации информации и, наконец, нарушение конфиденциальности - к раскрытию информации.

Выделение этих категорий в качестве базовых составляющих информационной безопасности обусловлено необходимостью реализации комплексного подхода при обеспечении режима информационной безопасности. Кроме этого, нарушение одной из этих категорий может привести к нарушению или полной бесполезности двух других. Например, хищение пароля для доступа к компьютеру (нарушение конфиденциальности) может привести к его блокировке, уничтожению данных (нарушение доступности информации) или фальсификации информации, содержащейся в памяти компьютера (нарушение целостности информации).

Угрозы целостности и конфиденциальности информации, а также работоспособности вычислительных систем могут быть выполнены при постоянном участии человека либо выполняться “злоумышленными” программами без непосредственного участия человека.

Задачи по защите от реализации угроз одинаковы независимо от их типа и включают следующие этапы:

1) преграждение несанкционированного доступа к корпоративным ресурсам;

2) невозможность несанкционированного использования компьютерных ресурсов, если доступ к ним все-таки осуществлен;

3) своевременное обнаружение факта несанкционированных действий и устранение причины, а также последствия их реализации.

Способы решения перечисленных задач по защите от несанкционированных действий со стороны людей и компьютерных программ существенно отличаются друг от друга.

Данное учебное пособие раскрывает существующие способы решения задач по защите информации в корпоративных вычислительных сетях и поддерживающие их технологии защиты.

1. ОСОБЕННОСТИ ОРГАНИЗАЦИИ КОРПОРАТИВНЫХ СЕТЕЙ

1.1. Структура корпоративной сети

Корпоративная вычислительная сеть - это сложный комплекс взаимосвязанных и согласованно функционирующих программных и аппаратных компонентов.

Изучение сети в целом предполагает знание принципов работы её отдельных элементов:

– компьютеров;

– коммуникационного оборудования;

– операционных систем;

– сетевых приложений.

Весь комплекс программно-аппаратных средств сети может быть описан многослойной моделью. В основе любой вычислительной сети лежит аппаратный слой. Его составляют стандартизованные компьютерные платформы.

Платформенный подход предполагает разработку не с «нуля», а с использованием специально разработанным для этой сети набором аппаратных решений, служб и примитивов, специально разработанных программных продуктов (программная платформа).

Основные слои корпоративной сети составляют:

Первый слой – компьютеры - от персональных до мэйнфреймов и супер ЭВМ. Например, для всемирного экологического мониторинга США построили суперкомпьютер с объявленным быстродействием $3 \cdot 10^{13}$ флопс.

Второй слой - это коммуникационное оборудование: кабельные системы, повторители, мосты, коммутаторы, маршрутизаторы и модульные концентраторы. Сегодня коммуникационное устройство может представлять собой сложный специализированный мультипроцессор, который нужно конфигурировать, оптимизировать, администрировать и обеспечивать его информационную безопасность. Оборудование второго слоя раньше считалось дополнительным, но сейчас по сложности реализации и выполняемым функциям стало основным как по влиянию на характеристики сети, так и по стоимости. Изучение принципов работы коммуникационного оборудования требует знания протоколов, используемых как в локальных, так и глобальных сетях.

Третьим слоем, образующим программную платформу сети, являются операционные системы. Операционные системы обеспечивают управление локальными и распределенными ресурсами, а именно:

– планирование ресурса: кому, когда, в каком количестве выделить данный ресурс, речь идёт о разделяемом ресурсе;

– мониторинг состояния ресурса, т.е. получение и анализ оперативной информации о состоянии ресурса: занят/свободен, в случае делимого ресурса – какая часть занята/свободна;

– взаимодействие с другими операционными системами;

– безопасность и защищенность данных.

Важным сервисом операционной системы являются различные сетевые приложения, такие как сетевые базы данных, почтовые системы, средства архивирования данных, системы автоматизации коллективной работы и др.

Сетевая операционная система выполняет:

– управление отдельными ресурсами: распределение оперативной памяти между процессами; планирование и диспетчеризация процессов управления процессорами;

– обмен сообщениями в сети: адресация и буферизация сообщений, выбор маршрута передачи, т.е. обеспечение транспортировки сообщений.

Функционально сетевая операционная система делится на две части:

– серверную, которая предоставляет собственные ресурсы локальных серверов в общее пользование. Она обеспечивает обработку запросов удаленного доступа к собственной файловой системе и базам данных, управляет очередями запросов удаленных пользователей к своим локальным серверам;

– клиентскую, которая обеспечивает доступ к удаленным ресурсам и услугам и их использование, прием ответов от удаленных серверов и преобразование их в локальный формат. Выполняет распознавание запроса, преобразование формы запроса.

Корпоративные сети называют также сетями масштаба предприятия. Такие сети объединяют большое количество компьютеров на всех территориях отдельного предприятия. Они могут быть сложно связаны и покрывать город, регион или даже континент.

Число пользователей и компьютеров может измеряться тысячами, а число серверов - сотнями, расстояния между сетями отдельных территорий могут оказаться такими, что становится необходимым использование глобальных связей. Пример корпоративной сети приведен на рис. 1.1.

Непременным атрибутом корпоративной сети является высокая степень гетерогенности. В корпоративной сети используются различные типы компьютеров от мэйнфреймов до персональных, несколько типов операционных систем и множество различных приложений. Неоднородные части корпоративной сети должны работать как единое целое, предоставляя пользователям по возможности прозрачный доступ ко всем необходимым ресурсам.

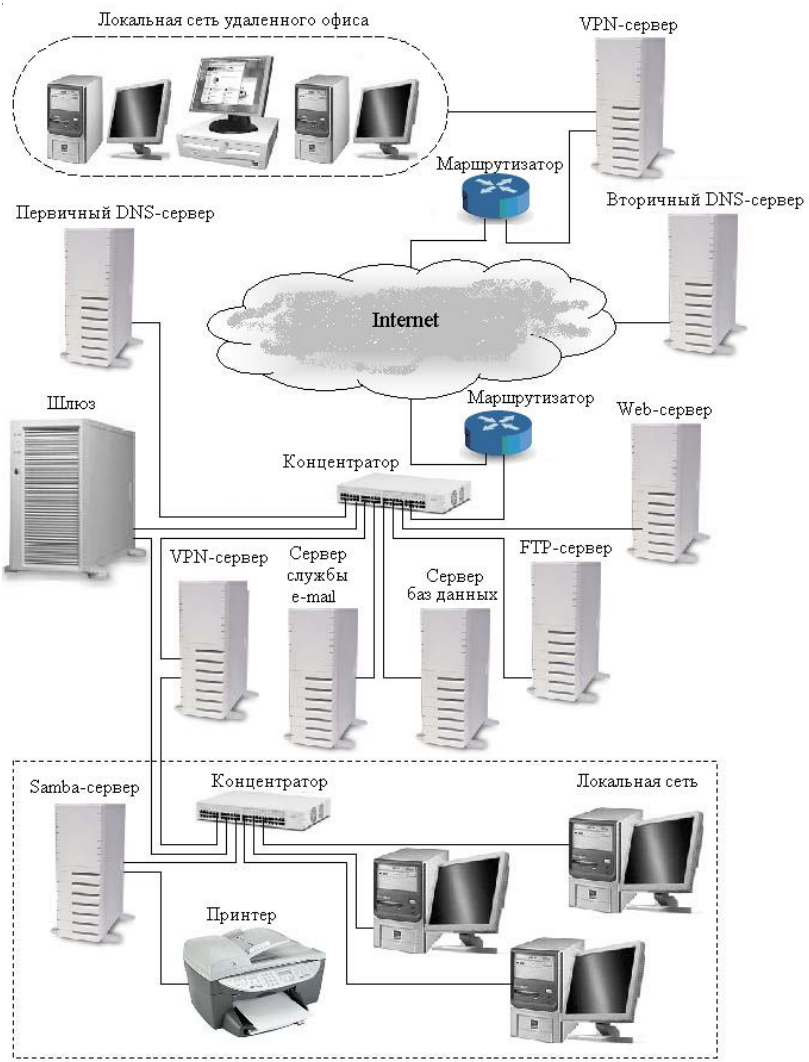

Рис 1.1. Пример корпоративной сети предприятия

1.2. Характеристики корпоративной сети

Производительность сети

Потенциально высокая производительность – это одно из основных свойств компьютерных сетей. Это свойство обеспечивается возмож-

ностью распараллеливания работ между несколькими компьютерами сети. Существует несколько основных характеристик производительности сети:

– время реакции;

– пропускная способность;

– задержка передачи и вариация задержки передачи.

Время реакции сети является интегральной характеристикой производительности сети с точки зрения пользователя. В общем случае время реакции определяется как интервал времени между возникновением запроса пользователя к какой-либо сетевой службе и получением ответа на этот запрос.

Очевидно, что значение этого показателя зависит от типа службы, к которой обращается пользователь, от того, какой пользователь и к какому серверу обращается, а также от текущего состояния элементов сети: загруженности сегментов, коммутаторов и маршрутизаторов, через которые проходит запрос, загруженности сервера и т.п.

Поэтому имеет смысл использовать также и *средневзвешенную оценку времени реакции сети,* усредняя этот показатель по пользователям, серверам и времени дня (от которого в значительной степени зависит загрузка сети).

Время реакции сети обычно складывается из нескольких составляющих (рис.1.2). В общем случае в него входит время подготовки запросов на клиентском компьютере (t_1), время передачи запросов между клиентом и сервером через сегменты сети и промежуточное коммуникационное оборудование ($t_2 - t_{n-1}$), время обработки запросов на сервере (t_n), время передачи ответов от сервера клиенту ($t`_{n-1} - t`_2$) и время обработки получаемых от сервера ответов на клиентском компьютере $t`_1$.

Пропускная способность отражает объём данных, переданных сетью или её частью в единицу времени. Пропускная способность уже не является пользовательской характеристикой, так как она говорит о скорости выполнения внутренних операций сети – передаче пакетов данных между узлами сети через различные коммуникационные устройства. Она непосредственно характеризует качество выполнения основной функции сети – транспортировки сообщений и поэтому чаще используется при анализе производительности сети, чем время реакции. Пропускная способность измеряется либо в битах в секунду, либо в пакетах в секунду. Пропускная способность может быть среднеймгновенной, максимальной.

Средняя пропускная способность вычисляется путём деления общего объёма переданных данных на время их передачи, причём выбирается достаточно длительный промежуток времени: час, день или полная неделя.

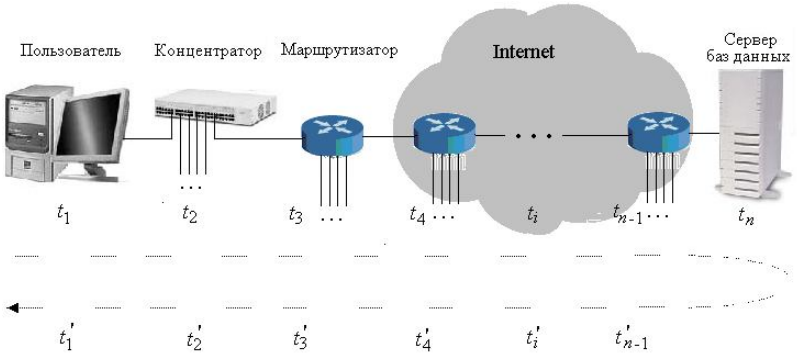

Рис 1.2. Время реакции сети

Мгновенная пропускная способность отличается от средней тем, что для усреднения выбирается очень маленький промежуток времени, например, 10 мс или 1 с.

Максимальная пропускная способность – это наибольшая мгновенная пропускная способность, зафиксированная в течение периода наблюдения. Пропускную способность можно измерять между любыми двумя узлами или точками сети, например, между клиентским компьютером и сервером, между входным и выходным портами маршрутизатора. Для анализа и настройки сети очень полезно знать данные о пропускной способности отдельных элементов сети.

Из-за последовательного характера передачи пакетов различными элементами сети общая пропускная способность любого составного пути в сети будет равна минимальной из пропускных способностей составляющих элементов маршрута.

Иногда полезно оперировать с общей пропускной способностью сети, которая определяется как среднее количество информации, переданной между всеми узлами сети в единицу времени. Этот показатель характеризует качество сети в целом, не дифференцируя его по отдельным сегментам или устройствам.

Задержка передачи определяется как задержка между моментом поступления пакета на вход какого-либо сетевого устройства или части сети и моментом появления его на выходе этого устройства. Этот параметр производительности по смыслу близок ко времени реакция сети, но отличается тем, что всегда характеризует только сетевые этапы обработки данных; без задержек обработки компьютерами сети. Например, задержку передачи запроса от пользователя к серверу баз данных на рис. 1.2 характеризуют временные составляющие от t_2 до t_{n-1} включительно.

Обычно качество сети характеризуют величинами максимальной задержки передачи и вариацией задержки. Не все типы трафика чувствительны к задержкам передачи, которые характерны для компьютерных сетей. Обычно задержки не превышают сотен миллисекунд, реже - нескольких секунд. Такого порядка задержки пакетов, порождаемых файловой службой, службой электронной почты или службой печати, мало влияют на качество этих служб с точки зрения пользователя сети. С другой стороны, такие же задержки пакетов, переносящих голосовые данные или видеоизображение, могут приводить к значительному снижению качества предоставляемой пользователю информации: возникновению эффекта «эха», невозможности разобрать некоторые слова, дрожанию изображения и т. п.

Пропускная способность и задержки передачи являются независимыми параметрами, так что сеть может обладать, например, высокой пропускной способностью, но вносить значительные задержки при передаче каждого пакета.

Надежность и безопасность сети

Важно различать несколько аспектов надежности. Для технических устройств используются такие показатели надежности, как среднее время наработки на отказ, вероятность отказа, интенсивность отказов. Однако эти показатели пригодны для оценки надежности простых элементов и устройств, которые могут находиться только в двух состояниях: работоспособном или неработоспособном. Сложные системы, состоящие из многих элементов, могут иметь и другие промежуточные состояния, которые названные характеристики не учитывают. В связи с этим для оценки надежности сложных систем применяется другой набор характеристик.

Для оценки надежности сетей используются различные характеристики, в том числе: *коэффициент готовности,* означающий долю времени, в течение которого система может быть использована; *безопасность,* т.е. есть способность системы защитить данные от несанкционированного доступа; отказоустойчивость – способность системы работать в условиях отказа некоторых её элементов.

Расширяемость означает возможность сравнительно легкого добавления отдельных элементов сети (пользователей, компьютеров, приложений, сервисов), наращивания длины сегментов сети и замены существующей аппаратуры более мощной.

Масштабируемость означает, что сеть позволяет наращивать количество узлов и протяженность связей в очень широких пределах, при этой производительность сети не ухудшается.

Прозрачность – свойство сети скрывать от пользователя детали своего внутреннего устройства, упрощая тем самым его работу в сети.

Управляемость сети подразумевает возможность централизованно контролировать состояние основных элементов сети, выявлять и разрешать проблемы, возникающие при работе сети, выполнять анализ производительности и планировать развитие сети.

Совместимость означает, что сеть способна включать в себя самое разнообразное программное и аппаратное обеспечение.

1.3. Адресация компьютеров в корпоративной сети

При объединении нескольких сетей в корпоративную сеть возникает проблема адресации в них компьютеров. Адрес - уникальный идентификатор компьютера в сети. Адрес узла сети должен уникально идентифицировать компьютер в сети любого масштаба. Схема назначения адресов должна сводить к минимуму ручной труд администратора и вероятность дублирования адресов. Адрес должен иметь иерархическую структуру, удобную для построения больших сетей, должен быть удобен для пользователей сети, а это значит, что он должен иметь символьное представление например, Server3 или www.cisco.com, должен иметь по возможности компактное представление, чтобы не перегружать память коммуникационной аппаратуры: сетевых адаптеров, маршрутизаторов и т.п.

Наибольшее распространение получили три схемы адресации узлов.

1. *Аппаратные* (*hardware*) *адреса* (*MAC-адреса*). Эти адреса предназначены для сети небольшого или среднего размера, они не имеют иерархической структуры. Такой адрес обычно используется только аппаратурой, поэтому его стараются сделать по возможности компактным и записывают в виде двоичного или шестнадцатеричного значения, например 0081005e24a8. Аппаратные адреса либо встраиваются в аппаратуру компанией-изготовителем, либо генерируются автоматически при каждом новом запуске оборудования, причём уникальность адреса в пределах сети обеспечивает оборудование. Использование аппаратных адресов связано с известным недостатком: при замене аппаратуры, например, сетевого адаптера, изменяется и адрес компьютера.

MAC-адрес имеет формат 6 байтов: старшие 3 байта – идентификатор фирмы производителя, младшие 3 байта назначаются уникальным образом самим производителем.

2. *Символьные адреса или DNS-имена.* Эти адреса предназначены для запоминания людьми и поэтому обычно несут смысловую нагрузку. Символьные адреса легко использовать как в небольших, так и крупных сетях. Для работы в больших сетях символьное имя может иметь сложную иерархическую структуру.

3. *Числовые составные адреса* (*IP-адреса*). Символьные имена удоб-ны для людей, но из-за переменного формата и потенциально большой длины их передача по сети не очень экономична. Поэтому в больших се-тях в качестве адресов узлов используют числовые составные адреса фиксированного и компактного форматов. В них поддерживается двух-уровневая иерархия, адрес делится на старшую часть – номер сети и младшую – номер узла. Такое деление позволяет передавать сообщения между сетями только на основании номера сети, а номер узла используется только после доставки сообщения в нужную сеть.

IP-адреса версии 4 представляют собой 32-битовые идентификаторы, структура которых оптимизирована для решения основной задачи протокола IP – маршрутизации. Обычно для удобства представления IP-адресов используется их цифровое написание в виде 4-х разрядов, разделённых точками, например, 192.168.123.132.

Для распознавания узлов, сетей и подсетей используется понятие «маска подсети». Для понимания, как маска подсети используется для определения адреса сети и узла, необходимо представить IP-адрес в двоичном обозначении.

IP-адрес 192.168.123.132 – это (в двоичном обозначении) 32-разрядный номер 11000000010100011110111000010 0. Такой номер сложно интерпретировать, поэтому лучше разбить его на четыре части по восемь двоичных знаков. Эти 8-разрядные секции называются «октеты». Тогда данный IP-адрес будет иметь вид: 11000000.10101000.01111011.10000100. Этот номер ненамного понятнее, поэтому в большинстве случаев следует преобразовывать двоичный адрес в формат разделенных точками разрядов (192.168.123.132). Десятичные числа, разделенные точками, и есть октеты, преобразованные из двоичного в десятичное обозначение.

В этом примере маской подсети является 255.255.255.0. Значение этого номера понятно, если знать, что число 255 в двоичном обозначении соответствует числу 11111111; таким образом, маской подсети является номер:

11111111.11111111.11111111.0000000

Расположив следующим образом IP-адрес и маску подсети, можно выделить составляющие сети и узла:

11000000.10101000.01111011.10000100 – IP-адрес (192.168.123.132)

11111111.11111111.11111111.00000000 – маска подсети (255.255.255.0)

Первые 24 разряда (число единиц в маске подсети) распознаются как адрес сети, а последние 8 разрядов (число оставшихся нолей в маске подсети) – адрес узла. Таким образом, получаем следующее:

11000000.10101000.01111011.00000000 – адрес сети (192.168.123.0)

00000000.00000000.00000000.10000100 – адрес узла (000.000.000.132)

или 192.168.123.0 – адрес сети, 0.0.0.132 – адрес узла.

Из данного примера с использованием маски подсети 255.255.255.0 видно, что код сети 192.168.123.0, а адрес узла 0.0.0.132. Когда пакет с конечным адресом 192.168.123.132 доставляется в сеть 192.168.123.0 (из локальной подсети или удаленной сети), компьютер получит его из сети и обработает.

Почти все десятичные маски подсети преобразовываются в двоичные числа, представленные единицами слева и нолями справа.

Символьные и числовые адрес относятся к классу логических адресов.

В современных сетях для адресации узлов применяются, как правило, одновременно все три приведенные выше схемы. Пользователи адресуют компьютеры символьными именами, которые автоматически заменяются в сообщениях, передаваемых по сети, на числовые номера. С помощью этих числовых номеров сообщения передаются из одной сети в другую, а после доставки сообщения в сеть назначения вместо числового номера используется аппаратный адрес компьютера. Проблемой установления соответствия между адресами различных типов занимается специальная служба разрешения имен DNS (Domain network service).

Служба DNS организует имена узлов в иерархию доменов. Домен - это набор узлов, в некотором смысле связанных между собой. Все эти узлы могут принадлежать к одной сети (например, все машины, входящие в состав локальной сети университета), все они также могут принадлежать к одной организации (например, все компьютеры, принадлежащие правительству), наконец, все они могут быть просто близко расположены друг от друга в географическом смысле. Например, все учебные заведения входят в состав домена edu, а каждому университету или колледжу соответствует свой субдомен, в состав которого входят все его компьютеры. Например электротехническому университету соответствует домен eltech.edu, а радиотехническому факультету этого университета – radio.eltech.edu. Все компьютеры, входящие в состав локальной сети данного факультета, должны содержать в своем названии имя домена. Например, полное имя компьютера person1 будет person1.radio.eltech.edu. Это имя называется полным доменным именем (fully qualified domain name, FQDN). Оно точно идентифицирует сетевой узел в рамках всемирной сети.

Иерархическая древовидная структура допускает использование в имени произвольного количества составных частей (рис. 1.3).

Иерархия доменных имен аналогична иерархии имен файлов, принятой во многих популярных файловых системах. Дерево имен начинается с корня, обозначаемого здесь точкой (**.**). Затем следует старшая символьная часть имени, вторая по старшинству символьная часть имени и т. д. Младшая часть имени соответствует конечному узлу сети. В отличие от имен файлов, при записи которых сначала указывается самая старшая составляющая, затем составляющая более низкого уровня и т. д., запись

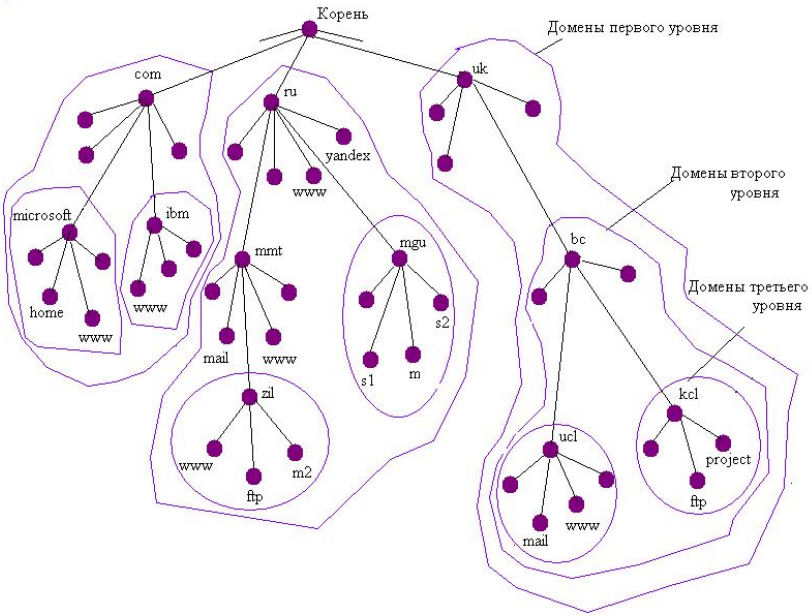

Рис 1.3. Пространство доменных имен

доменного имени начинается с самой младшей составляющей, а заканчивается самой старшей. Составные части доменного имени отделяются друг от друга точкой. Например, в имени partnering.microsoft.com составляющая partnering является именем одного из компьютеров в домене Microsoft.com.

Разделение имени на части позволяет разделить административную ответственность за назначение уникальных имен между различными людьми или организациями в пределах своего уровня иерархии. Так, для примера, приведенного на рис. 1.3, один человек может нести ответственность за то, чтобы все имена, которые имеют окончание «ru», имели уникальную следующую вниз по иерархии часть, т.е. все имена типа www.ru, mail.mmt.ru или m2.zi1.mmt.ru будут отличаться второй по старшинству частью.

Разделение административной ответственности позволяет решить проблему образования уникальных имен без взаимных консультаций между организациями, отвечающими за имена одного уровня иерархии. Очевидно, что должна существовать одна организация, отвечающая за назначение имен верхнего уровня иерархии.

Совокупность имен, у которых несколько старших составных частей совпадают, образуют *домен* имен *(domain)*. Например, имена wwwl.zil.

mmt.ru, ftp.zil.mmt.ru, yandex.ru и sl.mgu.ru входят в домен ru, так как все эти имена имеют одну общую старшую часть - имя ru. Другим примером является домен mgu.ru. Из представленных на рис. 1.3 имен в него входят имена sl.mgu.ru, s2.mgu.ru и m.mgu.ru. Этот домен образуют имена, у которых две старшие части всегда равны mgu.ru. Имя www.mmt.ru в домен mgu.ru не входит, так как имеет отличающуюся составляющую mmt.

Если один домен входит в другой домен как его составная часть, то такой домен могут называть поддоменом *(*subdomain*)*, хотя название домен за ним также остается. Обычно поддомен называют по имени той его старшей составляющей, которая отличает его от других поддоменов. Например, поддомен mmt.ru обычно называют поддоменом (или доменом) mmt. Имя поддомену назначает администратор вышестоящего домена. Хорошей аналогией домена является каталог файловой системы.

Если в каждом домене и поддомене обеспечивается уникальность имен следующего уровня иерархии, то и вся система имен будет состоять из уникальных имен.

По аналогии с файловой системой в доменной системе имен различают краткие имена, относительные имена и полные доменные имена. Краткое имя - это имя конечного узла сети - хоста или порта маршрутизатора. Краткое имя - это лист дерева имен. Относительное имя - это составное имя, начинающееся с некоторого уровня иерархии, но не самого верхнего. Например, wwi.zil - это относительное имя. *Полное доменное имя (fully qualified domain name, FQJDN)* включает составляющие всех уровней иерархии, начиная от краткого имени и кончая корневой точкой: wwl.zil.mmt.ru.

Необходимо подчеркнуть, что компьютеры входят в домен в соответствии со своими составными именами, при этом они могут иметь совершенно различные IP-адреса, принадлежащие к различным сетям и подсетям. Например, в домен mgu.ru могут входить хосты с адресами 132.13.-34.15, 201.22.100.33, 14.0.0.6. Доменная система имен реализована в сети Internet, но она может работать и как автономная система имен в крупной корпоративной сети, использующей стек TCP/IP, но не связанной с Internet.

В Internet корневой домен управляется центром InterNIC. Домены верхнего уровня назначаются для каждой страны, а также на организационной основе. Имена этих доменов должны следовать международному стандарту ISO 3166.

Для обозначения стран используются трехбуквенные и двухбуквенные аббревиатуры, так называемые *географические домены***.**

Каждая страна (государство) имеет свой географический домен из двух букв, например:

- для России – ru;
- для Австралии – au;

- для Англии – uk;
- для Бельгии – be.

Для различных типов организаций существуют *организационные домены,* использующие следующие обозначения:

- com - коммерческие организации (например, microsoft.com);
- edu - образовательные (например, mit.edu);
- gov - правительственные организации (например, nsf.gov);
- org - некоммерческие организации (например, fidonet.org);
- net - организации, поддерживающие сети (например, nsf.net).

Каждый домен администрируется отдельной организацией, которая обычно разбивает свой домен на поддомены и передает функции администрирования этих поддоменов другим организациям. Чтобы получить доменное имя, необходимо зарегистрироваться в какой-либо организации, которой InterNIC делегировал свои полномочия по распределению имен доменов. В России такой организацией является РосНИИРОС (Российский научно-исследовательский институт развития общественных сетей), которая отвечает за делегирование имен поддоменов в домене ru.

Соответствие между доменными именами и IP-адресами может устанавливаться как средствами локального узла, так и средствами централизованной службы. На раннем этапе развития Internet на каждом узле вручную создавался текстовый файл с известным именем hosts. Этот файл состоял из некоторого количества строк, каждая из которых содержала одну пару символов «IP-адрес - доменное имя», например, 102.54.94.97 - rhino.acme.com.

По мере роста Internet файлы hosts также росли и создание масштабируемого решения для разрешения имен стало необходимостью.

Таким решением стала централизованная служба DNS, основанная на распределенной базе отображений «доменное имя - IP-адрес». Служба DNS использует в своей работе протокол типа «клиент-сервер». В нем определены DNS-серверы и DNS-клиенты. DNS-серверы поддерживают распределенную базу отображений, а DNS-клиенты обращаются к серверам с запросами о разрешении доменного имени в IP-адрес.

Служба DNS использует текстовые файлы почти такого формата, как и файл hosts, и эти файлы администратор также подготавливает вручную. Однако служба DNS опирается на иерархию доменов, и каждый сервер службы DNS хранит только часть имен сети, а не все имена, как это происходит при использовании файлов hosts. При росте количества узлов в сети проблема масштабирования решается созданием новых доменов и поддоменов имен и добавлением в службу DNS новых серверов.

Для каждого домена имен создается свой DNS-сервер. Этот сервер хранит только имена, которые заканчиваются на следующем ниже уровне иерархии. (Аналогично каталогу файловой системы, который содержит

записи о файлах и подкаталогах, непосредственно в него «входящих»). Именно при такой организации службы DNS нагрузка по разрешению имен распределяется более-менее равномерно между всеми DNS-серверами сети. Например, этот сервер хранит отображения только имен типа mail.mmt.ru, www.mmt.ru, а все остальные отображения должны храниться на DNS-сервере поддомена zil.

Каждый DNS-сервер кроме таблицы отображений имен содержит ссылки на DNS-серверы своих поддоменов. Эти ссылки связывают отдельные DNS-серверы в единую службу DNS. Ссылки представляют собой IP-адреса соответствующих серверов. Для обслуживания корневого домена выделено несколько дублирующих друг друга DNS-серверов, IP-адреса которых являются широко известными (их можно узнать, например, в InterNIC).

Процедура разрешения DNS-имени во многом аналогична процедуре поиска файловой системой адреса файла по его символьному имени. Действительно, в обоих случаях составное имя отражает иерархическую структуру организации соответствующих справочников: каталогов файлов или таблиц DNS. Здесь домен и доменный DNS-сервер являются аналогом каталога файловой системы. Для доменных имен, так же как и для символьных имен файлов, характерна независимость именования от физического местоположения.

Процедура поиска адреса файла по символьному имени заключается в последовательном просмотре каталогов, начиная с корневого. При этом предварительно проверяется кэш и текущий каталог. Для определения IP-адреса по доменному имени также необходимо просмотреть все DNS-серверы, обслуживающие цепочку поддоменов, входящих в имя хоста, начиная с корневого домена. Существенным же отличием является то, что файловая система расположена на одном компьютере, а служба DNS по своей природе является распределенной.

Существуют две основные схемы разрешения DNS-имен. В первом варианте работу по поиску IP-адреса координирует DNS-клиент:

- DNS-клиент обращается к корневому DNS-серверу с указанием полного доменного имени;

- DNS-сервер отвечает, указывая адрес следующего DNS-сервера, обслуживающего домен верхнего уровня, заданный в старшей части запрошенного имени;

- DNS-клиент делает запрос следующего DNS-сервера, который отсылает его к DNS-серверу нужного поддомена, и т. д., пока не будет найден DNS-сервер, в котором хранится соответствие запрошенного имени IP-адресу. Этот сервер дает окончательный ответ клиенту. Такая схема взаимодействия называется нерекурсивной или итеративной, когда клиент сам итеративно выполняет последовательность запросов к разным серве-

рам имен. Так как эта схема загружает клиента достаточно сложной работой, то она применяется редко. Во втором варианте реализуется рекурсивная процедура:

- DNS-клиент запрашивает локальный DNS-сервер, т.е. тот сервер, который обслуживает поддомен, к которому принадлежит имя клиента. Если локальный DNS-сервер знает ответ, то он сразу же возвращает его клиенту; это может соответствовать случаю, когда запрошенное имя входит в тот же поддомен, что и имя клиента, а также может соответствовать случаю, когда сервер уже узнавал данное соответствие для другого клиента и сохранил его в своем кэше. Если же локальный сервер не знает ответ, то он выполняет итеративные запросы к корневому серверу и т. д. точно так же, как это делал клиент в первом варианте; получив ответ, он передает его клиенту, который всё это время просто ждал его от своего локального DNS-сервера.

1.4. Формат информационного пакета

Данные в сетях передаются блоками. Такие блоки принято называть *Пакеты* или *Кадры* (Packet, Frame). Каждый стандарт вычислительной сети определяет свой формат пакета. Они различаются по длине, расположению полей, однако, в независимости от типа сети, структура пакета одинакова (рис. 1.4).

Назначение полей:

- Преамбула (Preamble) - служит для синхронизации работы приёмника и передатчика;

- АП - Адрес Приёмника (DA - Destination Address) - адрес станции, которой направляется пакет;

- АИ - Адрес Источника (SA - Source Address) - адрес передающей станции;

- Поле Данных (Data) - содержит управляющую информацию, собственно данные либо пакет с другим протоколом (при передах через шлюзы);

Преамбула	АП	АИ	Поле данных	ПОО

Рис 1.4. Структура пакета

- ПОО - Поле Обнаружения Ошибок (CRC) - служит для определения достоверности полученной информации.

В качестве адресов могут использоваться логические или аппаратные (физические) адреса.

Логический адрес (Logical Address) определяется используемым протоколом обмена данными и может быть изменен в процессе работы.

С помощью логических адресов можно создать группы устройств, выполняющих одинаковые функции: серверы, маршрутизаторы и т.п. Это упрощает управление работой сети.

Физический адрес (Physical Address) определяется стандартом локальной сети, однозначно идентифицирует в сети данный узел и не может быть изменен после подключении устройства к сети. В Ethernet на сетевом адаптере устанавливается ПЗУ, в которой прошит физический адрес сетевого адаптера. Изменить его можно, только заменив микросхему ПЗУ.

В качестве адреса приёмника могут использоваться:

- широковещательный или Общий Адрес (Broadcast). Пакет с таким адресом принимается и обрабатывается всеми станциями сети. Широковещательный адрес используется и при логической адресации.

- групповой Адрес (Multicast). Пакет с таким адресом принимается и обрабатывается определенной группой станций. Например, только серверами, только маршрутизаторами и т.п. Этот адрес может быть только логическим.

- частный Адрес (Unicast или Private). Пакет с таким адресом принимается и обрабатывается только определенной станцией, адрес которой соответствует частному адресу. В качестве частных адресов используются логические или физические адреса.

1.5. Классы корпоративных сетей

IP-адреса распределены по классам. Наиболее распространены классы A, B и C. Классы D и E существуют, но обычно не используются конечными пользователями. Каждый из классов адресов имеет свою маску подсети по умолчанию.

В протоколе IP версии 4 (IPv4) имеется пять классов адресов, приведённых в табл. 1.1, где жирным шрифтом выделена старшая часть IP-адреса, указывающая номер сети.

Таблица 1.1

Классы адресов в версии IPv4

Класс	Первые биты IP-адреса	Наименьший номер сети	Наибольший номер сети	Максимальное число сетей	Максимальное число узлов в каждой сети
A	0	**0**.0.0.0	**127**.0.0.0	2^7-2	$2^{24}-2$
B	10	**128.0**.0.0	**191.255**.0.0	$2^{14}-2$	$2^{16}-2$
C	110	**192.0.0**.0	**223.255.255**.0	$2^{21}-2$	2^8-2
D	1110	**224**.0.0.0	**255**.255.255.255		
E	1111	**240**.0.0.0	**255**.255.255.255		

Большие сети используют адреса класса A, средние – класса B, маленькие – класса C.

В IPv4 существуют определенные соглашения об использовании адресов.

1. Сеть с номером 0.0.0.0 зарезервирована для использования в служебных сообщениях, а сеть с номером 127.0.0.0 используется для петлевого соединения (пересылки пакетов самим себе), поэтому общее количество сетей класса A равно 126.

2. Маршрутизация пакета в публичной сети всегда производится на основании классического IP-адреса номера сети, согласно табл. 1.1, поэтому адрес сети не может быть назначен ни одному узлу.

3. Адрес узла со всеми двоичными "1" предназначен для адресации всем узлам соответствующей сети (широковещательная рассылка), поэтому этот адрес не может быть назначен ни одному узлу. Совместно с пунктом 2 это означает, что число узлов в любой сети уменьшается на 2.

4. В каждом классе имеется диапазон сетевых адресов для частного использования, которые в публичных сетях отсутствуют. Они используются для построения локальных корпоративных сетей. В классе A – это сеть 10.0.0.0, в классе B – диапазоны сетей от 172.16.0.0 до 172.31.0.0, в классе C – диапазон сетей от 192.168.0.0. до 192.168.255.255.

Основное назначение адресов класса D – распространение информации по схеме "один-ко-многим" для групповой рассылки в Интернет аудио- и видеоинформации. Узел, который хочет осуществить рассылку, с помощью протокола группового обслуживания Интернет (Internet Group Management Protocol – IGMP) сообщает о создании в сети мультивещательной группы с определенным адресом. Устройства, которые хотят присоединиться к создаваемой группе, высылают свое подтверждение. Одно и то же устройство может входить в несколько групп, в одну и ту же группу могут входить устройства различных сетей.

Адреса класса E зарезервированы для будущих применений.

Маршрутизация пакета в публичной сети всегда производится на основании классического IP-адреса номера сети, согласно табл.1.1. Номер сети принято обозначать с помощью маски, количество лидирующих "единиц" в маске показывает число старших разрядов, которые определяют номер сети. Запись маски производится в формате IP-адреса. Таким образом, для сети класса A стандартная маска имеет вид 255.0.0.0 (в двоичном коде 11111111.00000000.00000000.00000000), для сети класса B – 255.255.0.0 (11111111.11111111.00000000.00000000), для сети класса C – 255.255.255.0 (11111111.11111111.11111111.00000000).

Наличие только четырех классов адресов часто бывает неудобно. Например, администратору необходимо создать сеть из 8000 узлов. Сеть класса C (254 узла) слишком мала, а сеть класса B (65534) слишком велика.

Эта проблема решается с помощью создания подсетей, путём переназначения части битов узла в качестве битов сети. Процесс заимствования части битов всегда начинается с крайнего левого бита.

Системный администратор, выделивший блок IP-адресов, возможно, администрирует сети, организованные не соответствующим для них образом. Например, имеется глобальная сеть с 150 узлами в трёх сетях (в разных городах), соединенных маршрутизатором TCP/IP. У каждой из этих трех сетей 50 узлов. Выделяем сеть класса С 192.168.123.0. Это значит, что адреса с 192.168.123.1 по 192.168.123.254 можно использовать для этих 150 узлов. Два адреса, которые нельзя использовать в данном примере, – 192.168.123.0 и 192.168.123.255, так как двоичные адреса с составляющей узла из одних единиц и нолей недопустимы. Адрес с 0 недопустим, поскольку он используется для определения сети без указания узла. Адрес с числом 255 (в двоичном обозначении адрес узла, состоящий из одних единиц) используется для доставки сообщения на каждый узел сети. Следует просто запомнить, что первый и последний адрес в любой сети и подсети не может быть присвоен отдельному узлу.

Теперь осталось дать IP-адреса 254 узлам. Это несложно, если все 150 компьютеров являются частью одной сети. Однако в данном примере 150 компьютеров работают в трех отдельных физических сетях. Вместо запроса на большее количество адресных блоков для каждой сети сеть разбивается на подсети, что позволяет использовать один блок адресов в нескольких физических сетях.

В данном случае сеть разбивается на четыре подсети с помощью маски подсети, которая увеличивает адрес сети и уменьшает возможный диапазон адресов узлов. Другими словами, мы «одалживаем» несколько разрядов, обычно используемых для адреса узла, и используем их для составляющей сети в адресе. Маска подсети 255.255.255.192 позволяет создать четыре сети с 62 узлами в каждой. Это возможно, поскольку в двоичном обозначении 255.255.255.192 – то же самое, что и 1111111.11111111.1111111.11000000. Первые две цифры последнего октета становятся адресами сети, поэтому появляются дополнительные сети 00000000 (0), 01000000 (64), 10000000 (128) и 11000000 (192). (Некоторые администраторы применяют только две из этих подсетей, используя номер 255.255.255.192 в качестве маски подсети. Для получения дополнительной информации по этому вопросу предлагается обратиться к RFC 1878. В этих четырех сетях последние 6 двоичных цифр можно использовать в качестве адресов узлов.

Использование маски подсети 255.255.255.192 преобразует сеть 192.168.123.0 в четыре сети: 192.168.123.0, 192.168.123.64, 192.168.123.128 и 192.168.123.192. Эти четыре сети будут иметь следующие действующие адреса узлов:

192.168.123.1- 62
192.168.123.65 -126
192.168.123.129 -190
192.168.123.193 -254

Не забываем, что двоичные адреса узлов с одними только единицами и нулями недействительны, поэтому нельзя использовать адреса со следующими числами в последнем октете: 0, 63; 64, 127; 128, 191; 192, 255.

Обратим внимание на следующие два адреса узлов: 192.168.123.71 и 192.168.123.133. Если использовать по умолчанию маску подсети класса С 255.255.255.0, оба адреса будут в сети 192.168.123.0. Однако, если использовать маску подсети 255.255.255.192, они окажутся в разных сетях: 192.168.123.71 – в сети 192.168.123.64, в то время как 192.168.123.133 – в сети 192.168.123.128.

Выводы

Корпоративная вычислительная сеть - это совокупность компьютеров, соединенных линиями связи. Линии связи образованы кабелями, сетевыми адаптерами и другими коммуникационными устройствами. Все сетевое оборудование работает под управлением системного и прикладного программного обеспечения.

Основная цель вычислительной сети - обеспечить её пользователям потенциальную возможность совместного использования ресурсов всех компьютеров.

В стеке TCP/IP используются три типа адресов: локальные (называемые также аппаратными), IP-адреса и символьные доменные имена. Все эти типы адресов присваиваются узлам составной сети независимо друг от друга.

IP-адрес имеет длину 4 байта и состоит из номера сети и номера узла. Для определения границы, отделяющей номер сети от номера узла, реализуются два подхода. Первый основан на понятии класса адреса, второй - на использовании масок.

Класс адреса определяется значениями нескольких первых бит адреса. В адресах класса А под номер сети отводится один байт, а остальные три байта - под номер узла, поэтому они используются в самых больших сетях. Для небольших сетей больше подходят адреса класса С, в которых номер сети занимает три байта, а для нумерации узлов может быть использован только один байт. Промежуточное положение занимают адреса класса В.

Другой способ определения, какая часть адреса является номером сети, а какая номером узла, основан на использовании маски. Маска - это число, которое используется в паре с IP-адресом; двоичная запись маски

содержит единицы в тех разрядах, которые в IP-адресе должны интерпретироваться как номер сети.

В стеке TCP/IP применяется доменная система символьных имен, которая имеет иерархическую древовидную структуру, допускающую использование в имени произвольного количества составных частей. Совокупность имен, у которых несколько старших составных частей совпадают, образуют домен имен. Доменные имена назначаются централизованно, если сеть является частью Internet, в противном случае - локально.

Соответствие между доменными именами и IP-адресами может устанавливаться как средствами локального хоста с использованием файла hosts, так и с помощью централизованной службы DNS, основанной на распределенной базе отображений «доменное имя - IP-адрес».

2. КОММУНИКАЦИОННОЕ ОБОРУДОВАНИЕ КОРПОРАТИВНЫХ СЕТЕЙ

В сетях с небольшим (10-30) количеством компьютеров используется одна из типовых топологий: общая шина, кольцо, звезда или полносвязная сеть. Все они обладают свойством однородности. Однородность структуры упрощает процедуру наращивания числа компьютеров, облегчает обслуживание и эксплуатацию сети.

При построении корпоративных сетей использование типовых структур порождает различные ограничения, к ним относятся:

- ограничения на длину связи между узлами;
- ограничения на количество узлов в сети;
- ограничения на интенсивность трафика, порождаемого узлами сети.

Для снятия этих ограничений используются специальные методы структуризации сети и специальное структурообразующее оборудование: повторители, концентраторы, мосты, коммутаторы, маршрутизаторы. Оборудование такого рода называют коммуникационным, имея в виду, что с помощью него отдельные сегменты сети взаимодействуют между собой.

Под физической структуризацией понимается конфигурация связей, образованных отдельными частями кабеля, а под логической - конфигурация информационных потоков между компьютерами сети. Физическая и логическая топологии могут совпадать, а могут и не совпадать.

2.1. Физическая структуризация сети

Простейшее из коммуникационных устройств – повторитель сигнала (repeator). Он используется для физического соединения сегментов кабеля локальной сети с целью увеличения общей длины сети. Повторитель передает сигналы, приходящие из одного сегмента сети в другие её сегменты (рис. 2.1) и позволяет преодолеть ограничения на длину линий связи за счёт улучшения качества передаваемого сигнала. Повторитель восстанавливает мощность сигнала амплитуды.

Повторитель, который имеет несколько портов и соединяет несколько физических сегментов, часто называют концентратором (concentrator) или хабом (hub). Концентратор - многопортовый повторитель. Концентратор всегда изменяет физическую топологию сети, но при этом оставляет без изменения её логическую топологию. Концентраторы образуют из отдельных физических отрезков кабеля общую среду передачи данных - логический сегмент (рис. 2.2). Поэтому сети, построенные на основе концентраторов, не могут расширяться в требуемых пределах - при определённом количестве компьютеров в сети или при появлении новых

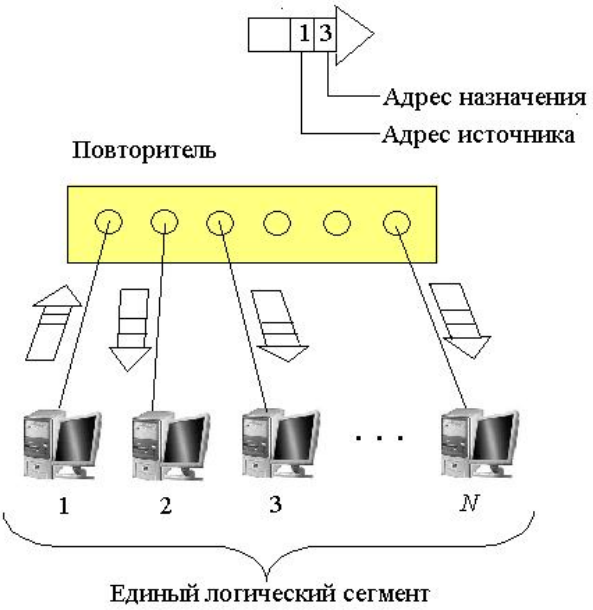

Рис 2.1. Повторитель Ethernet синхронно повторяет биты кадра на всех своих портах

приложений всегда происходит насыщение передающей среды, и задержки в её работе становятся недо-пустимыми. Эта проблема может быть решена путем логической струк-туризации сети с помощью мостов, коммутаторов и маршрутизаторов.

Важной проблемой, не решаемой путём физической структуризации, остается проблема перераспределения передаваемого трафика между различными физическими сегментами сети. Для повышения эффективности работы сети необходимо учитывать неоднородность информационных потоков.

2.2. Логическая структуризация сети

Крупные сети практически никогда не строятся без логической структуризации. Для отдельных сегментов и подсетей характерны типовые однородные топологии базовых технологий, и для их объединения используется оборудование, обеспечивающее локализацию трафика, мосты, коммутаторы, маршрутизаторы и шлюзы.

Распространение трафика, предназначенного для компьютеров некоторого сегмента сети, только в пределах этого сегмента, называется локализацией трафика. Логическая структуризация сети - это процесс

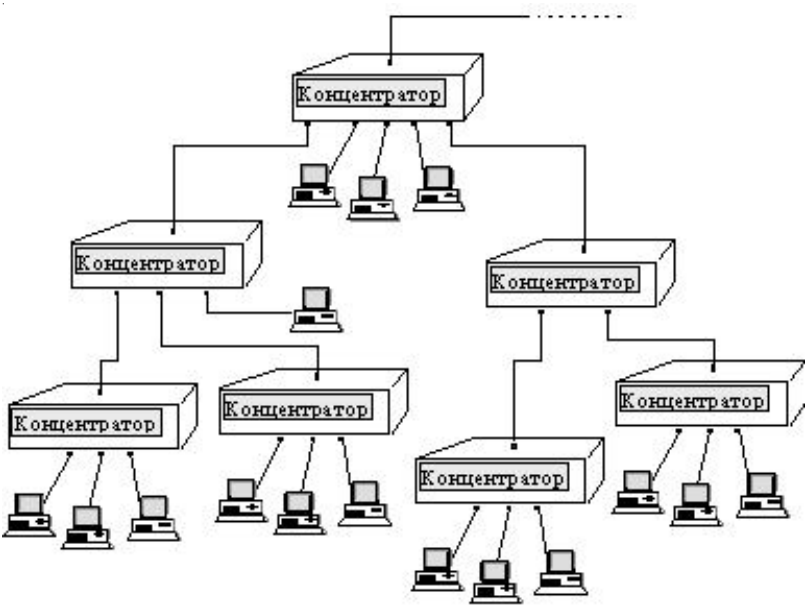

Рис 2.2. Логический сегмент, построенный с использованием концентраторов

разбиения сети на сегменты с локализованным трафиком.

***Мост* (bridge)**, а также его быстродействующий функциональный аналог - коммутатор (switching hub), делит общую среду передачи данных на логические сегменты. Логический сегмент образуется путём объединения нескольких физических сегментов (отрезков кабеля) с помощью одного или нескольких концентраторов. Каждый логический сегмент подключается к отдельному порту моста/коммутатора (рис. 2.3). При поступлении кадра на какой-либо из портов мост/коммутатор повторяет этот кадр, но не на всех портах, как это делает концентратор, а только на том порту, к которому подключен сегмент, содержащий компьютер-адресат.

Разница между мостом и коммутатором состоит в том, что мост в каждый момент времени может осуществлять передачу кадров только между одной парой портов, а коммутатор одновременно поддерживает потоки данных между всеми своими портами. Другими словами, мост передает кадры последовательно, а коммутатор - параллельно.

При работе коммутатора среда передачи данных каждого логического сегмента остается общей только для тех компьютеров, которые подключены к этому сегменту непосредственно. Коммутатор осуществляет связь сред передачи данных различных логических сегментов. Он передает кадры между логическими сегментами только при необходимости, т.е. только

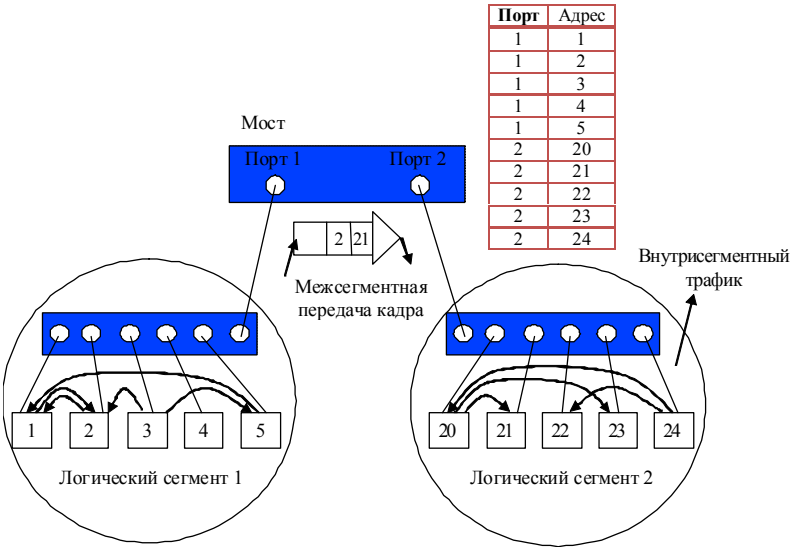

Рис 2.3. Разделение сети на логические сегменты

тогда, когда взаимодействующие компьютеры находятся в разных сегментах.

Существуют три архитектурных решения реализации коммутаторов, различающиеся способами комплексирования его функциональных модулей. Это коммутаторы на основе матрицы, общей шины и общей памяти.

Коммутаторы на основе матрицы.

Коммутатор матричного типа обеспечивает самый быстрый способ взаимодействия входных портов с выходными. Построение таких коммутаторов осуществляется на основе двоичных коммутационных элементов с двумя входами и двумя выходами.

Детальное представление одного из возможных вариантов реализации коммутационной матрицы для 8 портов дано на рис. 2.4. Во входном порту по адресу назначения, записанного в служебной части информационного кадра на основании просмотра адресной таблицы определяется номер выходного порта. Эта информация добавляется к байтам исходного кадра в виде специального ярлыка – тега (tag). Для данного примера тег представляет собой 3-х разрядное двоичное число, соответствующее номеру выходного порта.

Матрица состоит из трех уровней (каскадов) двоичных переключателей – коммутационных элементов, которые соединяют свой вход с одним из двух выходов в зависимости от значения бита тега.

Коммутационный элемент может работать в одном из двух режимов: «транзит» или «кросс» (рис. 2.5). Переключатели первого уровня управляются первым битом тега, второго – вторым, а третьего – третьим.

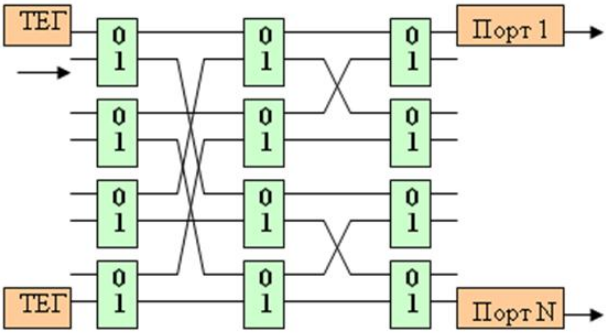

Рис 2.4. Вариант реализации коммутационной матрицы
для восьми портов

Известным недостатком этой технологии является отсутствие буферизации данных внутри коммутационной матрицы: если составной канал невозможно построить из-за занятости выходного порта или промежуточного КЭ, то данные должны накапливаться в буферных запоминающих устройствах (БЗУ) порта коммутатора.

Коммутаторы на базе общей шины.

Коммутаторы с общей шиной для связи входных портов с выходными применяют высокоскоростную шину, используемую в режиме разделения времени. В этой архитектуре шина (моноканал) пассивна, а активную роль выполняют специализированные процессоры портов.

Пример такой архитектуры приведен на рис. 2.6.

Кадр должен передаваться по шине небольшими частями, по несколько байт, чтобы передача кадров между несколькими портами происходила в псевдопараллельном режиме, не внося задержек в передачу кадра в целом. Размер такой ячейки данных определяется производителем коммутатора.

Во входном порту формируется тег, в котором указывает номер порта назначения и добавляется к информационной ячейке, переносимой по шине. Каждый выходной порт содержит фильтр тегов, который выбирает только те теги, которые предназначенные данному порту.

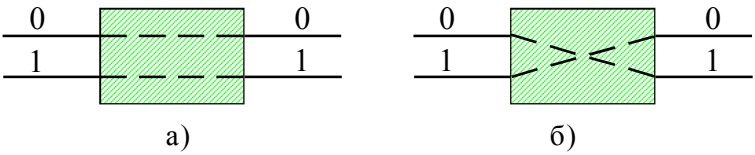

Рис 2.5. Режимы работы коммутационного элемента:
а) «транзит» б) «кросс»

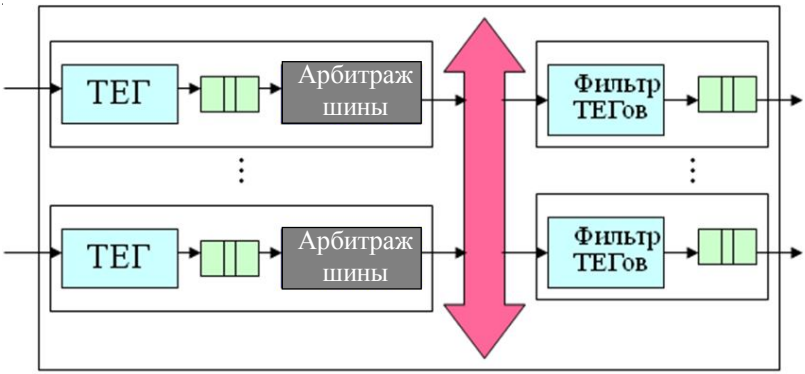

Рис 2.6. Структура коммутатора на базе общей шины

Шина не может осуществлять промежуточную буферизацию, но считается, что поскольку информационный кадр разбивается на небольшие ячейки, то задержек с начальным ожиданием доступности выходного порта в такой схеме не возникает.

Для того чтобы шина не была узким местом коммутатора, её производительность должна быть в несколько раз выше скорости поступления данных на входные порты.

Коммутатор с разделяемой памятью.

В коммутационной схеме с общей разделяемой памятью входные и выходные порты коммутатора соединены между собой не через шину, а через общую память. Пример такой архитектуры приведен на рис. 2.7.

Входные порты (конкретно, специализированные процессоры этих портов) соединяются с переключаемым входом разделяемой памяти, а выходные порты соединяются с переключаемым выходом этой памяти. Переключением входа и выхода разделяемой памяти управляет менеджер очередей. Менеджер организует в разделяемой памяти несколько очередей данных, по одной для каждого выходного порта. Входные порты передают менеджеру запросы на запись данных в очередь того порта, который соответствует адресу назначения пакета. Менеджер по очереди подключает вход памяти к одному из входных портов, и тот переписывает данные в очередь определенного выходного порта. По мере заполнения очередей менеджер производит также поочередное подключение выхода разделяемой памяти к выходным портам, и данные из очереди переписываются в выходной буфер соответствующего порта.К недостаткам коммутаторов этого типа относят их высокую сложность и стоимость изготовления.

Маршрутизатор **(router)**. Маршрутизаторы образуют логические сегменты посредством явной адресации, поскольку используют не плоские аппаратные, а составные числовые адреса. В этих адресах имеется

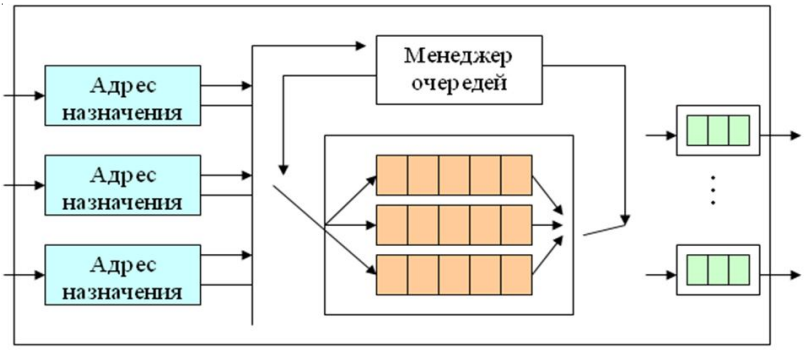

Рис 2.7. Структура коммутатора на базе общей памяти

поле номера сети, так что все компьютеры, у которых значение этого поля одинаково, принадлежат к одному сегменту, называемому в данном случае подсетью.

Кроме локализации трафика маршрутизаторы выполняют ещё много других полезных функций, они осуществляют выбор наиболее рационального маршрута из нескольких возможных.

Маршрутизатор имеет в своем распоряжении базу топологической информации о том, между какими подсетями корпоративной сети имеются связи и в каком состоянии (работоспособном или нет) они находятся. На основании такой карты сети маршрутизатор принимает решение о выборе одного из нескольких возможных маршрутов доставки пакета адресату в соответствии с таблицей маршрутов.

Таблица маршрутов в общем случае содержит следующие колонки:

– Пункт назначения (Destination) – определяет IP-адрес сети назначения.

– Маска сети (Subnet Mask) – задает количество лидирующих бит в IP-адресе, которые определяют адрес сети.

– Пункт пересылки (Next Hop) – задает IP-адрес интерфейса следующего маршрутизатора, на который следует направить поступивший пакет.

– Интерфейс (Interface) – задает собственный выходной порт, маршрутизатора, на который следует направить поступивший пакет.

– Метрика (Metric) – задает предпочтение в выборе альтернативных маршрутов. Маршруты с меньшей метрикой более предпочтительны.

Например, на рис. 2.8 для связи рабочей станций PC2 локальной вычислительной сети ЛВС1 и PC1 сети ЛВС6 через глобальные вычислительные сети (ГВС) имеются два маршрута: М1-М5-М7 и М1-М6-М7.

Рассмотрим пример работы маршрутизатора в качестве межсетевого узла, сопрягающего разные сети внутри корпоративной.

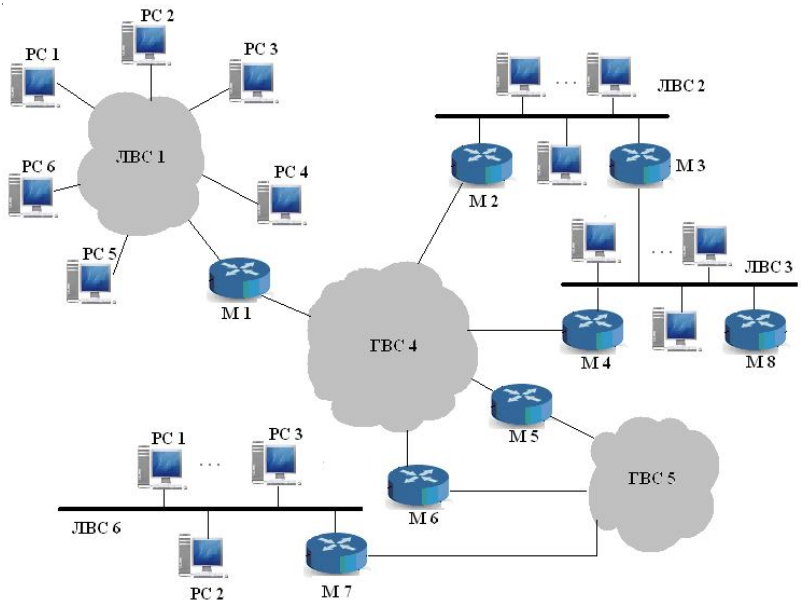

Рис 2.8. Структура интерсети, построенной на основе маршрутизаторов

Пусть, например, для нашего случая администратор получил адрес сети 135.38.0.0 (адрес класса B, маска сети 255.255.0.0). В этой сети 16 битов выделено под адрес сети и 16 битов – на адрес узла. Администратору необходимо иметь 8000 узлов, на это необходимо выделить только 13 битов на адреса узлов ($2^{13} = 8192$), следовательно, оставшиеся 16 – 13 = 3 бита адреса узла можно переназначить как адрес сети. Тогда маска образованной подсети в двоичном коде будет иметь вид 11111111.11111111. **111**00000.00000000 или 255.255.224.0 (жирным шрифтом выделены заимствованные биты адреса узлов класса B). В результате такого деления получим следующие адреса подсетей, приведенные в табл. 2.1.

Две полученные подсети 135.38.0.0 и 135.38.224.0 использовать нельзя, так как сетевой адрес первой подсети 135.38.0.0 совпадает с адресом исходной классической сети класса B, а адрес широковещательной рассылки внутри второй подсети 135.38.224.0 совпадает с адресом широковещательной рассылки исходной классической сети класса B. Теперь одну из оставшихся 6 подсетей (например, подсеть 135.38.32.0) администратор использует для своих нужд, а оставшиеся 5 сетей может отдать другому администратору.

Архитектура местоположения подсети 135.38.32.0 приведена на рис. 2.9. Для обслуживания подсети 135.38.32.0 маршрутизатор M (рис. 2.9) использует таблицу маршрутов (табл. 2.2).

Таблица 2.1

Адреса подсетей

Номер сети	Число узлов в подсети
~~**10000111 00100110 000**00000 00000000~~ ~~**135. 38. 0.** 0~~	~~8190~~
10000111 00100110 00100000 00000000 **135. 38. 32.** 0	8190
10000111 00100110 01000000 00000000 **135. 38. 64.** 0	8190
10000111 00100110 01100000 00000000 **135. 38. 96.** 0	8190
10000111 00100110 10000000 00000000 **135. 38. 128.** 0	8190
10000111 00100110 10100000 00000000 **135. 38. 160.** 0	8190
10000111 00100110 11000000 00000000 **135. 38. 192.** 0	8190
~~**10000111 00100110 111**00000 00000000~~ ~~**135. 38. 224.** 0~~	~~8190~~

Для определения дальнейшего маршрута следования поступившего пакета маршрутизатор производит следующие операции.

1. Поступивший IP-адрес в двоичном коде с помощью логической операции “И” складывается поразрядно с маской сети первой строки таблицы маршрутизации. Правило сложения разрядов с помощью логической операции “И”: 0+0=0, 0+1=0, 1+0=0, 1+1=1.

2. Полученный в результате сложения адрес сети сравнивается с IP-адресом пункта назначения первой строки. При их совпадении поступивший пакет направляется на интерфейс s1.

3. В случае несовпадения те же операции, начиная с пункта 1, проделываются с последующими строками маршрутной таблицы, если они имеются.

4. Все поступившие пакты из подсети 135.38.32.0 направляются по умолчанию на интерфейс s2.

Например, пусть из публичной сети поступили следующие пакеты с IP-адресами назначения: 135.38.16.15, 135.38.56.211, 135.38.92.10. Определим, на какие интерфейсы они будут направлены.

Для определения номера адресуемой сети складываем по “И” IP-адрес назначения первого пакета с маской сети, получаем

IP = 135.38.16.15 $\Rightarrow$ 10000111. 00100110. 00010000. 00001111

Mask = 255.255.224.0 $\Rightarrow$ 11111111. 11111111. 11100000. 00000000

Destination = $10000111.\ 00100110.\ 00000000.\ 00000000_2 \Rightarrow$ 135. 38.0.0.

Такой записи в таблице маршрутизации нет, пакет уничтожается.

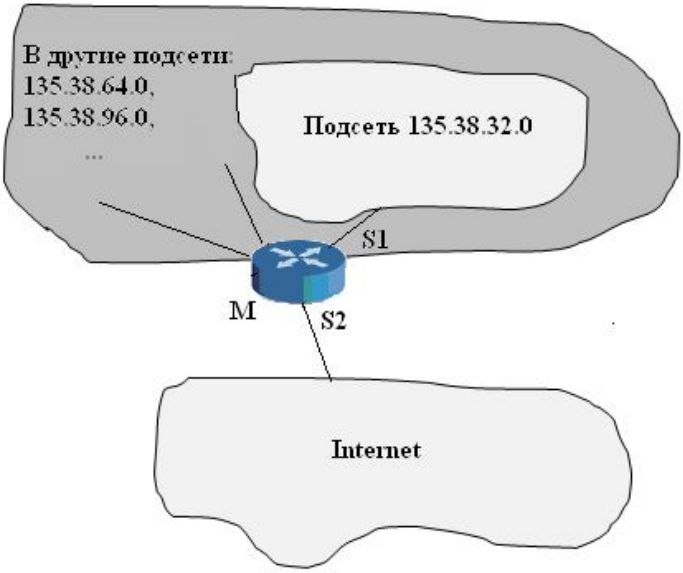

Рис 2.9 Архитектура местоположения подсети 135.38.32.0

Для второго пакета

IP = 135.38.56.211 ⇒ 10000111. 00100110. 00111000. 11010011

Mask = 255.255.224.0 ⇒ 11111111. 11111111. 11100000. 00000000

Destination = 10000111. 00100110. 00100000. 00000000$_2$ ⇒ 135.38.32.0.

Пакет будет направлен на интерфейс s1.

Для третьего пакета

IP = 135.38.92.10 ⇒ 10000111. 00100110. 01011100. 00001010

Mask = 255.255.224.0 ⇒ 11111111. 11111111. 11100000. 00000000

Destination = 10000111. 00100110. 01000000. 00000000$_2$ ⇒ 135.38.64.0.

Такой записи в таблице маршрутизации нет, пакет уничтожается.

Другой важной способностью маршрутизатора является способность связывать в единую сеть подсети, построенные с использованием разных сетевых технологий, например, Ethernet и X.25.

Маршрутизаторы позволяют объединять сети с различными принципами организации в единую internet-сеть. Название интерсеть подчёркивает ту особенность, что образованное с помощью маршрутизаторов объединение компьютеров представляет собой совокупность нескольких сетей, сохраняющих большую степень автономности. В каждой из сетей, образующих интерсеть, сохраняются присущие им принципы адресации узлов и протоколы обмена информацией. Поэтому маршрутизаторы

Таблица 2.2

Таблица маршрутов для обслуживания подсети 135.38.32.0

Пункт назначения (Destination)	Маска сети (Subnet Mask)	Пункт пересылки (Next Hop)	Интерфейс (Interface)	Метрика (Metric)
135.38.32.0	255.255.224.0	0.0.0.0	s1	1
Default		0.0.0.0	s2	20

могут объединять не только локальные сети с различной технологией, но и локальные сети с глобальными.

В результате, маршрутизатор оказывается сложным интеллектуальным устройством, построенным на базе нескольких мощных процессоров. Такой специализированный мультипроцессор работает, как правило, под управлением специализированной операционной системы. Типичная архитектура маршрутизатора/шлюза приведена на рис. 2.10.

Когда пакет прибывает на маршрутизатор или шлюз, то в порту отрезаются заголовки и концевики кадров и остаются только поля данных, которые и передаются в общее поле памяти маршрутизатора. Далее анализируется заголовок пакета, и, в соответствии с записанным в нем заданием, строится последовательный алгоритм (цепочка команд) обработки пакета протокольными процессами. В маршрутизаторе и шлюзе одновременно выполняется несколько заданий, так как протоколы могут иметь свои копии по уровням Эталонной модели взаимодействия открытых систем (ЭМВОС) и общая память разделена на секции. Это обеспечивает

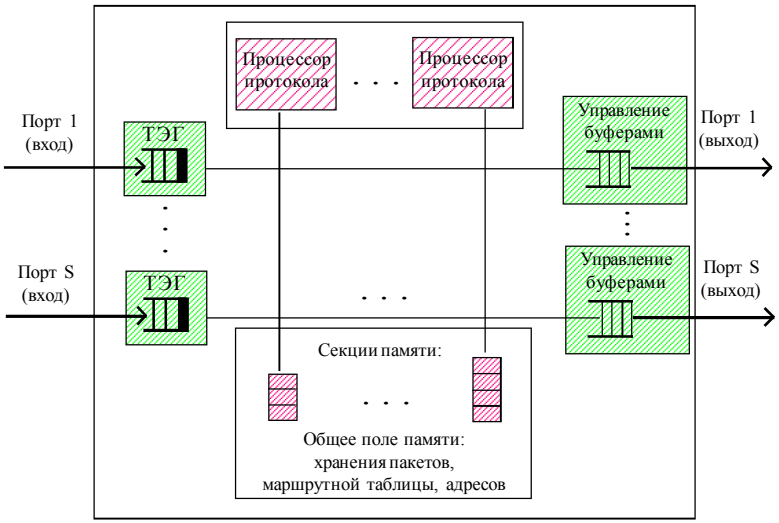

Рис. 2.10. Архитектура маршрутизатора/шлюза

параллельную обработку пакетов в маршрутизаторе.

Кроме перечисленных устройств отдельные части сети может соединять ***шлюз*** **(gateway)**. Обычно основной причиной, по которой в сети используют шлюз, является необходимость объединить сети с разными типами системного и прикладного программного обеспечения. Например, шлюз e-mail может переводить электронные письма в формат SMS-сообщений для мобильных телефонов.

Выводы

Для снятия ограничений на длину связей между узлами вычислительной сети, на количество узлов в сети и на интенсивность трафика, порождаемого узлами сети, используются специальные методы структуризации сети и специальное структурообразующее оборудование - повторители, концентраторы, мосты, коммутаторы, маршрутизаторы. С помощью этого оборудования отдельные сегменты сети взаимодействуют между собой, поэтому называется коммуникационным.

Существуют два варианта структуризации: физическая и логическая. Под физической структуризацией понимается конфигурация связей, образованных отдельными частями кабеля. Под логической структуризацией понимается конфигурация информационных потоков между компьютерами сети. Физическая структуризация реализуется с помощью повторителей и концентраторов, логическая – с помощью мостов, коммутаторов разной архитектуры, маршрутизаторов, шлюзов.

3. УГРОЗЫ БЕЗОПАСНОСТИ КОРПОРАТИВНЫМ СЕТЯМ

3.1. Жизненный цикл сетевой атаки

Вопросы сетевой безопасности и раннего обнаружения атак с каждым днём становятся всё более и более насущными как для частных пользователей, корпоративных сетей, так и для средних и крупных операторов связи. Сетевые атаки последнее время приобретают массовый характер. Известны случаи вывода из строя крупных всемирных порталов, банков, оборонных ведомств.

Прежде чем начать разговор о способах выявления атак, определим, что же она собой представляет. Итак, атака - это совокупность действий злоумышленника, приводящих к нарушению информационной безопасности компьютерной сети (КС). Результатом успешно реализованной атаки может стать, например, несанкционированный доступ нарушителя к информации, хранящейся в КС, потеря работоспособности системы или искажение содержимого (данных) КС. В качестве потенциальных целей могут рассматриваться серверы, рабочие станции пользователей или коммуникационное оборудование сети. В общем случае любая атака

Рис 3.1. Жизненный цикл типовой атаки

может быть разделена на четыре стадии, как показано на рис. 3.1.

Рекогносцировка. На этом этапе нарушитель старается получить как можно больше информации об объекте атаки, чтобы на её основе спланировать дальнейшие этапы вторжения. Примерами такой информации являются: тип и версия операционной системы, установленной на хостах информационной системы (ИС), список пользователей, зарегистрированных в системе, сведения об используемом прикладном програм-

мном обеспечении (ПО) и др.

Вторжение. На этом этапе нарушитель получает несанкционированный доступ к ресурсам тех хостов, на которые совершается атака.

Атакующее воздействие. На данной стадии реализуются те цели, ради которых и предпринималась атака. Например, нарушение работоспособности ИС, кража конфиденциальной информации, хранимой в системе, удаление или модификация данных и др. При этом атакующий часто выполняет операции, направленные на удаление следов его присутствия в ИС.

Развитие атаки. Когда злоумышленник стремиться расширить объекты атаки, чтобы продолжить несанкционированные действия на других составляющих ИС.

Рассмотрим конкретные примеры, демонстрирующие, как могут реализовываться эти стадии. На этапе рекогносцировки действия нарушителя могут быть нацелены на получение следующих данных:

- информация о структуре и топологии компьютерной сети. Для получения данных этого типа нарушитель может воспользоваться стандартными утилитами типа «traceroute», входящими в состав практически любой операционной системы (ОС), которые позволяют сформировать список IP-адресов транзитных маршрутизаторов вплоть до хоста-объекта нападения. Информацию о структуре ИС злоумышленник может получить и путём обращения к DNS-серверу;

- информация о типе ОС. Один из наиболее распространённых методов определения типа ОС основан на том факте, что различные системы по-разному реализуют правила взаимодействия с сетевыми протоколами: при одних и тех же сетевых запросах разные ОС отправляют в ответ отличные друг от друга данные, используя которые можно с большой долей вероятности определить характеристики атакуемой ОС и даже тип аппаратной платформы;

- информация о типе прикладных сервисов. Эти знания нарушитель может получить путём сканирования открытых портов и анализа заголовков ответов, полученных от этих служб;

- информация о зарегистрированных пользователях. Данные этого типа злоумышленник может извлечь из базы данных SNMP MIB, установленной на рабочих станциях и серверах компьютерной сети.

Когда необходимая информация собрана, можно начинать *вторжение.* Любое вторжение основано на наличии в компьютерной сети уязвимостей, и использование хотя бы одной из них открывает злоумышленнику вход в систему.

Примеры уязвимостей: ошибки при конфигурировании сетевых служб компьютерной сети, ошибки в программном обеспечении, использование «слабых» и «нестойких» паролей, отсутствие необходимых средств

защиты. Результат: нарушитель получает несанкционированный доступ к ресурсам атакованного узла, что позволяет ему перейти к следующей стадии информационной атаки.

На стадии *атакующего воздействия* нарушитель выполняет те действия, которые позволяют ему осуществить цель атаки. Например, извлекает из системы управления базами данных (СУБД) атакованного узла сети конфиденциальную информацию.

После атакующего воздействия нарушитель может перевести атаку в фазу её *дальнейшего развития*. Для этого в систему обычно несанкционированно внедряется программа, с помощью которой можно организовать атаку на другие узлы ИС. После установки такой программы опять начинается первый этап атаки - сбор информации о следующей цели.

В основном атаки имеют распределенный массовый характер, когда на информационный узел сети осуществляется одновременное обращение с десятков тысяч (и более) зараженных компьютеров. Узел не справляется с таким количеством одновременных запросов и выходит из строя, прекращая выполнять свои основные функции. Данный вид атаки является самым популярным и именуется «Отказ в обслуживании» или DoS атакой (Denied of Service attack). По статистике, 90% всех отказов атакуемых узлов были инициированы именно DoS-атаками.

Другим неприятным моментом DoS-атаки является огромное количество входящего сетевого трафика, который зачастую оплачивается, что влечёт большие расходы компании, подвергшейся атаки.

3.2. Классификация угроз безопасности функционирования корпоративных сетей

Общая классификация типов угроз, которым подвергается компьютерная сеть, приведена на рис. 3.2. Классификация произведена по степени риска, т.е. объёму наносимого ущерба в случае успешной реализации атаки как на информацию, защищаемую в сети, так и, собственно на саму сеть.

Дадим более подробную характеристику каждому классу угроз и приведем примеры наиболее представительных из них и их характерные признаки.

“Отказ в обслуживании”

Отказ в обслуживании - это любое действие или последовательность действий, которая приводит любую часть атакуемой системы к выходу из строя, при котором та перестает выполнять свои функции. Причиной может быть несанкционированный доступ, задержка в обслуживании и т.д. К угрозам этой группы относятся:

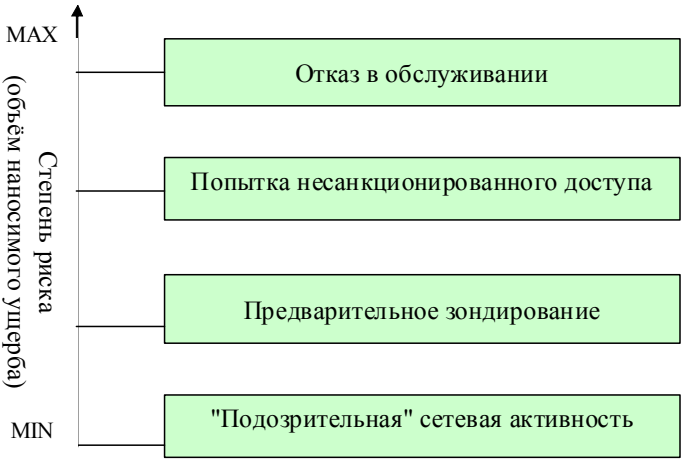

Рис 3.2. Классификация видов угроз безопасности функционирования корпоративных сетей

Фрагментация данных. При передаче пакета данных протокола IP по сети может осуществляться деление этого пакета на несколько фрагментов. Впоследствии, при достижении адресата, пакет восстанавливается из этих фрагментов. Злоумышленник может инициировать посылку большого числа фрагментов, что приводит к переполнению программных буферов на приёмной стороне и, в ряде случаев, к аварийному завершению системы. Данная атака эффективна против компьютеров с операционной системой Windows. Другие варианты подобных атак используют неправильные смещения в IP-фрагментах, что приводит к некорректному выделению памяти, переполнению буферов и, в конечном итоге, к сбоям в работе систем.

---

[1] ICMP (англ. *Internet Control Message Protocol* - межсетевой протокол управляющих сообщений)-сетевой протокол, входящий в стек протоколов TCP/IP. В основном ICMP используется для передачи сообщений об ошибках и других исключительных ситуациях, возникших при передаче данных. Сетевым администраторам ICMP предоставляет средства тестирования достижимости узлов сети, которые представляют собой эхо-протокол. Компьютер или маршрутизатор посылают по интерсети эхо-запрос, в котором указывают IP-адрес узла, достижимость которого нужно проверить. Узел, который получает эхо-запрос, формирует и отправляет эхо-ответ и возвращает сообщение узлу - отправителю запроса. Так как эхо-запрос и эхо-ответ передаются по сети внутри IP-пакетов, то их успешная доставка означает нормальное функционирование всей транспортной системы интерсети. Во многих операционных системах используется утилита *ping,* которая предназначена для тестирования достижимости узлов. Эта утилита обычно посылает серию эхо-запросов к тестируемому узлу и предоставляет пользователю статистику об утерянных эхо-ответах и среднем времени реакции сети на запросы.

Методы противодействия: для выявления таких атак необходимо осуществлять и анализировать сборку пакетов "на лету", а это существенно повышает требования к аппаратному обеспечению (производительности процессора, памяти и т.п.) средства контроля информационных потоков.

Ping flooding (от англ. *ping-flood*, дословно: наводнение пакетами) - тип атаки на сетевое оборудование, ставящий своей целью отказ в обслуживании. Ключевой особенностью (по сравнению с остальными видами флуд-атак) является возможность осуществления атаки «бытовыми средствами», такими как программы и утилиты, входящие в состав домашних/офисных версий операционных систем.

Злоумышленник посылает продолжительные серии эхо-запросов по протоколу ICMP [1]. Атакуемая система тратит свои вычислительные ресурсы, отвечая на эти запросы. Таким образом, существенно снижается производительность системы и возрастает загруженность каналов связи.

Методы противодействия: блокирования трафика с отдельных узлов и сетей, отключение ответов на ICMP-запросы, понижение приоритета обработки ICMP-сообщений, отбрасывание или фильтрация ICMP-трафика средствами межсетевого экрана.

Атака UDP bomb основана на передаче пакетов по протоколу UDP [2], в которых содержится неправильный формат служебных полей. Некоторые версии сетевого программного обеспечения приводят при получении подобного пакета к аварийному завершению системы.

Методы противодействия: для распознавания данной атаки необхо-димо анализировать форматы служебных полей.

SYN flooding - одна из разновидностей сетевых атак типа «отказ в обслуживании», которая заключается в отправке большого количества

---

[2] Протокол UDP (User Datagram Protocol) - прозрачный протокол в группе протоколов Internet, как и протокол IP предоставляет прикладным процессам транспортные услуги, но без подтверждения гарантий доставки. Протокол UDP обеспечивает ненадежную доставку датаграмм и не поддерживает соединений из конца в конец. Это означает, что пакеты могут быть потеряны, продублированы или прийти не в том порядке, в котором они были отправлены. К заголовку IP-пакета он добавляет два поля, одно из которых, поле "порт", обеспечивает мультиплексирование информации между разными прикладными процессами, а другое поле - "контрольная сумма" - позволяет поддерживать целостность данных.

[3] TCP (Transmission Control Protocol) - это транспортный механизм, предоставляющий поток данных с предварительной установкой соединения, за счёт этого дающий уверенность в достоверности получаемых данных, осуществляющий повторный запрос данных в случае потери данных и устраняющий дублирование при получении двух копий одного. В отличие от UDP гарантирует, что приложение получит данные точно в такой же последовательности, в какой они были отправлены, и без потерь. Протокол TCP один из основных сетевых транспортных протоколов Интернета.

SYN-запросов (запросов на подключение по протоколу TCP[3]) в достаточно короткий срок.

При установлении соединения по протоколу TCP приёмная сторона, получив запрос на соединение (пакет с флагом SYN), посылает источнику ответ (пакет с флагами SYN и ACK) о готовности установить это соединение. При этом система размещает в своей памяти служебную запись об устанавливаемом соединении и хранит её до тех пор, пока источник не пришлет пакет-подтверждение либо не истечет время ожидания данного пакета.Злоумышленник посылает большое количество запросов на установление соединения без передачи пакетов подтверждения. Вследствие этого происходит резкое снижение производительности и при определённых обстоятельствах аварийное завершение системы.

Методы противодействия: для распознавания данной атаки необходимо анализировать загрузку канала и определять причины снижения пропускной способности.

Атака SMURF - это одна из наиболее опасных атак DoS, поскольку при её реализации на целевые узлы осуществляется усиленное воздействие. Эффект усиления возникает из-за рассылки направленных широковещательных ping-запросов на узлы сети, которые должны сгенерировать ответные сообщения.

Атака SMURF заключается в передаче в сеть широ-ковещательных запросов от имени компьютера-жертвы. В результате компьютеры, принявшие такие широковещательные пакеты, отвечают компьютеру-жертве, что приводит к существенному снижение пропуск-ной способности канала связи и, в ряде случаев, к полной изоляции атакуемой сети.

Методы противодействия: для распознавания данной атаки необ-ходимо анализировать загрузку канала и определять причины снижения пропускной способности.

Атака Land использует уязвимости реализаций стека TCP/IP в некоторых операционных системах. Она заключается в передаче на открытый порт компьютера-жертвы TCP-пакета с установленным флагом SYN, причём исходный адрес и порт такого пакета соответственно равны адресу и порту атакуемого компьютера. Это приводит к тому, что компьютер-жертва пытается установить соединение сам с собой, в результате чего сильно возрастает загрузка процессора и может произойти “зависание” или перезагрузка системы. Успешное применение такой атаки к маршрутизатору может вывести из строя всю сеть организации.

Методы противодействия: защититься от данной атаки можно, на-пример, фильтруя пакеты между внутренней сетью и сетью Интернет по правилу, указывающему подавлять пакеты, пришедшие из сети Интернет, но с исходными адресами компьютеров внутренней сети.

Атака DNS flooding - это атака, направленная на сервера имен Интернет. Она заключается в передаче большого числа DNS запросов и приводит к тому, что у пользователей нет возможности обращаться к сервису имен и, следовательно, обеспечивается невозможность работы обычных пользователей.

Методы противодействия: для выявления данной атаки необходимо анализировать загрузку сервера DNS и выявлять источники запросов.

Попытка несанкционированного доступа

Попытка несанкционированного доступа представляет собой любое действие или последовательность действий, которая приводит к попытке чтения файлов или выполнения команд в обход установленной политики безопасности, также включает попытки злоумышленника получить привилегии, большие, чем установлены администратором системы. К этой группе угроз относятся:

Переполнение буферов. Данная атака заключается в посылке на компьютер-жертву сообщения, приводящего к переполнению буфера-приемника. Переполнение буфера возможно из-за отсутствия проверки длины принимаемых данных в большинстве приложений. При переполнении буфера обычно происходит затирание части кода или других данных приложения, в связи с чем появляется возможность исполнения собственного кода, подготовленного злоумышленником, на компьютере-жертве (возможно, в привилегированном режиме). Атака, связанная с переполнением буферов приложений и нацеленная на осуществление несанкционированного доступа, является одной из самых распространённых.

Методы противодействия: для выявления и противодействия атакам такого типа необходимо осуществлять фильтрацию протоколов прикладного уровня с учётом особенностей конкретных приложений.

Атака DNSspoofing. Результатом данной атаки является внесение навязываемого соответствия между IP-адресом и доменным именем в кэш-памяти сервера DNS. В результате успешного проведения такой атаки все пользователи сервера DNS получат неверную информацию о доменных именах и IP-адресах. Данная атака характеризуется большим количеством DNS пакетов с одним и тем же доменным именем. Это связано с необходимостью подбора некоторых параметров DNS обмена.

Методы противодействия: для выявления такой атаки необходимо анализировать содержимое DNS трафика.

Атака IPspoofing (*syslog*). Связана с подменой исходного IP-адреса в сети Интернет. Действие атаки заключается в передаче на компьютер-жертву сообщения от имени другого компьютера внутренней сети. По-

скольку протокол syslog используется для ведения системных журналов, путём передачи ложных сообщений на компьютер-жертву можно навязать информацию или скрыть следы несанкционированного доступа.

Методы противодействия: выявление атак, связанных с подменой IP-адресов, возможно при контроле получения на одном из локальных узлов пакета с исходным адресом этого же узла или при контроле получения на внешнем узле пакетов с IP-адресами внутренней сети.

Предварительное зондирование

Предварительное зондирование - любое действие или последовательность действий по получению информации из или о сети (например, имена и пароли пользователей), используемые в дальнейшем для осуществления неавторизованного доступа.

Сканирование Half scan. Атака состоит в незаметном выявлении каналов информационного воздействия на систему. Злоумышленник посылает пакеты установления соединения и при получении ответов от системы сбрасывает соединение. При этом стандартные средства не фиксируют попытку установления соединения, в то время как злоумышленник определяет присутствие служб на определенных портах.

Методы противодействия: для определения сканирования необходимо фиксировать попытки установления соединения.

Сканирование сети посредством DNS. Известно, что прежде чем начинать атаку, злоумышленники осуществляют выявление целей, т.е. выявление компьютеров, которые будут жертвами атаки, а также компьютеров, которые осуществляют информационный обмен с жертвами. Одним из способов выявления целей заключается в опросе сервера имён и получение от него всей имеющейся информации о домене.

Методы противодействия: для определения такого сканирования необходимо анализировать DNS-запросы, приходящие, быть может, от разных DNS-серверов, но за определенный, фиксированный промежуток времени.

Сканирование TCP портов. Сканирование портов представляет собой известный метод распознавания конфигурации компьютера и доступных сервисов. Для успешного проведения атак злоумышленникам необходимо знать, какие службы установлены на компьютере-жертве.

Методы противодействия: выявить данную атаку можно путем полного перехвата трафика TCP и анализа номеров портов. Кроме того, существуют возможности противодействия TCP сканированию. Это противодействие можно осуществлять, например, передавая TCP пакеты от имени сканируемого компьютера на компьютер злоумышленника, таким образом, вводя его в заблуждение.

Сканирование UDP портов. Другой вид сканирования портов основывается на использовании протокола UDP и заключается в следующем: на сканируемый компьютер передаётся UDP-пакет, адресованный к порту, который проверяется на предмет доступности. Если порт недоступен, то в ответ приходит сообщение о недоступности, в противном случае ответа нет. Данный вид сканирования достаточно эффективен. Он позволяет за короткое время сканировать все порты на компьютере-жертве. Кроме того, этот вид сканирования широко известен в Интернет.

Методы противодействия: противодействовать сканированию данного рода возможно путём передачи сообщений о недоступности порта на компьютер злоумышленника.

Подозрительная сетевая активность

Подозрительная сетевая активность представляет класс атак, характерной особенностью которых является наличие сетевого трафика, выходящего за рамки определения "стандартного" трафика. Подобная активность может указывать на подозрительные действия, осуществляемые в сети. К данной группе угроз относятся:

Использование маршрутизации источника. При пересылке пакетов IP по сети Интернет обычно используется динамическая маршрутизация, то есть решение о направлении дальнейшего продвижения каждого конкретного пакета по сети принимается каждым отдельным маршрутизатором в момент получения данного пакета исходя из алгоритма маршрутизации. Однако, существует и возможность указания в пакете конкретного маршрута, по которому должен быть послан пакет. Эта возможность может быть использована злоумышленником для обхода элементов защиты (например, межсетевого экрана) локальной сети.

Методы противодействия: для противодействия подобной атаке необходимо запретить маршрутизацию источника внутри локальной сети.

Дублирующий IP-адрес. Каждая система в сети Интернет характеризуется своим уникальным цифровым адресом. Если обнаруживается, что одна система (имеющая другой MAC-адрес) посылает пакет с IP-адресом, совпадающий с адресом другой, то значит одна из этих систем была неправильно настроена. Подобная техника может применяться атакующей стороной, для незаметной подмены работающей "доверенной" системы и осуществления атак от её имени.

Методы противодействия: защита от данной атаки может быть реализована путем хранения для всех активных систем пары адресов (IP и MAC) и анализа адресов в заголовках пакетов, пересылаемых по локальной сети.

3. 3. Программные закладки

В процессе передачи или хранения данных в корпоративной сети актуальным становится вопрос защиты информационных массивов, баз данных и программных средств от различных воздействий.

Для защиты от несанкционированного доступа к информации во время её передачи и хранения используются криптографические методы и, соответственно, средства (программные или аппаратные) для их реализации. Для поддержания целостности и авторизации сообщений в электронном виде - системы цифровой аутентификации (цифровая подпись).

Кроме того, при работе этих средств защиты необходимо обеспечить потенциальное невмешательства присутствующих прикладных или системных программ в процесс обработки информации средствами защиты. Приведём несколько примеров:

Служба безопасности одного из крупных коммерческих банков зарегистрировала действия, которые могли быть проделаны лишь при знании некоторой конфиденциальной информации, которая хранилась в виде базы данных в зашифрованном виде. Уязвимость алгоритма шифрования не была доказана, утери паролей для шифрования выявлено не было. Изучение компьютеров выявило наличие в загрузочных секторах ПЭВМ своеобразных вирусов - программ, которые сохраняли вводимую с клавиатуры информацию (в том числе и пароли для шифрования) в несколько зарезервированных для этого секторов.

Другой пример: одно из малых предприятий, занятое посреднической деятельностью и, как следствие, обладающее конфиденциальной информацией о предметах возможных сделок, также использовало шифрование как средство защиты своих интересов. В данном случае использовался стандарт ГОСТ 28147-89. Для шифрования использовалась плата Krypton-3, реализующая данный алгоритм шифрования, который, как известно, обеспечивает гарантированную защиту информации. Через некоторое время выяснилось, что шифруемая информация становится известной третьей стороне. А ещё через некоторое время была выявлена внедренная в систему закладка, подменившая собой плату шифрования. При этом алгоритм ГОСТ был заменен другим, крайне простым и легко читаемым без ключа.

Третий пример: спор противников и сторонников программы Pretty Good Privacy (PGP) был завершен написанием закладки, подделывающей электронную подпись под файлами, выполненную данной программой.

Во всех трех случаях программа никак не проявляла себя внешне, однако сохраняла весь ввод с клавиатуры в скрытом файле. В дальнейшем злоумышленникам требовалось лишь считать файл или просмотреть сектора, чтобы узнать пароли и по ним расшифровать интересовавшие

их данные. Такие программы большинство специалистов сразу назвали закладкой - по аналогии с незаметно внедряемыми в помещения миниатюрными электронными системами звукового подслушивания или телевизионного наблюдения.

Программная закладка - это компьютерная программа, которая обладает хотя бы одним из трёх перечисленных ниже свойств:

– внесение произвольного искажения в коды других программ, находящихся в оперативной памяти компьютера (программная закладка первого типа);

– перенос фрагментов информации из одних областей оперативной или внешней памяти компьютера в другие (программная закладка второго типа);

– произвольного искажения выводимой на внешние компьютерные устройства или в канал связи информации, полученной в результате работы других программ (программная закладка третьего типа).

Программные закладки можно также классифицировать по методу их внедрения в компьютерную систему:

– программно-аппаратные закладки, ассоциированные с аппаратной средой компьютера (их средой обитания, как правило, является BIOS - набор программ, записанных в виде машинного кода в постоянном запоминающем устройстве - ПЗУ);

– загрузочные закладки, ассоциированные с программами первичной загрузки, которые располагаются в загрузочных секторах (загрузочными являются несколько секторов диска, из которых в процессе выполнения начальной загрузки компьютер считывает программу, берущую на себя управление с целью последующей загрузки операционной системы);

– драйверные закладки, ассоциированные с драйверами (компьютерными файлами, в которых содержится информация, необходимая операционной системе для управления подключенными к компьютеру периферийными устройствами);

– прикладные закладки, ассоциированные с прикладным программным обеспечением общего назначения (текстовые редакторы, утилиты, антивирусные мониторы и программные оболочки);

– исполняемые закладки, ассоциированные с исполняемыми программными модулями, содержащими код этой закладки (чаще всего эти модули представляют собой пакетные файлы, которые состоят из команд операционной системы, выполняемые друг за другом, как если бы их набирали с клавиатуры компьютера);

– закладки-имитаторы, интерфейс которых совпадает с интерфейсом некоторых служебных программ, требующих ввода конфиденциальной информации (паролей, криптографических ключей, номеров кредитных карточек);

- замаскированные закладки, которые маскируются под программные средства оптимизации работы компьютера (файловые архиваторы, дисковые дефрагментаторы) или под программы игрового и развлекательного назначения.

При рассмотрении воздействия закладки и программ защиты информации уместны аналогии с взаимодействием вируса и прикладной программы. Вирус может присоединиться к исполняемому файлу, соответствующим образом изменив его, может уничтожить некоторые файлы или встроиться в цепочку драйверов.

Компьютерный вирус – программа, которая может включать в другие программы свою, иногда модифицированную копию, способную к дальнейшему размножению и выполнению вредных воздействий. Вирус может присоединиться к исполняемому файлу, соответствующим образом изменив его, может уничтожить некоторые файлы или встроиться в цепочку драйверов. Основная цель компьютерных вирусов – дестабилизация работы, уничтожение программ или наборов данных, т.е. нанесение максимального ущерба вычислительной системе. При этом, действие компьютерных вирусов не является направленным: воздействию подвергаются все программные объекты, предусмотренные алгоритмом работы вируса вне зависимости от содержащейся в них информации. Как правило, компьютерные вирусы попадают в вычислительную систему в процессе её эксплуатации вместе с получаемыми из различных источников (внешние диски, компьютерные сети) программами или данными.

Закладка отличается более направленным и тонким воздействием. Особенностью закладок может быть и то, что они фактически становятся неотделимы от прикладных или системных программ, если внедрены в них на стадии разработки или путем обратного проектирования (путём дисассемблирования прикладной программы, внедрения кода закладки и последующей компиляции).

Программная закладка (ПЗ) - это программа или фрагмент программы, скрытно внедряемый в защищенную систему и позволяющий лицу или процессу, внедрившему его, осуществлять в дальнейшем несанкционированные действия к тем или иным ресурсам защищенной системы. Основной целью программных закладок может быть получение или создание условий для получения информации о паролях, кодовых комбинациях, обрабатываемых данных и передача собранных сведений заданному адресу по сети, электронной почте и т.д. или просто копирование в другие, легко доступные области памяти. Действие программных закладок является селективным и направленным только на программные объекты, содержащие интересующую информацию. Программные закладки могут попадать в вычислительную систему как на этапе её разработки, так и в процессе её эксплуатации. Особенностью закладок может

быть и то, что они фактически становятся неотделимы от прикладных или системных программ, если внедрены в них на стадии разработки путём обратного проектирования (путем дисассемблирования прикладной программы, внедрения кода закладки и последующей компиляции).

Объединяет вирусы и программные закладки то, что и вирус, и закладка должны скрывать свое присутствие в операционной среде компьютерной системы.Для того чтобы программная закладка могла произвести какие-либо осмысленные действия по отношению к другим программам или данным, процессор должен приступить к исполнению команд, входящих в состав кода программной закладки. Это возможно только при одновременном выполнении двух следующих условий:

1. Программная закладка должна попасть в оперативную память компьютера (в случае программной закладки первого типа это проделывается либо до начала работы другой программы, которая является предметом конечного воздействия закладки, либо во время её работы);

2. Исполнение кода закладки, находящейся в оперативной памяти, начинается при выполнении ряда условий, которые называются активизирующими.

Это достигается путем анализа и обработки закладкой общих относительно закладки и прикладной программы воздействий, как правило, прерываний. Причём прерывания должны сопровождать работу прикладной программы или работу всей ПЭВМ. В качестве таких прерываний можно выделить:

- прерывания от таймера ПЭВМ;
- прерывания от внешних устройств;
- прерывания от клавиатуры;
- прерывания при работе с диском;
- прерывания операционной среды, (в том числе прерывания при работе с файлами и запуск исполняемых модулей).

В противном случае активизации кода закладки не произойдет, и он не сможет оказать какого-либо воздействия на работу программы защиты информации. Кроме того, возможен случай, когда при запуске программы (в этом случае активизирующим событием является запуск программы) закладка разрушает некоторую часть кода программы, уже загруженной в оперативную память, и, возможно, систему контроля целостности кода или контроля иных событий и на этом заканчивает свою работу.

Таким образом, можно выделить закладки:

- резидентного типа, которые находятся в памяти постоянно с некоторого момента времени до окончания сеанса работы персонального компьютера (выключения питания или перезагрузки). Они начинают работу при загрузке операционной среды или запуске некоторой программы (которая по традиции называется вирусоносителем), а также запущена отдельно;

- нерезидентного типа, которые начинают работу по аналогичному событию, но заканчивают её самостоятельно по истечении некоторого промежутка времени или некоторому событию, при этом выгружая себя из памяти целиком.

Существуют три основные группы деструктивных действий, которые могут осуществляться программными закладками:

- копирование информации пользователя компьютерной системы (паролей, криптографических ключей, кодов доступа, конфиденциальных электронных документов) в ее оперативной или внешней памяти либо в памяти другой компьютерной системы, подключенной к ней посредством локальной или глобальной компьютерной сети;

- изменение алгоритмов функционирования системных, прикладных и служебных программ (например, введение изменений в программу разграничения доступа может привести к тому, что она разрешит вход в компьютерную систему всем без исключения пользователям вне зависимости от правильности введенного пароля);

- навязывание определенных режимов работы (например, блокирование записи на диск при стирании информации, причем она, естественно, не уничтожается и может быть впоследствии скопирована злоумышленником).

У всех без исключения программных закладок, независимо от метода их внедрения в компьютерную систему, срока пребывания в оперативной памяти и выполняемых действий, есть одна важная общая черта: в программных закладках обязательно присутствует операция записи в оперативную или внешнюю память компьютерной системы. Без этой операции никакое негативное влияние программной закладки на компьютерную систему невозможно. Ясно, что для целенаправленного воздействия программная закладка должна выполнять также операцию чтения, иначе в ней может быть реализована только функция разрушения (например, стирание или замена информации в определенных секторах жесткого диска).

Жизненный цикл ПЗ выглядит следующим образом (рис. 3.3). Обобщённо функционирование уже внедренной программной закладки можно представить в виде схемы, приведенной на рис. 3.4.

Структурно ПЗ состоит из четырех основных функциональных блоков: исследование, активизация, проявление деструктивных действий (разрушения) и маскировка. В маскировку может входить и защита от исследования закладки - противодействие программной закладки обнаружению. Причём наличие блоков исследования и маскировки не обязательно. Начало функционирования программной закладки осуществляется в момент передачи управления программе-носителю закладки (на схеме - точка начало функционирования программной закладки).

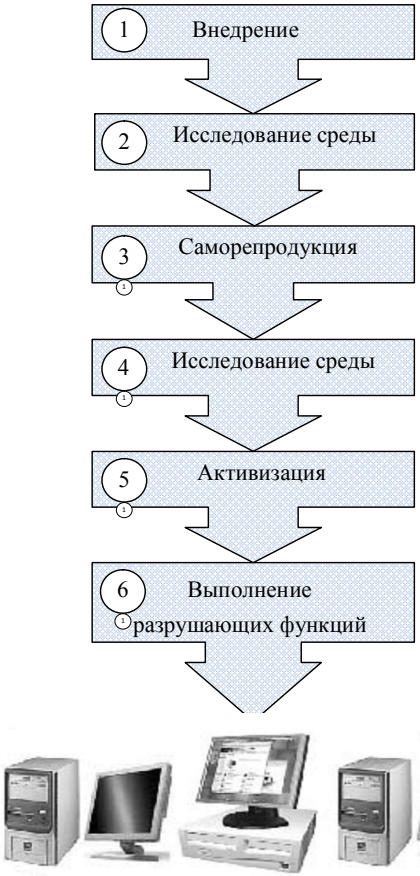

Рис 3.3. Этапы жизненного цикла программной закладки

Выполнению разрушающей функции предшествуют процессы исследования и активизации. Процесс исследования заключается в определении наиболее уязвимых мест безопасности системы, установки резидентных модулей и т.д. Таким образом, программная закладка получает информацию для дальнейшего функционирования. Активизация программной закладки представляет собой непосредственный переход к разрушающей функции посредством проверки выполнения некоторого логического условия или условий в программно-аппаратной среде. Выполнение разрушающей функции - завершающий этап жизненного цикла программной закладки. Соответствующие внутренние и внешние связи программной закладки показаны на схеме. Передача управления внутри программной закладки осуществляется в одном направлении - в сторону

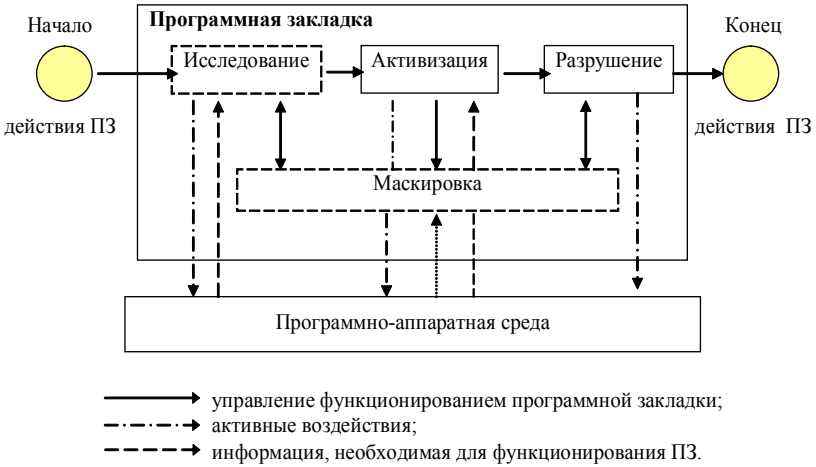

Рис. 3.4. Схема функционирования программной закладки

задействования разрушающей функции. За время своего жизненного цикла программная закладка активно воздействует на программно-аппаратную среду, как результат проявления разрушающей функции, так и с целью исследования окружающей программно-аппаратной среды. Маскировка необходима для сокрытия присутствия закладки. Процесс маскировки может быть начат в любой момент функционирования программной закладки. Особенностью данного процесса является наличие защиты от исследования закладки вследствие активного воздействия программно-аппаратной среды на область программной закладки. *Методы* защиты от закладок основаны на семантическом анализе программного обеспечения - носителя закладки**.**

1. Метод поиска программных закладок по сигнатурам. Его использование подразумевает применение побитного сравнения программ и наборов данных с сигнатурами (наборами двоичных кодов), однозначно идентифицирующими ту или иную из уже известных вредоносных программ.

Достоинство данного метода заключатся в гарантированности результатов в отношении известных вредоносных программ вне зависимости от времени их внедрения в систему и динамики изменения контролируемых программ и наборов данных. К недостаткам можно отнести необходимость постоянного обновления и пополнения набора сигнатур, а также неспособность определять новые виды вредоносных программ.

Как расширение метода поиска вредоносного программного обеспечения по сигнатурам можно рассмотреть *метод эвристического ана-*

лиза, который позволяет опять же путем сравнения выявить в кодах программ комбинации, характерные для вредоносных программ и предупредить о возможно внедренной закладке или вирусе. Этот метод позволяет производить поиск ранее неизвестного вредоносного программного обеспечения, но не позволяет принять однозначное решение о его присутствии в системе.

Кроме того, к недостаткам вышеизложенных методов можно добавить ещё один. Они могут не дать ожидаемых результатов при внедрённых вредоносных программах, способных навязывать конечный результат проверок или модифицировать свой код.

2. Метод экспериментов заключается в проведении многократных экспериментов с изучаемой программой и сравнительном анализе полученных результатов. Изучаемая программа рассматривается как «чёрный ящик», алгоритм работы которого восстанавливается путем подбора входных данных и анализа выходных. Эффективность метода экспериментов слабо зависит от программной реализации системы и определяется в первую очередь сложностью анализируемых алгоритмов. Метод экспериментов эффективен при анализе программ, реализующих относительно простые алгоритмы. Метод экспериментов редко применяется в чистом виде. Чаще он служит дополнением к динамическому или статическому методу. Это обусловлено тем, что, как правило, восстанавливаемые алгоритмы оказываются слишком сложными для данного метода.

3. *Статический метод* заключается в переводе двоичных кодов программ на язык, понятный аналитику. Как правило, в качестве такого языка выступает язык assembler, а основу для такого перевода составляют программы дизассемблирования (дизассемблеры). Дальнейшая работа после дизассемблирования сводится к анализу полученных листингов. К достоинствам данного метода можно отнести возможность восстановления алгоритма работы практически любого программного обеспечения, а к недостаткам – высокую трудоемкость, вызванную необходимостью анализа листингов дизассемблированных программ, как правило, имеющих большой объём. Поэтому, метод применим, в основном, для анализа небольших программ.

4. *Динамический метод*. Предполагает использование для выявления алгоритмов работы программы специальных программных средств, называемых отладчиками (debugger), позволяющих наблюдать за ходом выполнения, загруженной в оперативную память программы. При этом возможно выполнение программы по шагам, останов выполняемой программы в заранее обозначенных точках и просмотр фактически любой информации о состоянии системы.

Перечисленные методы семантического анализа программного кода позволяют выявлять не только ранее неизвестные вредоносные участки

кода, но и различного рода ошибки в самом программном обеспечении, т.е. устранить предпосылки вредоносного программного воздействия. С другой стороны, реализация методов защиты от закладок требует длительного времени, больших трудозатрат и высококвалифицированного персонала.

Последовательность выявления закладок в программно-аппаратной среде в общем виде может быть представлена в виде следующей последовательности шагов.

1. Выделяется группа прерываний, существенных с точки зрения обработки информации программой, относительно которой проводится защита. Обычно это прерывания int 13h, int 40h (запись и чтение информации на внешние накопители прямого доступа), int 14h (обмен с RS232 портом), int 10h (обслуживание видеотерминала), а также в обязательном порядке прерывания таймера int 8h, int 1Ch и прерывания клавиатуры int 9h и int 16h.

2. Для выделенной группы прерываний определяются точки входа (адреса входа) в ПЗУ, используя справочную информацию либо выполняя прерывание в режиме трассировки.

3. Для выделенных адресов создаются цепочки исполняемых команд от точки входа до команды IRET - возврату управления из BIOS.

Надо отметить, что запись в сегмент BIOS невозможна, и поэтому закладки в BIOS не могут применять механизм преобразования своего кода во время его исполнения в качестве защиты от изучения.

В цепочках исполняемых команд выделяются:

- команды работы с портами;
- команды передачи управления;
- команды пересылки данных.

Они используются либо для информативного анализа, либо порождают новые цепочки исполняемых команд.

Порождение новых цепочек исполняемых команд происходит тогда, когда управление передается внутри сегмента BIOS.

4. В цепочках анализируются команды выделенных групп.

Определяются:

- *опасные действия первой группы*: в прерываниях какого-либо класса присутствуют команды работы с недокументированными портами.

Наличие таких команд, как правило, указывает на передачу информации некоторому устройству, подключенному к параллельному интерфейсу (общей шине), например, встроенной радиопередающей закладке.

Данная ситуация имела место при закупке одной из партий персональных ЭВМ, где были обнаружены радиомаяки, посылавшие сигнал при выполнении программ тестирования и начальной загрузки в BIOS.

- *опасные действия второй группы*: в прерываниях какого-либо клас-

са присутствуют команды работы с портами, участвующие в работе другого класса прерываний;

- *опасные действия третьей группы*: в цепочках присутствуют команды перемещения данных из BIOS в оперативную память (кроме таблицы прерываний и RAM BIOS);

- *опасные действия четвертой группы*: в цепочках присутствуют команды передачи управления в оперативную память или в сегменты расширенного BIOS.

В случае если опасных действий не обнаружено, аппаратно-программная среда ПЭВМ без загруженной операционной среды считается безопасной.

Для проверки операционной системы используется аналогичный алгоритм:

1. По таблице прерываний определяются адреса входа для существенно важных прерываний.

2. Данные прерывания выполняются покомандно в режиме трассировки с анализом каждой команды по вышеприведенному алгоритму. В этом случае команды типа JMP не анализируются, поскольку в режиме покомандного выполнения переходы происходят автоматически. Выполнение происходит до того момента, когда будет достигнут адрес ПЗУ. Для полного анализа необходимо выполнить все используемые программой функции исследуемого прерывания.

Выводы

Атака - это совокупность действий злоумышленника, приводящих к нарушению информационной безопасности компьютерной сети (КС). В общем случае любая атака может быть разделена на четыре стадии: рекогносцировка, вторжение, атакующее воздействие, развитие.

Все угрозы ресурсам корпоративной сети могут быть классифицированы по степени риска на следующие: отказ в обслуживании, попытка несанкционированного доступа, предварительное зондирование, “подозрительная” сетевая активность. Перечисленные классы угроз упорядочены по убыванию степени риска.

При рассмотрении воздействия закладки и программ защиты информации уместны аналогии с взаимодействием вируса и прикладной программы.

Вирус может присоединиться к исполняемому файлу, соответствующим образом изменив его, может уничтожить некоторые файлы или встроиться в цепочку драйверов.

Закладка отличается более направленным и тонким воздействием. И вирус, и закладка должны скрывать свое присутствие в операционной

среде компьютерной системы.Особенностью закладок может быть и то, что они фактически становятся неотделимы от прикладных или системных программ, если внедрены в них на стадии разработки или путём обратного проектирования (путём дисассемблирования прикладной программы, внедрения кода закладки и последующей компиляции).

Структурно программная закладка состоит из четырех основных функциональных блоков: исследование, активизация, проявление деструктивных действий (разрушения) и маскировка.

Методы защиты от закладок основаны на семантическом анализе программного обеспечения - носителя закладки**.**

4. ПОДТВЕРЖДЕНИЕ ПОДЛИННОСТИ ПОЛЬЗОВАТЕЛЕЙ И РАЗГРАНИЧЕНИЕ ИХ ДОСТУПА К РЕСУРСАМ КОРПОРАТИВНОЙ СЕТИ

4.1. Основные этапы допуска в корпоративную информационную систему

Системой защиты по отношению к любому пользователю должны быть предусмотрены следующие этапы допуска в корпоративную информационную систему (систему корпоративного доступа):

1. Идентификация;

2. Установление подлинности (аутентификация);

3. Определение полномочий для последующего контроля и разграничения доступа к корпоративным (разделяемым) ресурсам.

Данные этапы должны выполняться и при подключении к корпоративной системе таких устройств, как удаленные рабочие станции и терминалы.

Идентификация необходима для указания корпоративной информационной системе уникального идентификатора обращающегося к ней пользователя с целью выполнения следующих защитных функций:

- установление подлинности и определение полномочий пользователя при его допуске в систему;

- контроль установленных полномочий и регистрация заданных действий пользователя в процессе его сеанса работы после допуска данного пользователя в корпоративную информационную систему;

- учёт обращений к корпоративной информационной системе.

Сам идентификатор может представлять собой последовательность любых символов и должен быть заранее зарегистрирован в системе администратором службы безопасности. В процессе регистрации администратором в базу эталонных данных системы защиты для каждого пользователя заносятся следующие элементы данных:

- фамилия, имя, отчество и, при необходимости, другие характеристики пользователя;

- уникальный идентификатор пользователя;

- имя процедуры установления подлинности;

- используемая для подтверждения подлинности эталонная информация, например, пароль;

- ограничения на используемую эталонную информацию, например, минимальное и максимальное время, в течение которого указанный пароль будет считаться действительным;

- полномочия пользователя по доступу к корпоративным ресурсам.

Процесс установления подлинности, называемый ещё аутентифика-

цией, заключается в проверке, является ли пользователь, пытающийся осуществить доступ к корпоративным ресурсам, тем, за кого себя выдает.

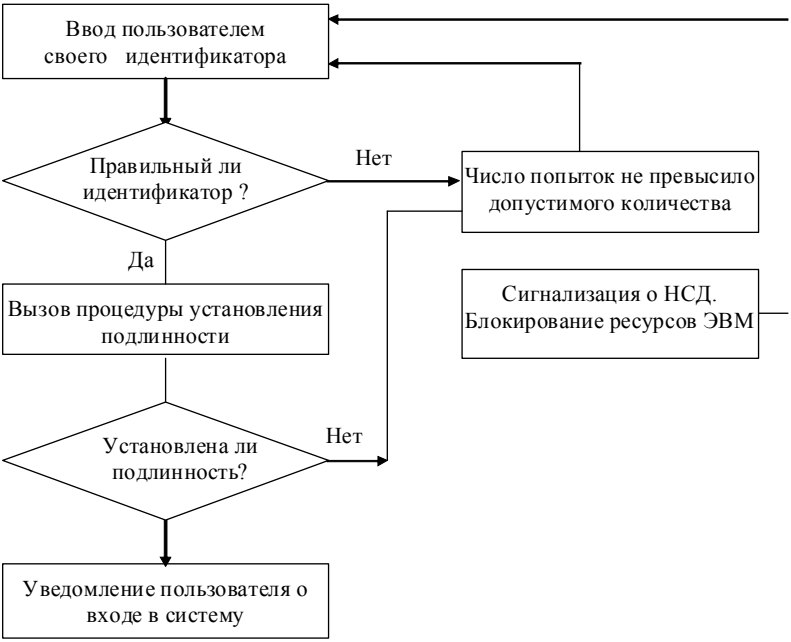

Рис. 4.1. Схема идентификации и аутентификации пользователя при его доступе в корпоративную вычислительную систему

Общая схема идентификации и установления подлинности пользователя при его доступе в компьютерную систему представлена на рис. 4.1.

Если в процессе аутентификации подлинность пользователя установлена, то система защиты должна определить его полномочия по использованию корпоративных ресурсов для последующего контроля установленных полномочий.

Основными и наиболее часто применяемыми методами установления подлинности пользователей являются методы, основанные на использовании паролей. Под *паролем* понимается некоторая последовательность символов, сохраняемая в секрете и предъявляемая при обращении к компьютерной системе. Ввод пароля, как правило, выполняется с клавиатуры после соответствующего запроса системы.

Для особо надежного опознавания могут применяться и методы, основанные на использовании технических средств определения сугубо индивидуальных характеристик человека (голоса, отпечатков пальцев,

структуры зрачка и т.д.). Однако такие средства требуют значительных затрат и поэтому используются редко.

Существующие парольные методы проверки подлинности пользователей при входе в корпоративную информационную систему можно разделить на две группы:

- методы проверки подлинности на основе простого пароля;
- методы проверки подлинности на основе динамически изменяющегося пароля.

Пароль подтверждения подлинности пользователя при использовании простого пароля не изменяется от сеанса к сеансу в течение установленного администратором службы безопасности времени его существования (действительности).

При использовании динамически изменяющегося пароля пароль пользователя для каждого нового сеанса работы или нового периода действия одного пароля изменяется по правилам, зависящим от используемого метода.

4.2. Использование простого пароля

Процедура опознавания с использованием простого пароля может выть представлена в виде следующей последовательности действий:

1) пользователь посылает запрос на доступ к компьютерной системе и вводит свой идентификатор;

2) система запрашивает пароль;

3) пользователь вводит пароль;

4) система сравнивает полученный пароль с паролем пользователя, хранящимся в базе эталонных данных системы защиты, и разрешает доступ, если пароли совпадают; в противном случае пользователь к ресурсам компьютерной системы не допускается.

Поскольку пользователь может допустить ошибку при вводе пароля, то системой должно быть предусмотрено допустимое количество повторений для ввода пароля. В базе эталонных данных системы защиты пароли, как и другую информацию, никогда не следует хранить в явной форме, а только зашифрованными. При работе с паролями должна предусматриваться и такая мера, как недопустимость их распечатки или вывода на экраны мониторов. Поэтому система защиты должна обеспечивать ввод пользователями запрошенных у них паролей без отображения этих паролей на мониторах.

Можно выделить следующие основные способы повышения стойкости системы защиты на этапе аутентификации:

- повышение степени нетривиальности пароля;
- увеличение длины последовательности символов пароля;

- увеличение времени задержки между разрешенными попытками повторного ввода неправильно введенного пароля;

- повышение ограничений на минимальное и максимальное время действительности пароля.

Чем нетривиальнее пароль, тем сложнее его запомнить. Плохо запоминаемый пароль может быть записан на листе бумаги, что повышает риск его раскрытия. Выходом здесь является использование определённого числа не записываемых на бумаге пробелов или других символов в начале, внутри, а также в конце последовательности основных символов пароля. Кроме того, отдельные символы пароля могут набираться на другом регистре (например, вместо строчных быть прописными или наоборот), что также не должно отражаться на листе бумаги. В этом случае незаконно полученный лист бумаги с основными символами пароля не будет являться достаточным условием раскрытия пароля целиком.

Вероятность подбора пароля уменьшается также при увеличении его длины и времени задержки между разрешенными попытками повторного ввода неправильно введенного пароля.

Ожидаемое время раскрытия пароля t_p можно вычислить на основе следующей полученной экспериментально приближённой формуле:

$$t_p = \left(A^k \cdot t_в\right) \Big/ 2,$$

где A - число символов в алфавите, используемом для набора символов пароля; k - длина пароля в символах, включая пробелы и другие служебные символы; $t_в$ - время ввода пароля с учётом времени задержки между разрешёнными попытками повторного ввода неправильно введенного пароля. Например, если A=26 символов (учтены только буквы английского алфавита), $t_в$= 2 секунды, а k = 6 символов, то ожидаемое время раскрытия t_p приблизительно равно одному году. Если в данном примере после каждой неудачной попытки ввода пароля предусмотреть временную задержку в 10 с., то ожидаемое время раскрытия увеличится в 5 раз.

Из приведенной выше формулы становится понятно, что повышения стойкости системы защиты на этапе аутентификации можно достигнуть и увеличением числа символов алфавита, используемого для набора символов пароля. Такое увеличение можно обеспечить путём использования нескольких регистров (режимов ввода) клавиатуры для набора символов пароля, например, путем использования строчных и прописных символов кириллицы.

Для исключения необходимости запоминания пользователями длинных и нетривиальных паролей в системе защиты может быть предусмотрена возможность записи паролей в зашифрованном виде на информационные носители, например, магнитные карты, носители данных в микросхемах и т.д., а также считывания паролей с этих информационных

носителей. Такая возможность позволяет повысить безопасность за счёт значительного увеличения длины паролей, записываемых на носители информации. Однако при этом администрации службы безопасности следует приложить максимум усилий для разъяснения пользователям вычислительной системы о необходимости тщательной сохранности носителей информации с их паролями.

На степень информационной безопасности при использовании простого парольного метода проверки подлинности пользователей большое влияние оказывают ограничения на минимальное и максимальное время действительности каждого пароля. Чем чаще меняется пароль, тем обеспечивается большая безопасность.

Минимальное время действительности пароля задает время, в течение которого пароль менять нельзя, а максимальное - время, по истечении которого пароль будет недействительным. Соответственно, пароль должен быть заменён в промежутке между минимальным и максимальным временем его существования. Поэтому понятно, что более частая смена пароля обеспечивается при уменьшении минимального и максимального времени его действительности.

4.3. Использование динамически изменяющегося пароля

Методы проверки подлинности на основе динамически изменяющегося пароля обеспечивают большую безопасность, так как частота смены паролей и них максимальна - пароль для каждого пользователя меняется ежедневно или через несколько дней. При этом каждый следующий пароль по отношению к предыдущему изменяется по правилам, зависящим от используемого метода проверки подлинности.

Существуют следующие методы парольной защиты, основанные на использовании динамически изменяющегося пароля:

- методы модификации схемы простых паролей;
- метод «запрос-ответ»;
- функциональные методы.

Наиболее эффективными из данных методов являются функциональные методы.

Методы модификации схемы простых паролей

К методам модификации схемы простых паролей относят случайную выборку символов пароля и одноразовое использование паролей.

При использовании первого метода каждому пользователю выделяется достаточно длинный пароль, причем каждый раз для опознавания используется не весь пароль, а только его некоторая часть. В процессе про-

верки подлинности система запрашивает у пользователя группу символов по заданным порядковым номерам. Количество символов и их порядковые номера для запроса определяются с помощью датчика псевдослучайных чисел.

При одноразовом использовании паролей каждому пользователю выделяется список паролей. В процессе запроса номер пароля, который необходимо ввести, выбирается последовательно по списку или по схеме случайной выборки.

Недостатком методов модификации схемы простых паролей является необходимость запоминания пользователями длинных паролей или их списков. Запись же паролей на бумагу или в записные книжки приводит к появлению риска потери или хищения носителей информации с записанными на них паролями.

Метод «запрос-ответ»

При использовании метода «запрос-ответ» в сети заблаговременно создается и особо защищается массив вопросов, включающий в себя как вопросы общего характера, так и персональные вопросы, относящиеся к конкретному пользователю, например, вопросы, касающиеся известных только пользователю случаев из его жизни. Для подтверждения подлинности пользователя система последовательно задает ему ряд случайно выбранных вопросов, на которые он должен дать ответ. Опознание считается положительным, если пользователь правильно ответил на все вопросы. Основным требованием к вопросам в данном методе аутентификации является уникальность, подразумевающая, что правильные ответы на вопросы знают только пользователи, для которых эти вопросы предназначены.

Функциональные методы

Среди функциональных методов наиболее распространенными являются метод функционального преобразования пароля, а также метод «рукопожатия».

Метод функционального преобразования основан на использовании некоторой функции F, которая должна удовлетворять следующим требованиям:

- для заданного числа или слова X легко вычислить $Y=F(X)$;
- зная X и Y, сложно или невозможно определить функцию $Y=F(X)$.

Необходимым условием выполнения данных требований является наличие в функции $F(X)$ динамически изменяющихся параметров, например, текущих даты, времени, номера дня недели, или возраста пользователя.

Пользователю сообщается:

- исходный пароль - слово или число X, например число 31;

- функция $F(X)$, например, $Y=(X \bmod 100)D+W^3$, где $(X \bmod 100)$ - операция взятия остатка от целочисленного деления X на 100, D -текущий номер дня недели, а W - текущий номер недели в текущем месяце);

- периодичность смены пароля, например, каждый день, каждые три дня или каждую неделю.

Паролями пользователя для последовательности установленных периодов действия одного пароля будут соответственно X, $F(X)$, $F(F(X))$, $F(F(F(X)))$ и т. д., т.е. для 1-го периода действия одного пароля паролем пользователя будет $F(X)$. Поэтому для того чтобы вычислить очередной пароль по истечении периода действия используемого пароля, пользователю не нужно помнить начальный (исходный) пароль, важно лишь не забыть функцию парольного преобразования и пароль, используемый до настоящего момента времени.

С целью достижения высокого уровня безопасности функция преобразования пароля, задаваемая для каждого пользователя, должна периодически меняться, например, каждый месяц. При замене функции целесообразно устанавливать и новый исходный пароль.

Согласно *методу «рукопожатия»* существует функция F, известная только пользователю и самой системе, доступ к которой он хочет получить. Данная функция должна удовлетворять тем же требованиям, которые определены для функции, используемой в методе функционального преобразования.

При входе пользователя в вычислительную систему системой защиты генерируется случайное число или случайная последовательность символов X и вычисляется функция $F(X)$, заданная для данного пользователя (рис. 4.2). Далее X выводится пользователю, который должен

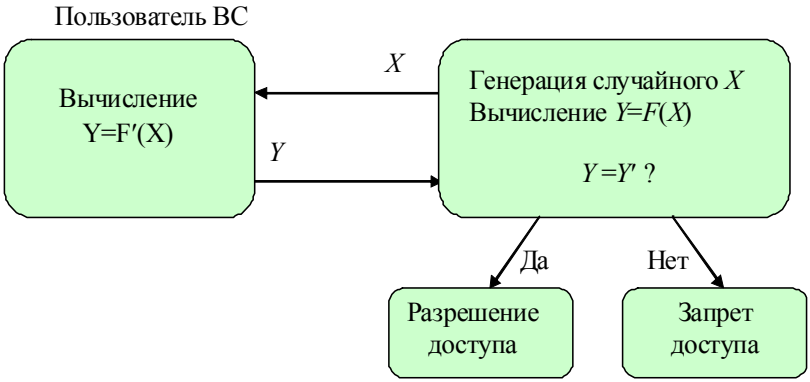

Рис. 4.2. Схема аутентификации по методу «рукопожатия»

вычислить $F'(X)$ и ввести полученное значение в систему. Значения $F(X)$ и $F'(X)$ сравниваются системой и, если они совпадают, то пользователь получает доступ в ВС.

Для высокой безопасности функцию «рукопожатия» целесообразно циклически менять через определенные интервалы времени, например, устанавливать разные функции для четных и нечётных чисел месяца.

Достоинством метода «рукопожатия» является то, что никакой конфиденциальной информации между пользователем и ВС не передается. По этой причине эффективность данного метода особенно велика при его применении в вычислительных сетях для подтверждения подлинности пользователей, пытающихся осуществить доступ к серверам или центральным ЭВМ.

В некоторых случаях может оказаться необходимым пользователю проверить подлинность той вычислительной системы, к которой он хочет осуществить доступ. Необходимость во взаимной проверке может понадобиться и когда два пользователя ВС хотят связаться друг с другом по линии связи. Методы простых паролей, а также методы модификации схем простых паролей в этом случае не подходят. Наиболее подходящим здесь является метод «рукопожатия». При его использовании ни один из участников сеанса связи не будет получать никакой секретной информации.

Приведём пример, доказывающий данное утверждение. Типовая структура корпоративной сети, приведена на рис. 4.3. На ней изображены:

- вспомогательный (Proxy) сервер, основными функциями которого является коммутация трафика между интерфейсами Int-1, Int-2 и Int-3 в соответствии со списками доступа администратора;
- узлы локального и удаленного пользователя;
- серверы:

DHCP для конфигурирования хостов,

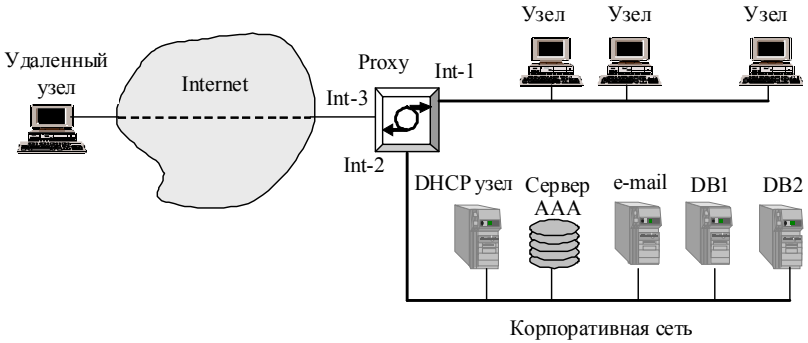

Рис. 4.3. Типовая схема корпоративной сети

AAA для аутентификации, авторизации и учета,

e-mail для обработки почтовых сообщений,

DB1и DB2 для хранения документации группового использования.

При попытке подключения пользователя к корпоративной сети proxy-сервер запрашивает его имя и пароль. Полученный ответ сравнивается с записью в списке доступа вида [имя пользователя (*Name* или *User ID*) – пароль (P*assword*)], которая внесена администратором сети и хранится на AAA-сервере.

Proxy-сервер посылает удаленному узлу пользователя некоторое случайное число *V*, а хост возвращает другое число *W*, вычисленное по заранее известной функции с использованием имени (*Name*) и пароля (*Password*). Иначе говоря, *W=F*(*V*, *Name*, *Password*). Предполагается, что злоумышленник в состоянии перехватить пересылаемые по сети значения *V*, *Name* и *W* и ему известен алгоритм вычисления функции *F*. Существо формирования *W* состоит в том, что исходные элементы (биты) случайного числа *V* различным образом «перемешиваются» с неизвестным злоумышленнику элементами пароля *Password*. Затем полученный зашифрованный текст подвергается сжатию. Такое преобразование называется дайджест-функцией (digest function) или хэш-функцией, а результат – дайджестом. Точная процедура формирования дайджеста определена алгоритмом MD5 и описана в RFC 1321, PS. Proxy-сервер запрашивает у AAA-сервера истинное значение *W*, пересылая ему значения Name и *V*. Сервер AAA на основании полученных от proxy-сервера значений *V* и *Name* и имеющегося у него в базе данных пароля *Password* по тому же алгоритму вычисляет *W* и возвращает его proxy-серверу. Proxy-сервер сравнивает два значения *W*, полученные от хоста и от AAA-сервера: если они совпадают, то хосту посылается сообщение об успешной аутентификации.

После успешной аутентификации пользователя proxy-сервер на основании списка управления доступом производит авторизацию, т.е. определяет, к каким серверам DB1 и DB2 группового использования может обращаться пользователь, а серверы DB1 и DB2 определяют, какие операции (только чтение или чтение/запись) он может осуществлять.

Для последующего возможного анализа успешных и неуспешных соединений пользователей выполняется процедура учёта, которая состоит в ведении записей истории соединений пользователей.

На рис. 4.4 приведена процедура аутентификации пользователя со следующими исходными данными: имя пользователя (*Name*) Ivanov, пароль (*Password* = K1m), случайное число (*V*) 123456. Процедура перемешивания состоит в последовательном перемешивании полубайтов пароля и случайного числа. Вычисление дайджеста состоит в вычислении остатка перемешенного числа по модулю *Password*.

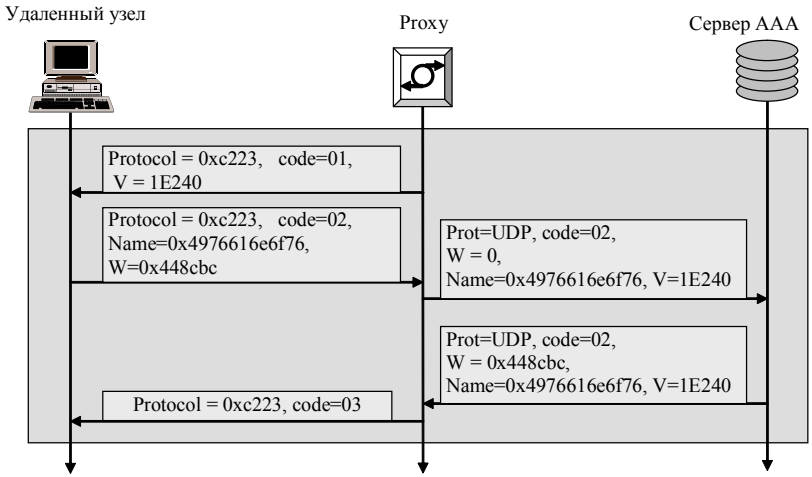

Рис. 4.4. Процедура аутентификации пользователя

В первом сообщении proxy-сервер запрашивает (code=01) по протоколу аутентификации CHAP (Protocol = 0xc223) у удаленного пользователя ответ на случайное число V = 123456 = 0x1E240. Хост удаленного пользователя производит следующие операции:

1) подставляет имя пользователя, используя таблицу кодов ASCII (табл.4.1); Для определения двоичного кода символа следует к коду колонки приписать код строки, а для определения шестнадцатеричного – к

Таблица 4.1

№	(0) 000	(1) 001	(2) 010	(3) 011	(4) 100	(5) 101	(6)110	(7) 111
(0) 0000	**NUL**	**DLE**	**SP**	**0**	**@**	**P**	**'**	**p**
(1) 0001	**SOH**	**DC1**	**!**	**1**	**A**	**Q**	**a**	**q**
(2) 0010	**STX**	**DC2**	**“**	**2**	**B**	**R**	**b**	**r**
(3) 0011	**ETX**	**DC3**	**#**	**3**	**C**	**S**	**c**	**s**
(4) 0100	**EOT**	**DC4**	**$**	**4**	**D**	**T**	**d**	**t**
(5) 0101	**ENQ**	**NAK**	**%**	**5**	**E**	**U**	**e**	**u**
(6) 0110	**ACK**	**SYN**	**&**	**6**	**F**	**V**	**f**	**v**
(7) 0111	**BEL**	**ETB**	**'**	**7**	**G**	**W**	**g**	**w**
(8) 1000	**BS**	**CAN**	**(**	**8**	**H**	**X**	**h**	**x**
(9) 1001	**HT**	**EM**	**)**	**9**	**I**	**Y**	**i**	**y**
(a) 1010	**LF**	**SUB**	*****	**:**	**J**	**Z**	**j**	**z**
(b) 1011	**VT**	**ESC**	**+**	**;**	**K**	**[**	**k**	**{**
(c) 1100	**FF**	**IS4**	**,**	**<**	**L**	****	**l**	**|**
(d) 1101	**CR**	**IS3**	**-**	**=**	**M**	**]**	**m**	**}**
(e) 1110	**SO**	**IS2**	**.**	**>**	**N**	**^**	**n**	**~**
(f) 1111	**S1**	**IS1**	**/**	**?**	**O**	**_**	**o**	**DEL**

значению кода колонки приписать значение кода строки. В соответствии с табл.4.1 имя пользователя Ivanov представляется как 0x4976616e6f76, а пароль K1m – как 0x4b316d.

2) перемешивает байты пароля 0x**4b316d** и случайного числа 0x*01e240*, получая перемешанное число *F*=0x**4***0***b***13***e***1***2***6***4***d***0*;

3) вычисляет ответ как $W = F \bmod Password = 40b13e1264d0 \bmod 0x4b316d = 71129994781904 \bmod 4927853 = 4493476 = 0x448cbc$. Во втором сообщении хост возвращает ответ в следующем виде *Name* = 0x4976616e6 f76 и *W* = 0x448cbc. В третьем сообщении proxy-сервер запрашивает истинное значение *W* у AAA-сервера, посылая ему те же значения *Name* и *V*. В четвёртом сообщении proxy-сервер получает от AAA-сервера истинное значение *W*, соответствующее *Name*=0x4976616e6f76 и *V*=0x1E240. В пятом сообщении proxy-сервер подтверждает (code=03) легитимность пользователя.

4.4. Протоколы установления подлинности

Протоколы установления подлинности (аутентификации) позволяют процессу убедиться, что он взаимодействует с тем, кто должен быть, а не с тем, кто лишь представляется таковым.

Очень часто путают проверку прав на выполнение тех или иных операций с аутентификацией. (В первом случае имеем авторизацию.) Аутентификация отвечает на вопрос: как убедиться, что вы взаимодействуете именно с определенным процессом. Если, например, к серверу процесс обратился с запросом удалить файл x.old и объявил себя процессом Вася, то сервер должен убедиться, что перед ним действительно Вася и что Вася имеет право делать то, что просит. Ключевым, конечно, является первый вопрос, ответ на второй вопрос – это дело просмотра таблицы прав пользователей.

Общая схема всех протоколов аутентификации такова: сторона *A* и сторона *B* начинают обмениваться сообщениями между собой или с Центром раздачи ключей (ЦРК). ЦРК всегда надежный партнер. Протокол аутентификации должен быть устроен так, что даже если злоумышленник перехватит сообщения между *A* и *B*, то ни *A*, ни *B* не спутают друг друга с злоумышленником. Обмен данными между *A* и *B* будет происходить по алгоритму с закрытым ключом, а вот устанавливаться соединение по алгоритму с открытым ключом.

Аутентификация на основе закрытого разделяемого ключа

Основная идея первого протокола аутентификации, так называемого протокола «ответ по вызову», состоит в том, что одна сторона посылает некоторое число (вызов), другая сторона, получив это число, преобразует

его по определенному алгоритму и отправляет обратно. Посмотрев на результат преобразования и зная исходное число, инициатор может судить, правильно ли сделано преобразование или нет. Алгоритм преобразования является общим секретом взаимодействующих сторон.

Будем предполагать, что стороны A и B имеют общий секретный ключ K_{AB}. Этот секретный ключ взаимодействующие стороны как-то установили заранее, например, по телефону. Описанная выше процедура показана на рис.4.5.

A,B - идентификаторы взаимодействующих сторон; R_i - вызов, где индекс указывает кто его послал; K_i - ключ, индекс которого указывает на его владельца.

На рис. 4.6 дана схема, где сокращено количество передач между сторонами, по сравнению с рис. 4.5, а рис. 4.7 показывает «дыру» в схеме 4.6 и как злоумышленник может этой дырой воспользоваться. Это, так называемая, *атака отражением*. Есть несколько общих правил построения протоколов аутентификации (протокол проверки подлинности или просто подлинности):

1) инициатор должен доказать кто он есть прежде, чем вы пошлёте ему какую-то важную информацию;

2) инициатор и отвечающий должны использовать разные ключи;

3) инициатор и отвечающий должны использовать начальные вызовы из разных непересекающихся множеств.

В схеме на рис.4.6 все эти три правила нарушены.

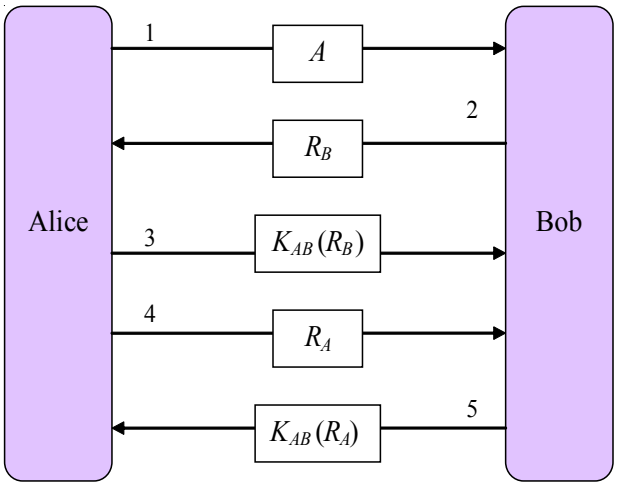

Рис. 4.5. Протокол аутентификации «Ответ по вызову»

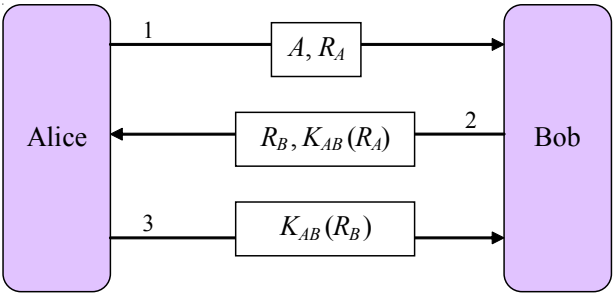

Рис. 4.6. Сокращение количества передач между сторонами в протколе аутентификации

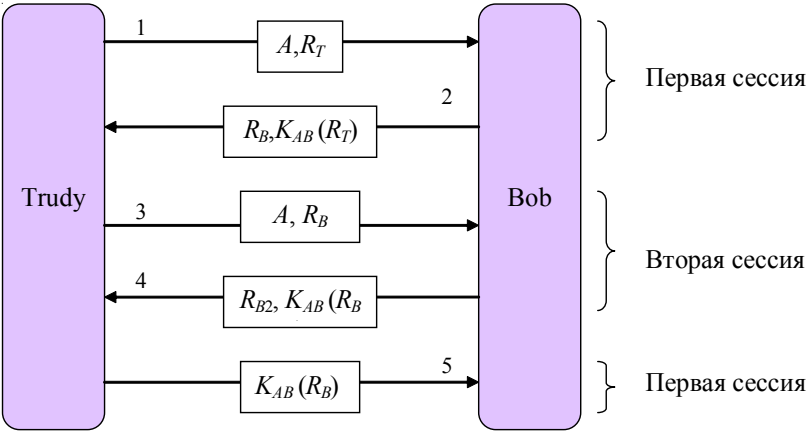

Рис. 4.7. Атака отражением

Установка разделяемого ключа

До сих пор мы предполагали, что A и B имеют общий секретный ключ. Рассмотрим теперь, как они могут его установить? Например, они могут воспользоваться телефоном. Однако, как B убедиться, что ему звонит именно A, а не злоумышленник? Можно договориться о личной встрече, куда принести паспорт и прочее, удостоверяющее личность. Однако есть протокол, который позволяет двум незнакомым людям установить общий ключ даже при условии, что за ними следит злоумышленник. Это протокол обмена ключом Диффи-Хеллмана. Его схема показана на рис.4.8.

Прежде всего A и B должны договориться об использовании двух больших простых чисел n и g, удовлетворяющих определенным условиям.

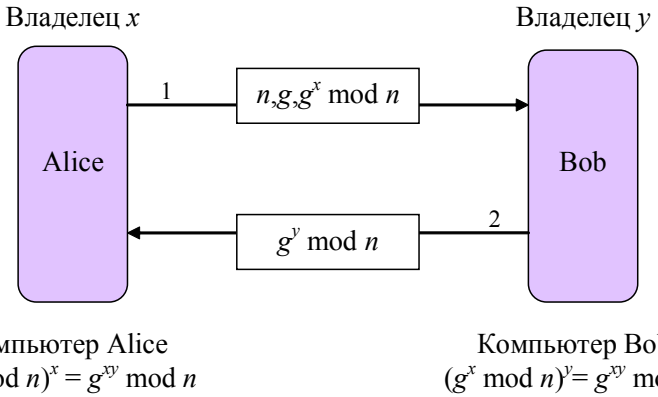

Рис. 4.8. протокол обмена ключом Диффи-Хеллмана

Эти числа могут быть общеизвестны. Затем *A* выбирает большое число, скажем *x*, и хранит его в секрете. То же самое делает *B*. Его число – *y*.

A шлет *B* сообщение ($n,g,g^x \bmod n$), *B* шлет в ответ ($g^y \bmod n$). Теперь *A* выполняет операцию $(g^y \bmod n)^x$, *B* – $(g^x \bmod n)^y$. Теперь оба имеют общий ключ – $g^{xy} \bmod n$. Например, n=47, g=3, x=8, y=10, то *A* шлет *B* сообщение (47,3,28), поскольку $3^8 \bmod 47 = 28$. *B* шлет *A* (17). *A* вычисляет $17^8 \bmod 47 = 4$, *B* вычисляет $28^{10} \bmod 47 = 4$. Ключ установлен, это – 4.

Злоумышленник следит за всем этим процессом. Единственно, что мешает ему вычислить *x* и *y* – это то, что не известен алгоритм с приемлемой сложностью для вычисления логарифма от модуля для простых чисел. Однако, у этого алгоритма есть слабое место, которое демонстрирует рис. 4.9. Такой приём называется *чужой в середине*.

Проверка подлинности через центр раздачи ключей

Договариваться с незнакомцем об общем секрете можно, но вряд ли это следует делать сразу (атака не спелого винограда). Кроме этого, общение с *n* людьми потребует хранения *n* ключей, что для общительных или популярных личностей может быть проблемой.

Другое решение можно получить, введя надежный центр распространения ключей (ЦРК). Его использование иллюстрирует рис.4.10.

Идея этого протокола состоит в следующем. *A* выбирает ключ сессии K_S. Используя свой ключ K_A, шлет в ЦРК запрос на соединение с *B*. ЦРК знает *B* и его ключ K_B. С помощью этого ключа ЦРК сообщает *B* ключ сессии K_S и кто хочет с ним с соединиться.

Однако решение с использованием ЦРК имеет изъян. Пусть злоумышленник как-то убедил *A* связаться с *B* и скопировал весь обмен сообщениями. Позже он может воспроизвести этот обмен за *A* и заставить *B*

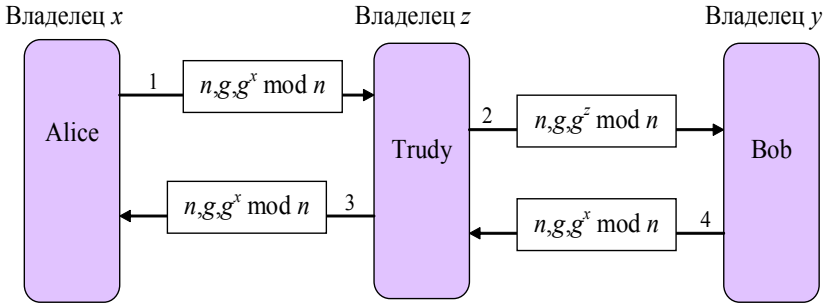

Рис. 4.9. Атака «чужой в середине»

действовать так, как если бы с *B* говорил *A*. Этот способ атаки называется *атака подменой*.

Против такой атаки есть несколько решений. Одно из них – *временные метки*. Это решение требует синхронизации часов. Поскольку в сети всегда есть расхождение в показаниях часов, то необходимо выделить определенный допуск, интервал, в течение которого считать сообщение верным. Злоумышленник может использовать приёмом атаки подменой в течение этого интервала.

Другое решение использование *разовых меток*. Однако каждая из сторон должна помнить все разовые метки, использованные ранее. Это обременительно. Кроме этого, если список использованных разовых меток будет утерян по каким-либо причинам, то весь метод перестанет работать. Можно комбинировать решения разовых меток и временных меток.

Более тонкое решение установления подлинности даёт многосторонний вызов-ответ протокол. Хорошо известным примером такого протокола является протокол Нидхема-Шредера, вариант которого показан на рис.4.11.

Вначале *A* сообщает ЦРК, что он хочет взаимодействовать с *B*. ЦРК сообщает ключ сессии, разовую метку R_A, шифруя сообщение ключом *A*. Разовая метка защищает *A* от подмены. Теперь, имея ключ сессии, *A* начинает обмен сообщениями с *B*. R_{A2} и R_B – разовые метки, защищающие *A* и *B* от подмен.

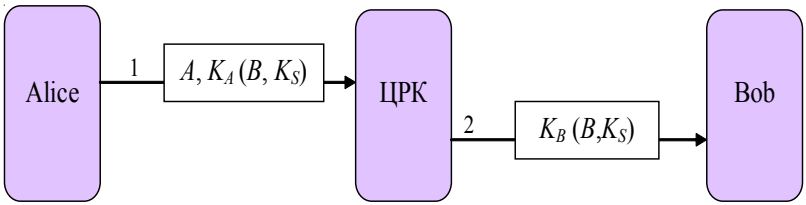

Рис. 4.10. Участие ЦРК в процессе аутентификации

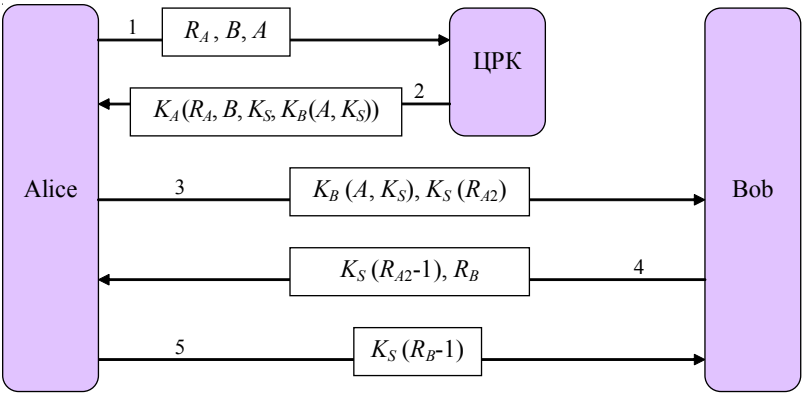

Рис. 4.11. Протокол аутентификации Нидхема-Шредера

Хотя этот протокол в целом надежен, но все-таки есть небольшая опасность. Если злоумышленник раздобудет все-таки старый ключ сессии, то он сможет подменить сообщение 3 старым и убедить *B*, что это *A*. На рис.4.12 приведена схема исправленного протокола, предложенного Отвей и Рисом. В этой модификации ЦРК следит, чтобы *R* было одним и тем же в обеих частях сообщения 2.

Установление подлинности протоколом Цербер

Протокол установления подлинности Цербер используется многими практически действующими системами. Он представляет собой вариант протокола Нидхема-Шредера и был разработан в Массачусетском технологическом университете для безопасного доступа в сеть - предот-

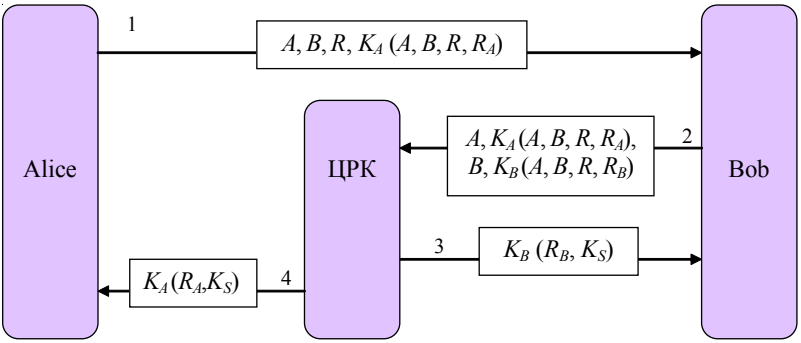

Рис. 4.12.Схема исправленного протокола, предложенного Отвей и Рисом

вратить несанкционированное использование ресурсов сети. В нём использовано предположение, что все часы в сети хорошо синхронизованы.

Протокол Цербер предполагает использование, кроме рабочей станции A еще трех серверов:

- сервер установления подлинности (СП) – проверяет пользователей на этапе login;
- сервер выдачи билета (СВБ) – идентификация билетов;
- сервер B – тот кто должен выполнить работу, необходимую A.

Сервер установления подлинности аналогичен Центру раздачи ключей и знает секретный пароль для каждого пользователя. Сервер выдачи билетов выдает билеты, которые подтверждают подлинность заказчиков работ.

На рис. 4.13 показана работа протокола Цербер. Сначала пользователь садится за рабочую станцию и шлёт открыто свое имя A серверу установления подлинности (СП). СП отвечает ключом сессии K_S и билетом $K_{\text{СВБ}}(A,K_S)$ к серверу выдачи билетов (СВБ) для предъявления этого билета на следующем шаге при обращении к СВБ. Все это зашифровано секретным ключом A. Когда сообщение 2 пришло на рабочую станцию у A запрашивают пароль, чтобы по нему установить K_A, для расшифровки сообщения 2. Пароль перезаписывается с временной меткой, чтобы предотвратить его захват злоумышленником. Выполнив login, пользователь может сообщить станции, что ему нужен сервер B. Рабочая станция обращается к СВБ за билетом для использования сервера B. Ключевым элементом этого запроса является $K_{\text{СВБ}}(A,K_S)$, зашифрованное секретным ключом СВБ. В ответ СВБ шлет ключ K_{AB} для работы A и B.

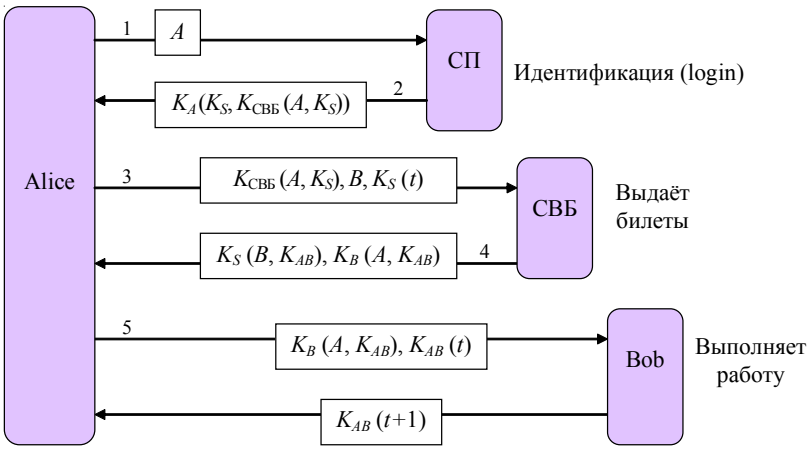

Рис. 4.13. Протокол Цербер

Теперь *A* может обращаться непосредственно к *B* с этим ключом. Это взаимодействие сопровождается временными метками, чтобы защититься от подмены. Если позднее *A* понадобиться работать с сервером *C*, то *A* должен будет повторить сообщение 3, но указать там сервер *C*.

Поскольку сеть может быть очень большой, то нельзя требовать, чтобы все использовали один и тот же СП. Сеть разбивают на области, в каждой свои СП и СВБ, которые взаимодействуют между собой.

Установление подлинности, используя шифрование с открытым ключом

Установить взаимную подлинность можно с помощью шифрования с открытым ключом. Пусть *A* и *B* уже знают открытые ключи друг друга. Они их используют, чтобы установить подлинность друг друга, а затем использовать шифрование с секретным ключом, которое на несколько порядков быстрее.

На рис. 4.14 показана схема установления подлинности с шифрованием открытыми ключами.

Здесь R_A и R_B используются, чтобы убедить *A* и *B* в их подлинности. Единственным слабым местом этого протокола является предположение, что *A* и *B* уже знают открытые ключи друг друга. Обмен такими ключами уязвим для атаки типа «чужой в середине».

Ривст и Шамир предложили протокол, защищенный от атаки «чужой в средине». Это, так называемый, протокол с внутренним замком. Его идея передавать сообщения в два этапа: сначала только чётные биты, затем нечётные.

Выводы

Основными этапами допуска в компьютерную систему являются идентификация, аутентификация и определение полномочий пользо-

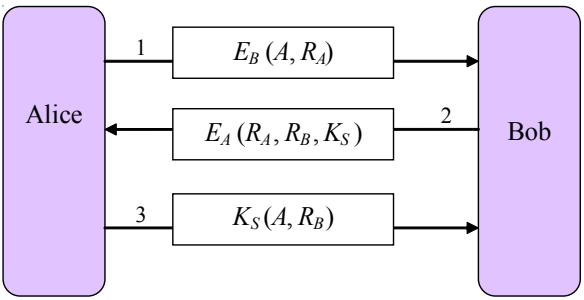

Рисунок 4.14. Схема установления подлинности с шифрованием открытыми ключами

вателя. Идентификация необходима для указания компьютерной системе уникального имени обращающегося к ней пользователя. Аутентификация заключается в проверке, является ли пользователь, пытающийся осуществить доступ к корпоративным ресурсам, тем, за кого себя выдает. Определение полномочий необходимо для последующего контроля и разграничения доступа к корпоративным ресурсам.

Основными и наиболее часто применяемыми методами аутентификации пользователей являются методы, основанные на использовании паролей. Они подразделяются на методы проверки подлинности на основе простого пароля и на основе динамически изменяющегося пароля.

Аутентификация на базе простого пароля предполагает, что пароль подтверждения подлинности пользователя не изменяется от сеанса к сеансу в течении установленного администратором службы безопасности времени его действительности.

При использовании динамически изменяющегося пароля для каждого нового сеанса работы или нового периода действия пароль изменяется по правилам, зависящим от используемого метода.

Последовательность и правила установления подлинности пользователей в корпоративной сети устанавливают протоколы аутентификации.

5. ТЕХНОЛОГИИ, МЕТОДЫ И СРЕДСТВА ЗАЩИТЫ ИНФОРМАЦИОННЫХ РЕСУРСОВ КОРПОРАТИВНОЙ СЕТИ

В качестве одного из базовых средств защиты корпоративных информационных ресурсов сегодня выступают системы обнаружения атак (СОА), позволяющие своевременно выявлять и блокировать атаки нарушителей.

Процесс выявления атак (рис. 5.1) начинается со сбора данных, необходимых для определения факта атаки на ресурсы сети. В частности, можно анализировать сведения о пакетах данных, поступающих из внешней сети в корпоративную сеть компании, производительность программно-аппаратных средств (вычислительную нагрузку на узлы сети, загруженность оперативной памяти, скорость работы прикладного программного обеспечения и др.), сведения о доступе к определенным файлам

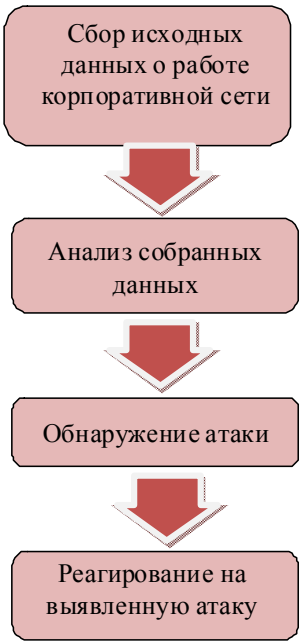

Рис. 5.1. Схема процесса обнаружения атаки

и т.д. Полезно также иметь полную информацию о регистрации пользователей при входе в корпоративную сеть.

Сбор исходной информации традиционно осуществляется с помощью специализированных датчиков СОА, размещаемых на разных эле-

ментах корпоративной сети. Существуют два типа таких датчиков: сетевые и узловые. Первые предназначены для сбора информации о пакетах данных, передаваемых в тех сегментах сети, где они установлены. Узловые датчики размещаются на определённые компьютеры и собирают информацию о событиях, возникающих на этих компьютерах (например, сведения о сетевом трафике, поступающем на узел или системных событиях, регистрируемых в журналах аудита операционной системы узла сети). При этом один узел может отслеживаться сразу несколькими узловыми датчиками, каждый из которых предназначен для сбора определенной информации.

Анализ данных, собранных сетевыми и узловыми датчиками, проводится СОА с использованием специальных методов выявления атак. Существуют две основные группы таких методов: сигнатурные и поведенческие.

Сигнатурные методы описывают каждую атаку в виде специальной модели или сигнатуры, в качестве которой могут применяться:

- строка символов;
- семантическое выражение на специальном языке;
- формальная математическая модель.

Суть сигнатурного метода в следующем: в исходных данных, собранных сетевыми и узловыми датчиками СОА, выполняется процедура поиска сигнатуры атаки с использованием специализированной базы данных сигнатур атак. Преимуществом данных методов является высокая точность определения факта атаки, а очевидным недостатком – невозможность обнаружения тех атак, сигнатуры которых пока не определены.

Поведенческие методы базируются не на моделях атак, а на моделях штатного процесса функционирования (поведения) сети. Принцип работы любого из таких методов основан на обнаружении несоответствия между текущим режимом работы сети и режимом работы, соответствующим штатной модели данного метода. Любое несоответствие рассматривается как атака. Преимущество методов данного типа - возможность обнаружения новых атак без модификаций или обновлений параметров модели. К сожалению, создать точную модель штатного режима функционирования сети очень сложно.

Для того чтобы лучше понять специфику сигнатурного и поведенческого метода выявления атак, рассмотрим их конкретные примеры, реализованные в современных СОА.

5.1. Сигнатурные методы выявления атак

Среди сигнатурных методов выявления атак наиболее распространён метод контекстного поиска, который заключается в обнаружении в

исходной информации определённого множества символов. Так, например, для выявления атаки на Web-сервер, направленной на получение несанкционированного доступа к файлу паролей, проводится поиск последовательности символов «GET */etc/passwd» в заголовке HTTP-запроса. Для расширения функциональных возможностей контекстного поиска в некоторых случаях используются специализированные языки, описывающие сигнатуру атаки.

Использование контекстного поиска позволяет эффективно выявлять атаки на основе анализа сетевого трафика, поскольку данный метод позволяет наиболее точно задать параметры сигнатуры, которую необходимо выявить в потоке исходных данных.

В ряде некоммерческих СОА были реализованы ещё два сигнатурных метода: анализа состояний и метод, базирующийся на экспертных системах.

Метод *анализа состояний* основан на формировании сигнатуры атак в виде последовательности переходов сети из одного состояния в другое. По сути, каждый такой переход определяется по наступлению в корпоративной сети определённого события, а набор таких событий задается параметрами сигнатуры атаки. Как правило, сигнатуры атак, созданные на основе анализа состояний, описываются математическими моделями, базирующимися на теории конечных автоматов или сетей Петри. На рис. 5.2 приведена сеть Петри, описывающая сигнатуру атаки, которая выполняет подбор пароля для получения несанкционированного доступа к ресурсам корпоративной сети. Каждый переход корпоративной сети в новое состояние в модели сети Петри связан с попыткой ввода пользователем пароля. Если пользователь в течение одной минуты четыре

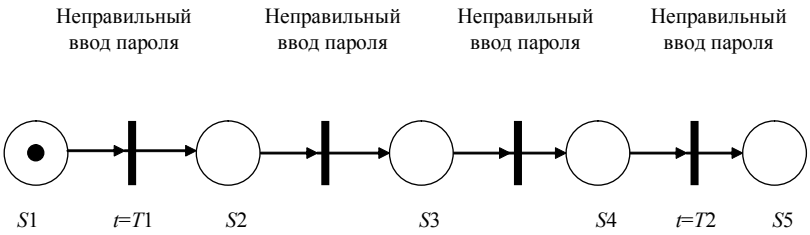

Рис. 5.2. Сеть Петри, описывающая сигнатуру атаки, осуществляющей подбор пароля

раза подряд введёт неправильный пароль, то метод зафиксирует факт осуществления атаки.

Методы выявления атак, базирующиеся на экспертных системах, позволяют описывать модели атак на естественном языке с высоким уров-

нем абстракции. Экспертная система – это система, которая в контексте обнаружения атак принимает решение о принадлежности того или иного события к классу атак на основании имеющихся правил. Эти правила (rules) основаны на опыте специалистов и хранятся в специальном хранилище, которое представляет собой базу знаний. Результирующая база знаний должна описывать характерные признаки атак, которые должна обнаруживать СОА. Исходные данные о работе корпоративной сети образуют базу данных экспертной системы и служат основанием для принятия решений о наличии атаки.

Экспертные системы нуждаются в постоянном обновлении для того, чтобы оставаться актуальными. Этот метод продемонстрировал, что он является сравнительно эффективным, если известны точные характеристики атаки. К достоинствам данного метода можно отнести простоту реализации, скорость функционирования и отсутствие ложных тревог.

Однако сетевые атаки постоянно изменяются, поскольку злоумышленники используют индивидуальные подходы, программное обеспечение и аппаратные средства регулярно совершенствуются. Поэтому даже специальные постоянные обновления базы знаний экспертной системы не способствуют точной идентификации всего диапазона атак. Таким образом, главными недостатками метода экспертных систем являются неспособность к обнаружению неизвестных атак и тот факт, что небольшие изменения в атаке приводят к невозможности её обнаружения.

Одной из наиболее перспективных сигнатурных групп являются методы, основанные на биологических моделях. Для их описания могут использоваться генетические или нейросетевые алгоритмы.

На сегодняшний день все методы, базирующиеся на биологических моделях, находятся пока в стадии исследования и коммерческого применения не имеют.

Таким образом, сигнатурный подход выявления атак сводится к обнаружению злоупотреблений, которое сводится к написанию атаки в виде шаблона (pattern) или сигнатуры (signature) и поиску данного шаблона в контролируемом пространстве. Типичными представителями, реализующими данную идею, являются антивирусные сканеры (работают с базой данных сигнатур вирусов) и системы обнаружения сетевых атак (работают с базой данных сигнатур удаленных атак). Система, построенная данным образом, может обнаруживать все известные атаки, но она мало приспособлена для обнаружения новых, ещё неизвестных атак.

Подход, реализованный в таких СОА, очень прост и именно на нём основаны практически все предлагаемые сегодня на рынке системы обнаружения атак. Однако администраторы сталкиваются с проблемами при эксплуатации этих систем. Первая проблема заключается в создании механизма описания сигнатур, т.е. языка описания атак. Вторая проблема

плавно вытекает из первой: каким образом нужно записать атаку, чтобы зафиксировать все возможные её модификации.

Схема типичной системы обнаружения атак с применением сигнатурного метода показана на рис. 5.3.

Обычно системы обнаружения атак задействуют в качестве источника данных журналы регистрации и сетевой трафик. Однако наиболее часто их применяют именно для анализа трафика.

5.2. Поведенческие методы выявления атак

Как уже отмечалось, поведенческие методы применяются для выявления атак по отклонениям от штатной работы корпоративной сети. Среди них наиболее распространены те, которые базируются на статистических моделях. Такие модели определяют статистические показатели, характеризующие параметры штатного поведения сети. Если с течением времени наблюдается определенное изменение данных параметров от заданных значений, то фиксируется факт обнаружения атаки.

Как правило, в качестве таких параметров могут выступать: уровень загрузки процессора, нагрузка на каналы связи, штатное время работы пользователей, количество обращений к сетевым ресурсам и др. Все множество параметров, которые включаются в шаблон штатного пове-дения сети, могут быть отнесены к следующим распространенным группам:

- числовые параметры (количество переданных данных по различным протоколам, загрузка центрального процессора, число файлов, к которым осуществляется доступ, и т.д.);
- категориальные параметры (имена файлов, команды пользователя, открытые порты и т.д.);
- параметры активности (число обращений к файлам или соединений за единицу времени и др.).

Очень важно правильно выбрать контролируемые параметры для системы обнаружения атак. Малое их число или неправильно отобранные параметры могут привести к тому, что модель описания поведения субъектов системы будет неполной и многие атаки останутся за пределами её рассмотрения. С другой стороны, слишком большое число параметров мониторинга вызовет снижение производительности контролируемого узла за счёт увеличенных требований к потребляемым ресурсам (оперативной и дисковой памяти, загрузке процессора и т.д.).

Примерами подобных статистических моделей могут служить пороговая модель, модели среднего значения и среднеквадратичного отклонения или её многовариационная модель.

В *пороговой модели*, как явствует из названия, для каждого статистического параметра определены пороговые величины. Если наблюдаемый

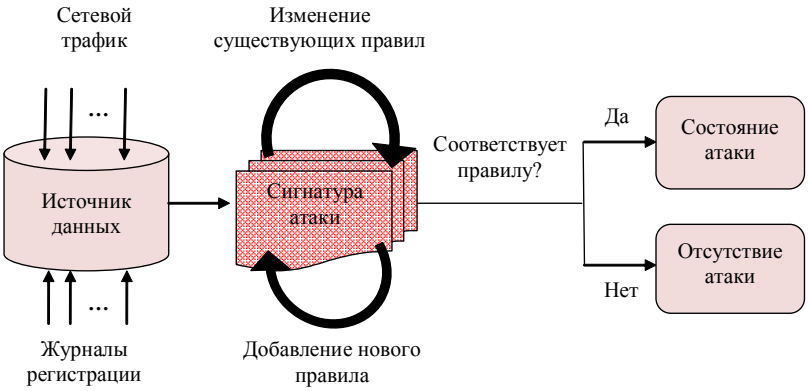

Рис. 5.4. Схема системы обнаружения атаки

параметр превышает заданный порог, то вызвавшее это событие является признаком потенциальной атаки. Например, превышение заданного количества запросов на доступ к ресурсам корпоративной сети может свидетельствовать о факте обнаружения атаки «отказ в обслуживании». Или, например, статистический анализ может помочь в обнаружении необычного события, заключающегося в том, что зарегистрированный пользователь, который никогда ранее не входил в сеть в не рабочее время (например, от 6 часов вечера до 8 часов утра), вдруг подключился к системе в 2 часа ночи.

И хотя пороговая модель достаточно эффективна и надежна для некоторых типов атак, широкого распространения в настоящее время она не получили из-за своих недостатков. Один из основных недостатков – это трудность задания порогового значения. Слишком большое пороговое значение приведет к тому, что многие атаки не будут обнаружены, а чересчур малое – обусловит большое число ложных срабатываний. Выбор этих значений – очень нетривиальная задача, которая требует глубоких знаний о работе контролируемой корпоративной сети.

Модели *среднего значения* и *среднеквадратичного отклонения* для каждого статистического параматра на основе математического ожидания и дисперсии определяет доверительный интервал, в пределах которого должен находится данный параметр. Если текущее значение параметра выходит за его границы, то фиксируется осуществление атаки. Например, если для каждого пользователя корпоративной сети определён доверительный интервал для времени его работы в системе, то факт регистрации пользователя вне этого интервала может рассматриваться как попытка получения несанкционированного доступа к ресурсам сети.

Многовариационная модель аналогична модели среднего значения и среднеквадратичного отклонения, но позволяет одновременно учитывать корреляцию между большим количеством статистических показателей.

Поведенческий метод может быть реализован также при помощи нейронных сетей и экспертных систем. В последнем случае база правил экспертной системы описывает штатное поведение корпоративной сети. Так, при помощи экспертной системы можно точно специфицировать взаимодействие между узлами сети, которое всегда осуществляется по определенным протоколам в соответствии с действующими стандартами. Если же в процессе обмена информацией между узлами будет выявлена неизвестная команда или нестандартное значение одного из параметров, это может служить признаком атаки.

Очевидно, что данная технология основана на выводе о том, что аномальное поведение субъекта (системы, программы, пользователя), т.е., как правило, атака или какое-нибудь враждебное действие часто проявляется как отклонение от нормального поведения. Примером аномального поведения может служить большое количество соединений за короткий промежуток времени, высокая загрузка центрального процессора и коэффициент сетевой нагрузки или использование периферийных устройств, которые обычно не используются. И если описать профиль нормального поведения субъекта, то любое отклонение от него можно охарактеризовать как аномальное поведение. Однако аномальное поведение не всегда является атакой. Например, таким не является приём большого числа ответов на запрос об активности станций от системы сетевого управления. Многие системы обнаружения атак идентифицируют данный случай как атаку типа «отказ в обслуживании». С учётом этого факта можно заметить, что возможны две крайности при использовании системы обнаружения аномалий:

1) обнаружение аномального поведения, которое не является атакой, и отнесение его к классу атак (false positive);

2) пропуск атаки, которая не попадает под определение аномального повеления (false negative).

Понятно, что последний случай гораздо более опасен, чем ложное причисление аномального поведения к классу атак. Поэтому при настройке и эксплуатации систем такой категории администраторы сталкиваются с двумя задачами:

1. Построение профиля субъекта – трудно формализуемая и времяёмкая задача, требующая от администратора большой предварительной работы;

2. Определение граничных значений характеристик поведения субъекта для снижения вероятности появления одного из двух вышеназванных крайних случаев.

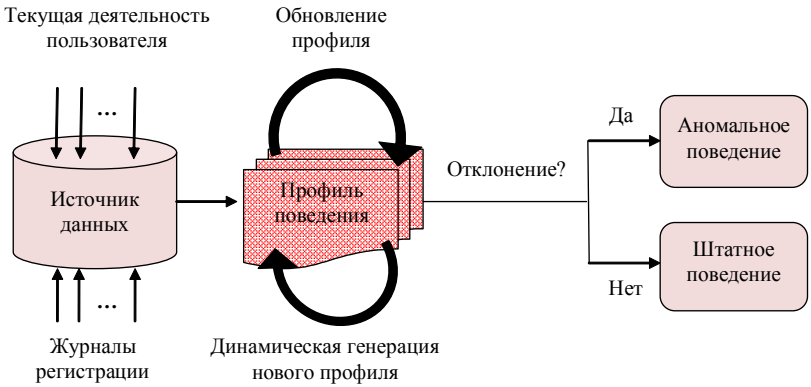

Рис. 5.5. Схема системы обнаружения аномального поведения

Схема типичной системы обнаружения аномального поведения представлена на рис. 5.5.

Обычно системы обнаружения аномальной активности используют в качестве источника данных журналы регистрации и текущую деятельность пользователя.

Данный подход получает всё большее развитие в современных системах обнаружения атак – всё чаще этот подход используют различные производители систем обнаружения атак. Так, распределенные и обычные DoS-атаки (отказ в обслуживании) обнаруживаются именно благодаря контролю за отклонениями от обычной сетевой нагрузки.

5.3. Практические аспекты выявления атак

Обнаружение атак СОА-системами должно осуществляться на различных уровнях корпоративной сети (рис. 5.6). На самом нижнем уровне СОА должны быть способны выявлять атаки на конкретных узлах корпоративной сети: рабочих станциях, серверах и маршрутизаторах. Следующий уровень обнаружения – сетевые сегменты, состоящие из группы узлов корпоративной сети. Обнаружение атак также возможно и в более крупных объединениях элементов корпоративной сети: в локальных, территориально-распределённых и глобальных системах. При этом в зависимости от инфраструктуры защищаемой сети на разных уровнях могут использоваться разные методы выявления атак.

Рассмотрим, как могут использоваться сигнатурный и поведенческий методы для обнаружения атак на различных стадиях развития. Следует отметить, что на стадии рекогносцировки, когда осуществляется сбор информации, эффективны лишь сигнатурные методы выявления атак.

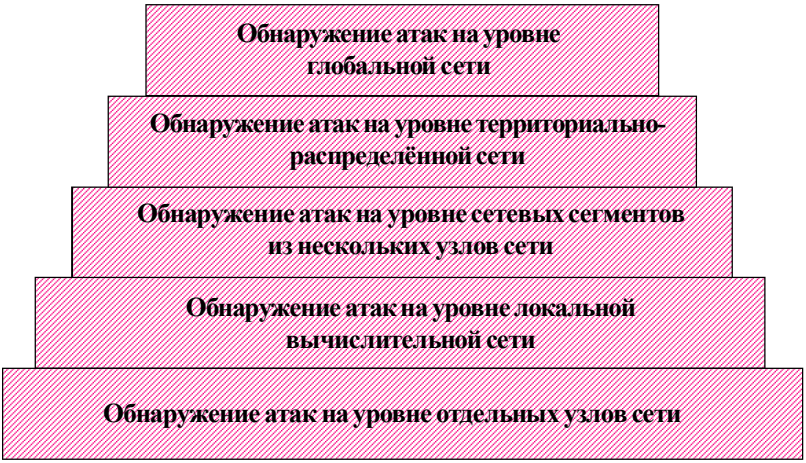

Рис. 5.6. Многоуровневая схема обнаружения атак в корпоративной сети

Это связано с тем, что все операции получения необходимой нарушителю информации в большинстве случаев не вызывают никакого отклонения штатного режима работы сети. Примерами признаков, характерных для этого этапа, являются: формирование запроса к DNS-серверу, получение информации из базы эталонных данных или многократные TCP-запросы на установление соединения с различными портами и т.д. На стадии рекогносцировки могут использоваться как сетевые, так и узловые датчики.

На стадии вторжения обнаружить атаку можно при помощи и сигнатурных, и поведенческих методов. Любое вторжение характеризуется определёнными признаками, которые, с одной стороны, могут быть представлены в виде сигнатуры, а другой - описаны как некое отклонение от штатного поведения сети. Наиболее эффективно сочетание обоих методов, при этом для получения необходимых исходных данных применимы любые (узловые или сетевые) датчики.

Эффективное выявление атак на этапах атакующего воздействия и развития атаки возможно только при помощи поведенческих методов, поскольку действия нарушителей зависят от целей проводимой атаки и фиксированным множеством сигнатур атак однозначно не определяются. Учитывая тот факт, на двух последних стадиях жизненного цикла атаки, самыми характерными объектами являются узлы, в этом случае наиболее целесообразно применение узловых датчиков. Применение сигнатурного и поведенческого методов для обнаружения атак на различных стадиях её существования приведено в табл. 5.1.

Таблица 5.1

Методы обнаружения атак на разных стадиях жизненного цикла атаки

Стадия атаки	Метод обнаружения	
	сигнатурный	поведенческий
Рекогносцировка	+ СУ	–
Вторжение	+ СУ	+ СУ
Атакующее воздействие	–	+ У
Развитие	–	+ У

Примечание: + - метод применим; – - метод неприменим; СУ - используются сетевые и узловые датчики; У - используются узловые датчики.

Обнаружение атак на ресурсы корпоративной сети является весьма сложным технологическим процессом, который связан со сбором немалых объёмов информации о функционировании сети, анализом этих данных и, наконец, выявлением факта атаки. Для эффективного обнаружения атаки на всех стадиях её жизненного цикла требуется совместное применение как поведенческих, так и сигнатурных методов. Соответственно, только комплексный подход к данной проблеме позволит значительно снизить риск вторжения в информационную систему и исключит потерю конфиденциальной информации.

5.4. Межсетевые экраны

Проблема защиты от несанкционированных действий при взаимодействии с внешними сетями успешно может быть решена с помощью специализированных программно-аппаратных комплексов, обеспечивающих целостную защиту компьютерной сети от потенциально враждебной внешней среды. Такие комплексы называют межсетевыми экранами, брандмауэрами или системами Firewall.

Межсетевой экран – это система межсетевой защиты, позволяющая разделить каждую сеть на две и более частей и реализовать набор правил, определяющих условия прохождения пакетов данных через границу из одной части общей сети в другую. Как правило, эта граница проводится между корпоративной (локальной) сетью предприятия и глобальной сетью Internet, хотя её можно провести и внутри корпоративной сети предприятия. Использование межсетевых экранов позволяет организовать внутреннюю политику безопасности сети предприятия, разделив всю сеть на сегменты, что позволяет сформулировать основные принципы архитектуры безопасности корпоративной сети:

1. Введение N категорий секретности и создание N выделенных сетевых сегментов пользователей. При этом каждый пользователь внутри

сетевого сегмента имеет одинаковый уровень секретности (допущен к информации одного уровня секретности).

2. Выделение в отдельный сегмент всех внутренних серверов компании. Эта мера также позволяет изолировать потоки информации между пользователями, имеющими различные уровни доступа.

3. Выделение в отдельный сегмент всех серверов компании, к которым будет предоставлен доступ из Интернета (создание демилитаризованной зоны для внешних ресурсов).

4. Создание выделенного сегмента административного управления.

5. Создание выделенного сегмента управления безопасностью.

Для противодействия несанкционированному межсетевому доступу брандмауэр должен располагаться между защищаемой сетью организации, являющейся внутренней, и потенциально враждебной внешней сетью (рис. 5.7). При этом все взаимодействия между этими сетями должны осуществляться только через межсетевой экран. Организационно экран входит в состав защищаемой сети.

Межсетевой экран не является симметричным. Для него отдельно задаются правила, ограничивающие доступ из внутренней сети во внешнюю сеть и наоборот. В общем случае работа межсетевого экрана основана на динамическом выполнении двух групп функций:

1) фильтрации проходящих через него информационных потоков;

2) посредничества при реализации межсетевых взаимодействий.

В зависимости от типа экрана эти функции могут выполняться с различной полнотой. Простые межсетевые экраны ориентированы на выполнение только одной из данных функций. Комплексные экраны обеспечивают совместное выполнение указанных функций защиты.

Фильтрация состоит в выборочном пропускании информационных потоков через экран и извещением отправителя о том, что его данным в пропуске отказано (рис. 5.8).

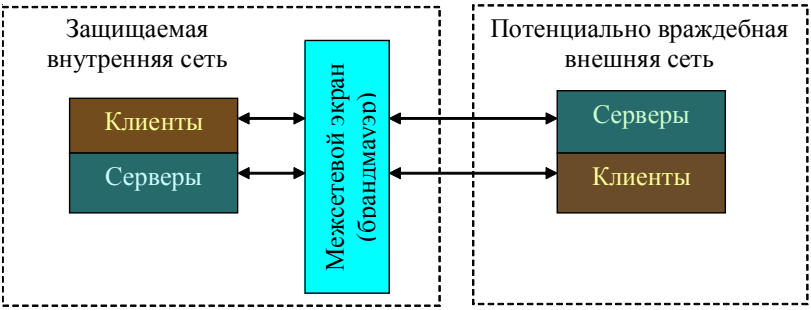

Рис. 5.7. Схема подключения межсетевого экрана

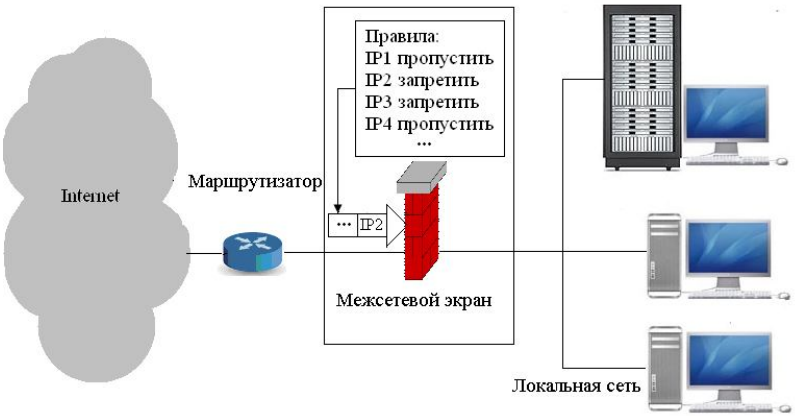

Рис. 5.8. Функция фильтрации, реализованная в межсетевом экране

Фильтрация осуществляется на основе набора правил, предварительно загруженных в экран и являющихся по своей сути принятой политикой безопасности. Для реализации этой функции Межсетевой экран представляется как последовательность фильтров, обрабатывающих информационный поток (рис.5.9).

Каждый из фильтров предназначен для отдельных правил фильтрации путем выполнения следующих стадий:

1. Анализ фильтруемых данных по заданным в правилах политики безопасности критериям, например, по адресам получателя и отправителя или по типу приложения, для которого эта информация предназначена.

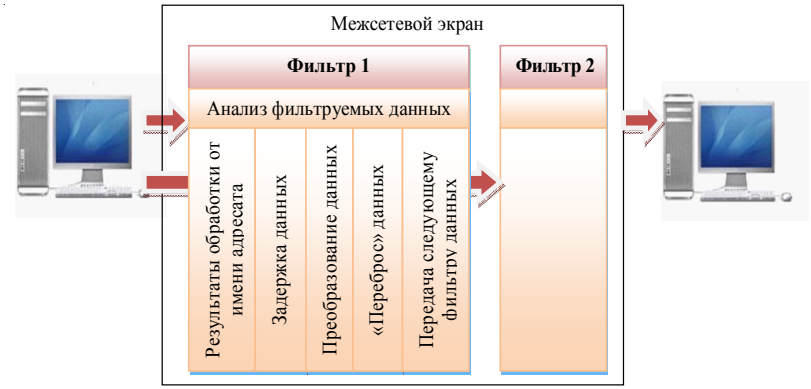

Рис. 5.9. Межсетевой экран как последовательность фильтров

2. Принятие на основе правил принятой политики безопасности одного из следующих решений:

a) не пропустить данные;

b) обработать данные от имени адресата (получателя) и возвратить результат отправителю;

c) передать данные на следующий фильтр для продолжения анализа;

d) пропустить данные, игнорируя следующие фильтры («переброс» данных).

Функции посредничества межсетевой экран выполняет с помощью специальных программ, называемых экранирующими агентами или просто программами-посредниками. Данные программы являются резидентными и запрещают непосредственную передачу пакетов сообщений между внешней и внутренней сетью.

При необходимости доступа из внутренней сети во внешнюю сеть или наоборот вначале должно быть установлено логическое соединение с программой-посредником, функционирующей на компьютере экрана. Программа-посредник проверяет допустимость запрошенного межсетевого взаимодействия и при его разрешении сама устанавливает отдельное соединение с требуемым компьютером. Далее обмен информацией между компьютерами внутренней и внешней сети осуществляется через программного посредника, который может выполнять фильтрацию потока сообщений, а также осуществлять другие защитные функции.

К функциям посредничества в общем случае относятся:

Идентификация и аутентификация пользователей. Для высокой степени безопасности необходима идентификация и аутентификация пользователей не только при их доступе из внешней сети во внутреннюю сеть, но и наоборот. Пароль не должен передаваться в открытом виде через общедоступные коммуникации. Оптимальным способом аутентификации является использование одноразовых паролей. Удобно и надежно также применение цифровых сертификатов, выдаваемых доверительными органами, например, центром распределения ключей. Большинство программ-посредников разрабатываются таким образом, чтобы пользователь аутентифицировался только в начале сеанса работы с межсетевым экраном. После этого от него не требуется дополнительная аутентификация в течение времени, определяемого администратором.

Проверка подлинности передаваемых данных. Программы-посредники могут осуществлять проверку подлинности получаемых и передаваемых данных. Это актуально не только для аутентификации электронных сообщений, но и мигрирующих программ (Java, ActiveX Controls), по отношению к которым может быть выполнен подлог. Проверка подлинности сообщений и программ заключается в проверке их цифро-

вых подписей. Для этого также могут применяться цифровые сертификаты.

Разграничение доступа к ресурсам внутренней сети. Идентификация и аутентификация пользователей при обращении к межсетевому экрану позволяет разграничить их доступ к ресурсам внутренней или внешней сети. Способы разграничения к ресурсам внутренней сети ничем не отличаются от способов разграничения, поддерживаемых на уровне операционной системы.

При разграничении доступа к ресурсам внешней сети чаще всего используется один из следующих подходов:

- разрешение доступа только по заданным адресам во внешней сети;
- фильтрация запросов на основе обновляемых списков недопустимых адресов и блокировка поиска информационных ресурсов по нежелательным ключевым словам;
- накопление и обновление администратором санкционированных информационных ресурсов внешней сети в дисковой памяти межсетевого экрана и полный запрет доступа во внешнюю сеть.

Фильтрация и преобразование потока сообщений. Под функциями фильтрации и преобразования потока сообщений понимается, например, динамический поиск вирусов и прозрачное шифрование информации. Фильтрация и преобразование потока сообщений выполняется посредником на основе заданного набора правил.

Программный посредник анализирует поступающие к нему пакеты данных и если какой-либо объект не соответствует заданным критериям, то посредник либо блокирует его дальнейшее продвижение, либо выполняет соответствующие преобразования, например, обезвреживание обнаруженных компьютерных вирусов.

Трансляция внутренних сетевых адресов для исходящих пакетов сообщений. Программы-посредники могут выполнять и такую важную функцию, как трансляция внутренних сетевых адресов**.** Данная функция реализуется по отношению ко всем пакетам, следующим из внутренней сети во внешнюю.

Для этих пакетов посредник выполняет автоматическое преобразование IP-адресов компьютеров-отправителей в один “надёжный” IP-адрес, ассоциируемый с межсетевым экраном, из которого передаются все исходящие пакеты. В результате все исходящие из внутренней сети пакеты оказываются отправленными межсетевым экраном, что исключает прямой контакт между авторизованной внутренней сетью и являющейся потенциально опасной внешней сетью. IP-адрес межсетевого экрана становится единственным активным IP-адресом, который попадает во внешнюю сеть.

Технология заключается в том, что на межсетевом экране, который играет роль маршрутизатора, при выходе во внешнюю сеть во всех сете-

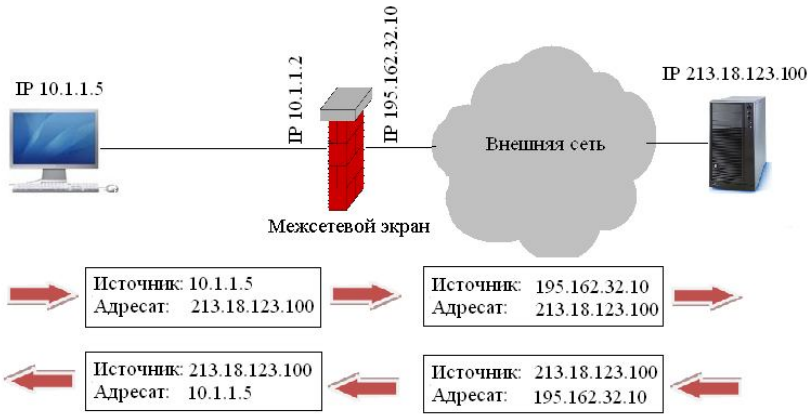

Рис. 5.10. Трансляция адресов на межсетевом экране

вых пакетах производится подмена внутреннего адреса на предопределённый внешний адрес. При этом маршрутизатор ведет таблицу соответствия отправленных пакетов таким образом, что для входящих пакетов из внешней сети производится обратная замена внешнего адреса на внутренний (рис.5.10).

При таком подходе топология внутренней сети скрыта от внешних пользователей, что усложняет задачу несанкционированного доступа.

Регистрация событий, реагирование на задаваемые события, а также анализ зарегистрированной информации и генерация отчетов. В качестве обязательной реакции на обнаружение попыток выполнения несанкционированных действий должно быть определено уведомление администратора, т.е. выдача предупредительных сигналов. Любой брандмауэр, который не способен посылать предупредительные сигналы при обнаружении нападения, не является эффективным средством межсетевой защиты.

Многие межсетевые экраны содержат мощную систему регистрации, сбора и анализа статистики. Учёт может вестись по адресам клиента и сервера, идентификаторам пользователей, времени сеансов, времени соединений, количеству переданных/принятых данных, действиям администратора и пользователей. Системы учёта позволяют произвести анализ статистики и представляют администраторам подробные отчёты. За счёт использования специальных протоколов посредники могут выполнить удалённое оповещение об определенных событиях в режиме реального времени.

Кэширование данных, запрашиваемых из внешней сети. При доступе пользователей внутренней сети к информационным ресурсам внешней сети вся информация накапливается на пространстве жесткого

диска брандмауэра, называемого в этом случае proxy-сервером. Поэтому если при очередном запросе нужная информация окажется на proxy-сервере, то посредник перешлёт её без обращения к внешней сети, что существенно ускоряет доступ. Администратору следует позаботиться только о периодическом обновлении содержимого proxy-сервера.

Функция кэширования успешно может использоваться для ограничения доступа к информационным ресурсам внешней сети. В этом случае все санкционированные информационные ресурсы внешней сети накапливаются и обновляются администратором на proxy-сервере. Пользователям внутренней сети разрешается доступ только к информационным ресурсам proxy-сервера, а непосредственный доступ к ресурсам внешней сети запрещается.

Для подключения межсетевых экранов используются различные схемы. Для подключения к внешней сети межсетевой экран может быть использован в качестве внешнего маршрутизатора (рис. 5.11).

Иногда находит применение схема, изображенная на рис. 5.12, однако использовать её следует только в крайнем случае, поскольку требуется очень аккуратная настройка маршрутизаторов и небольшие ошибки могут образовать серьёзные бреши в защите.

Если межсетевой экран может поддерживать два Ethernet интерфейса (так называемый dual-homed брандмауэр), то чаще всего подключение осуществляется через внешний маршрутизатор (рис. 5.13).

При этом между внешним маршрутизатором и межсетевым экраном имеется только один путь, по которому идет весь трафик. Обычно маршрутизатор настраивается таким образом, что брандмауэр является

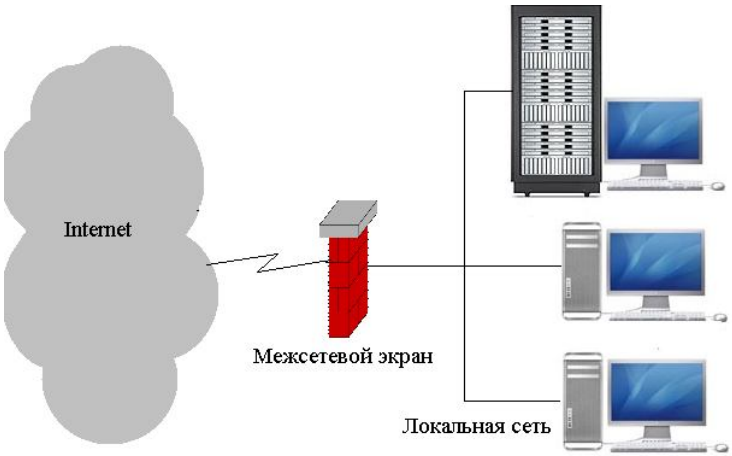

Рис. 5.11. Межсетевой экран с функциями маршрутизатора

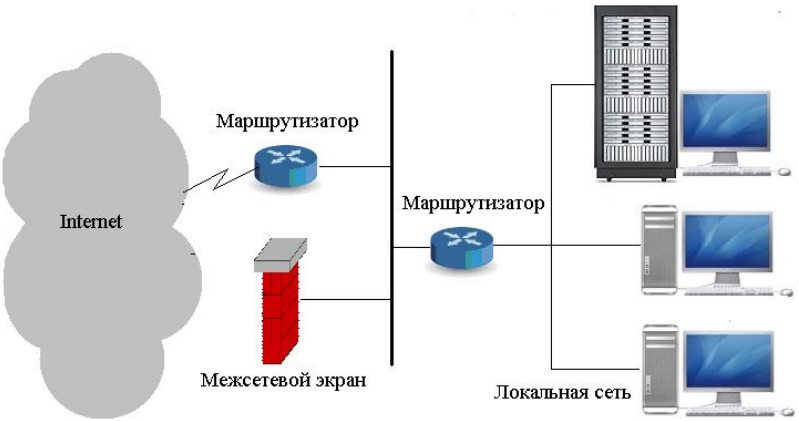

Рис. 5.12. Вариант подключения межсетевого экрана

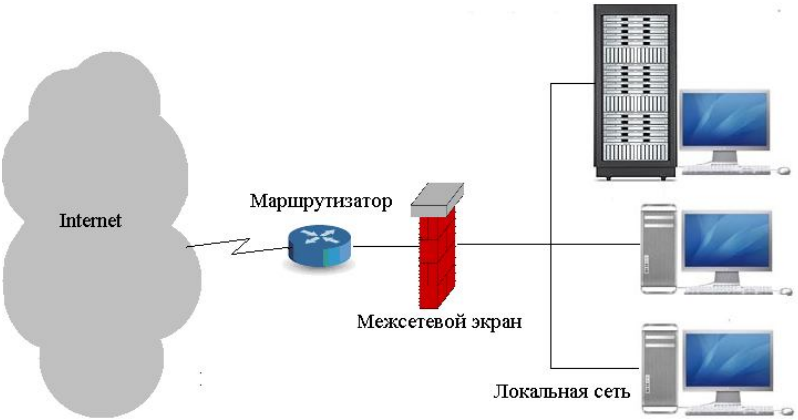

Рис. 5.13. Схема подключения межсетевого экрана, поддерживающего два Ethernet интерфейса

единственной видимой снаружи машиной. Эта схема является наиболее предпочтительной с точки зрения безопасности и надежности защиты.

Другая схема представлена на рис. 5.14. В этом варианте межсетевым экраном защищается только одна подсеть из нескольких выходящих из маршрутизатора. В незащищаемой межсетевым экраном области часто располагают серверы, которые должны быть видимы снаружи (WWW, FTP и т.д.). Некоторые производители межсетевых экранов предлагают разместить эти серверы на самом брандмауэре. Такие решения не являются рациональными с точки зрения загрузки машины и безопасности межсетевого экрана.

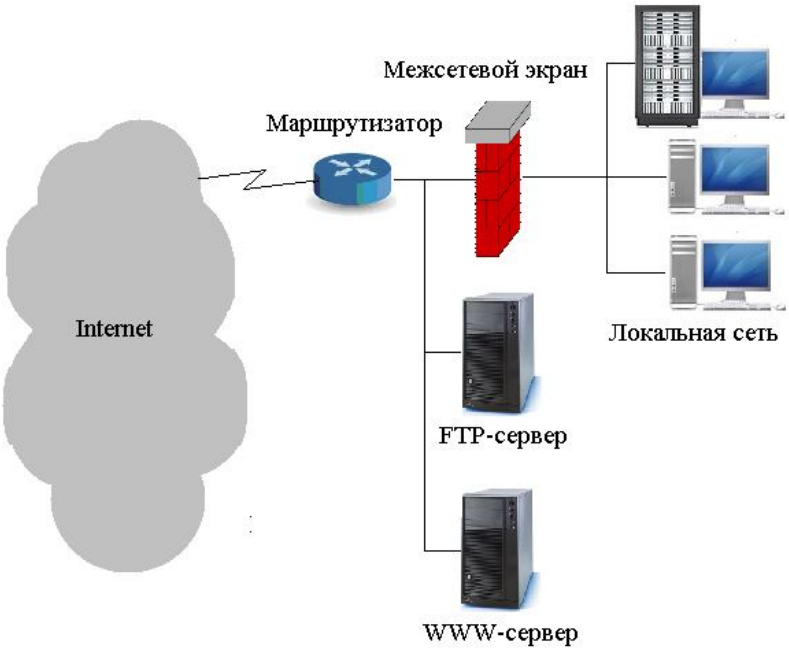

Рис. 5.14. Межсетевой экран защищает только локальную сеть предприятия

Современное развитие бизнеса предполагает, что внутренние ресурсы организации не должны быть полностью закрыты. Ряд узлов, таких как WWW-сервер, FTP-сервер, почтовый сервер должны быть в той или иной степени доступны для внешних пользователей, в том числе для тех, о ком нет никакой предварительной информации. Возникает вопрос, где размещать такие узлы. Если во внешней сети, перед межсетевым экраном, это значит, что их защищенность будет зависеть только от схемы безопасности операционной системы и приложения, что, как показывает опыт, недостаточно. Если разместить их во внутренней сети, за межсетевым экраном, то тогда придется пропускать внешних пользователей во внутреннюю сеть, а это всегда небезопасно, даже при точной настройке правил доступа. Вполне логично напрашивается вывод – создать для подобных ресурсов отдельную подсеть, свободную от элементов внутренней и внешней сети. Данная технология получила название демилитаризованной зоны (ДМЗ).

Поскольку обычно межсетевые экраны имеют по два сетевых интерфейса (один во внутреннюю и один во внешнюю сети), то для ДМЗ необходим третий сетевой интерфейс. Отдельные правила, прописанные

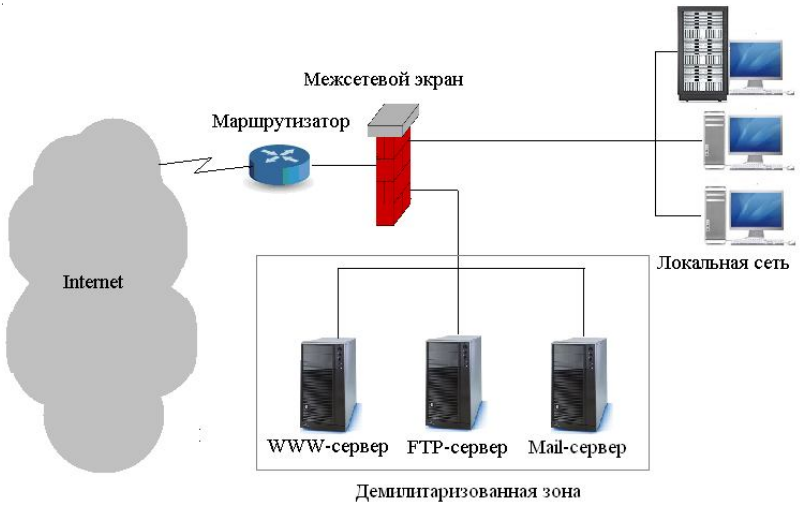

Рис. 5.15. Схема образования демилитаризованной зоны

на межсетевом экране для доступа в ДМЗ, позволят, с одной стороны, обеспечить защиту корпоративных ресурсов, а с другой стороны, не предоставят дополнительного доступа в локальную сеть (рис. 5.15).

При этом достаточно много внимания уделяется тому, чтобы пользователи внутренней сети не могли случайно или умышленно открыть брешь в локальную сеть через эти сервера. Для повышения уровня защищённости возможно использовать в одной сети несколько брандмауэров, стоящих друг за другом.

Выводы

Системы обнаружения атак позволяют своевременно выявлять и блокировать атаки нарушителей.

Процесс выявления атаки состоит из следующих этапов:

- сбор данных, необходимых для определения факта атаки на ресурсы сети;
- анализ данных, собранных сетевыми и узловыми датчиками;
- выявление атаки;
- реагирование на выявленную атаку.

При анализе данных, собранных сетевыми и узловыми датчиками, система обнаружения атак использует сигнатурные и поведенческие методы выявления атак.

Сигнатурные методы описывают каждую атаку в виде специальной модели или сигнатуры. В исходных данных, собранных сетевыми и узловыми датчиками выполняется процедура поиска сигнатуры атаки с ис-

пользованием базы данных сигнатур атак. Преимущество сигнатурных методов – высокая точность определения факта атаки, а недостаток – невозможность обнаружения тех атак, сигнатуры которых пока не определены.

Поведенческие методы базируются на моделях штатного процесса поведения сети. Методы основаны на обнаружении несоответствия между текущим режимом работы сети и режимом работы, соответствующим штатной модели данного метода. Любое несоответствие рассматривается как атака. Преимущество поведенческих методов – возможность обнаружения новых атак без модификаций или обновлений параметров модели, недостаток – сложно создать точную модель штатного режима функционирования сети.

Обнаружение атак должно осуществляться на различных уровнях корпоративной сети: конкретных узлах корпоративной сети, сетевых сегментах, локальных, территориально-распределённых и глобальных системах.

Использование межсетевых экранов позволяет организовать внутреннюю политику безопасности сети предприятия, разделив всю сеть на сегменты. Деление на сегменты позволяет ввести категории секретности и создание уровней секретности, выделить в отдельный сегмент все внутренние серверы компании, создать демилитаризованную зону для внешних ресурсов, создать выделенный сегмент административного управления и выделенный сегмент управления безопасностью.

Работа межсетевого экрана основана на динамическом выполнении двух групп функций: фильтрации информационных потоков и посредничества при реализации межсетевых взаимодействий.

Фильтрация состоит в выборочном пропускании информационных потоков в соответствии с принятой политикой безопасности через экран и извещением отправителя о том, что его данным в пропуске отказано.

Функции посредничества межсетевой экран выполняет с помощью специальных программ-посредников, запрещающих непосредственную передачу пакетов данных между внешней и внутренней сетями. К функциям посредничества относятся: идентификация и аутентификация пользователей, проверка подлинности передаваемых данных, разграничение доступа к ресурсам внутренней сети, поиск вирусов, трансляция внутренних сетевых адресов для исходящих пакетов данных и др.

6. ЭЛЕКТРОННАЯ ПОЧТА И ТЕХНОЛОГИИ ЕЁ ЗАЩИТЫ

6.1. Основные элементы службы электронной почты

Электронная почта (англ. *email, e-mail*, от англ. *electronic mail*) - технология и предоставляемые ею услуги по пересылке и получению электронных сообщений (называемых «письма» или «электронные письма») по распределённой (в том числе глобальной) компьютерной сети.

Структура электронного сообщения состоит из двух частей:

1. ***Заголовок письма***, иногда называемый по аналогии с бумажной почтой *конвертом*. В заголовке указывается служебная информация и пометки почтовых серверов, через которые прошло письмо, пометки о приоритете, указание на адрес и имя отправителя и получателя письма, тема письма и другая информация. В «электронный конверт», как и в обычный, можно вложить фотографии, картинку, рисунок и даже звуковой фрагмент.

2. ***Тело письма***. В теле письма находится, собственно, текст письма.

Общее развитие электронной почты шло через развитие локального взаимодействия пользователей на многопользовательских системах. Пользователи могли, используя программу mail (или её эквивалент), пересылать друг другу сообщения в пределах одного мейнфрейма (большого компьютера).

Следующий шаг был в возможности переслать сообщение пользователю на другой машине. Для этого использовалось указание имени машины и имени пользователя на машине. Адрес мог записываться в виде *station*!*joe* (пользователь *joe* на компьютере *station*).

Третий шаг для становления электронной почты произошёл в момент появления передачи писем через третий компьютер. Адрес пользователя включал в себя маршрут до пользователя через несколько промежуточных машин (например, *gate*1!*gate*2!*station*!*joe* - письмо для *joe* через машину *gate*1, *gate*2 на машину *station*). Недостатком такой адресации было то, что отправителю (или администратору машины, на которой работал отправитель) необходимо было знать точный путь до машины адресата.

После появления распределённой глобальной системы имён DNS, для указания адреса стали использоваться доменные имена - *user@example.com* - пользователь *user* на машине *example.com*. Одновременно с этим для почты стали использоваться выделенные серверы, на которые не имели доступ обычные пользователи (только администраторы), а пользователи работали на своих машинах, при этом почта приходила не на ра-

бочие машины пользователей, а на почтовый сервер, откуда пользователи забирали свою почту по различным сетевым протоколам.

Обычно почтовая система поддерживает пять базовых функций:

Композиция. Обеспечивает создание сообщений и ответов. Хотя для формирования тела сообщения может быть использован любой текстовый редактор, но система обеспечивает заполнение многочисленных полей заголовка сообщения. Например, если формируется ответ, то система автоматически выделит адрес из исходного сообщения и подставит его, как адрес получателя.

Передача. Эта функция обеспечивает передачу сообщения от отправителя к получателю без вмешательства пользователей.

Отчёт перед отправителем о доставке: было ли сообщение доставлено, было ли отвергнуто, было ли потеряно. Для многих приложений эти отчёты важны.

Показ сообщения является существенной функцией почтовой службы. Часто она должна выполнять перекодировку, изменять формат и т.д.

Размещение - это последний этап, на котором определяется, что делать с сообщением: надо ли его уничтожить после прочтения или до, если сохранить, то где, поиск интересующего сообщения, перенаправление сообщения, повторное прочтение ранее полученного сообщения.

Кроме этих обязательных функций большинство почтовых систем имеют ряд более сложных функций. Например, если пользователь уехал, он может перенаправить сообщения, поступающие в его отсутствие, куда-то ещё. Во многих системах пользователь может создавать, так называемые, почтовые ящики для поступающих сообщений, создавать лист рассылки, по которому одно и то же сообщение будет разослано всем его участникам. Важной функцией является сообщение с уведомлением, так как в любом случае для пользователя полезно получать сообщения о состоянии его сообщения, создание копии отправленных сообщений, приоритетность сообщений, секретность и т.д.

Ключевым моментом всех современных почтовых систем - разделение конверта для сообщения и собственно сообщения. Системой доставки используется конверт. Он содержит всю необходимую информацию о сообщении: адрес назначения, приоритет, секретность, требование об уведомлении и т.д.

Таким образом, система электронной почты на основе этих базовых функций дает возможность:

- посылать и получать сообщения;
- отвечать на письма корреспондентов автоматически, используя их адреса;
- рассылать копии письма сразу нескольким получателям;
- переправлять полученное письмо по другому адресу;

- использовать вместо адресов (числовых или доменных имен) логические имена;

- создавать несколько подразделов почтового ящика для разного рода корреспонденции;

- включать в письма текстовые файлы;

- пользоваться системой "отражателей почты" для ведения дискуссий с группой ваших корреспондентов и т.д.

Основными элементами службы электронной почты являются:

Почтовый сервер (сервер электронной почты – mail server) - сервер, обеспечивающий приём и передачу электронных писем пользователей, а также их маршрутизацию.

Почтовый ящик (mailbox) - область дискового пространства, в которой хранятся сообщения электронной почты, адресованные конкретному пользователю в корпоративной сети. Сообщения хранятся до тех пор, пока они не будут изъяты или переправлены в другой почтовый ящик.

Почтовый адрес (адрес электронной почты - e-mail address) - уникальный идентификатор почтового ящика пользователя. В сети Интернет почтовый адрес имеет вид ИмяПочтовогоЯщика@ИмяПочтовогоСервера. Символ @ («собачка») разделяет имя пользователя (логин) и адрес сервера.

Почтовый клиент (mail client) - программа, предназначенная для чтения, приема, отправки и других операций с письмами. С её помощью пользователь имеет возможность работать с почтовыми серверами.

Почтовый протокол, с помощью которого осуществляется пересылка почтового сообщения от клиента на почтовый клиент и загрузка почты с почтового сервера адресату.

6.2. Обзор почтовых протоколов

Развитие сетевых технологий привело к появлению современных протоколов для обмена сообщениями, которые предоставляют большие возможности для обработки писем, разнообразные сервисы и удобство в работе.

Среди распространённых на настоящий момент протоколов электронной почты - SMTP POP3, IMAP.

SMTP - это почтовый протокол, работающий по технологии «клиент-сервер». SMTP-сервер принимает письма от других систем и сохраняет их в почтовых ящиках пользователей. Сохраненные письма могут быть прочитаны несколькими способами. Пользователи с интерактивным доступом на почтовом сервере могут читать почту с помощью локальных почтовых приложений. Пользователи на других системах могут загрузить свои письма с помощью программ-почтовых клиентов по протоколам POP3 и IMAP (рис. 6.1).

Рис. 6.1. Место протоколов SMT, POP, IMAP в службе электронной почты

Протокол SMTP представляет собой простой ASCII протокол. Установив TCP-соединение с портом 25, передающая машина, выступающая в роли клиента, ждёт запроса принимающей машины, работающей в режиме сервера. Сервер начинает диалог с того, что посылает текстовую строку, содержащую его идентификатор и сообщающую о его готовности (или неготовности) к приёму почты. Если сервер не готов, клиент разрывает соединение и продолжает попытку позднее.

Протокол поддерживает создание очередей сообщений, переписывание заголовков писем, алиасы, списки рассылки и т.д.

Обычно доступ к серверу SMTP не защищается паролем, так что можно использовать для отправки писем любой известный сервер в сети. Обычно он конфигурируется так, что должен работать как привилегированный процесс. Это означает, что, если его защиту можно будет обойти каким-нибудь способом, атакующий сможет нанести вред, далеко превышающий удаление электронных писем.

В отличие от серверов для отправки писем, доступ к серверам для хранения сообщений защищается паролем. Поэтому необходимо использовать сервер или службу, в которой существует учётная запись. Эти серверы работают по протоколам POP и IMAP, которые различаются способом хранения писем.

У SMTP-протокола есть несколько проблем. Первая - длина сообщения не может превосходить 64 Кбайт. Другая - time out. Если задержка у отправителя и получателя не согласована, то один будет разрывать соединение, не дождавшись, тогда как другой просто очень загружен.

POP - это самый популярный протокол приёма электронной почты. POP-сервер позволяет POP-клиенту загрузить письма, которые были получены им от другого почтового сервера. Клиенты могут загрузить все

сообщения или только те, которые они ещё не читали. Он не поддерживает удаление сообщений перед загрузкой на основе атрибутов сообщения, таких как адрес отправителя или получателя. POP версии 2 поддерживает аутентификацию пользователя с помощью пароля, но пароль передается серверу в открытом (незашифрованном) виде.

POP версия 3 (POP3) представляет дополнительный метод аутентификации, называемой APOP, которая прячет пароль. Некоторые реализации POP могут использовать Kerberos для аутентификации.

В соответствии с протоколом POP3 поступающие на определенный адрес сообщения хранятся на сервере до того момента, пока они не будут в течение очередного сеанса загружены на компьютер. После загрузки сообщений, можно отключиться от сети и приступить к чтению почты. Таким образом, использование почты по протоколу POP3 является наиболее быстрым и удобным в использовании.

IMAP - это самый новый и поэтому менее популярный протокол чтения электронной почты.

Как сказано в RFC: IMAP поддерживает операции создания, удаления, переименования почтовых ящиков; проверки поступления новых писем; оперативное удаление писем; установку и сброс флагов операций; поиск среди писем; выборочное чтение писем.

Протокол IMAP более удобен для чтения почты в путешествии, чем протокол POP, так как сообщения могут быть оставлены на сервере, что избавляет от необходимости синхронизировать списки прочитанных писем на локальном хосте и на сервере.

Протокол IMAP удобен тем людям, которые пользуются постоянным подключением к сети. Сообщения, поступившие на адрес, также хранятся на сервере, но, в отличие от POP3, при проверке почты сначала будут загружены только заголовки сообщений. Само письмо можно будет прочитать после выбора заголовка сообщения (оно загрузиться с сервера). Ясно, что при коммутируемом соединении работа с почтой по этому протоколу приводит к неоправданным потерям времени. Протокол позволяет получать доступ к письму не только по его номеру, но и по содержанию.

6.3. Угрозы безопасности электронной почты

Хотя электронная почта является дешевым способом взаимодействия между деловыми партнерами, с её использованием связан ряд проблем с безопасностью:

1. Адреса электронной почты в Интернете легко подделать. Практически нельзя сказать наверняка, кто написал и послал электронное письмо только на основе его адреса.

2. Электронные письма могут быть легко модифицированы. Стандартное SMTP-письмо не содержит средств проверки целостности.

3. Существует ряд мест, где содержимое письма может быть прочитано теми, кому оно не предназначено. Электронное письмо, скорее, похоже на открытку - его могут прочитать на каждой промежуточной станции.

4. Обычно нет гарантий доставки электронного письма. Хотя некоторые почтовые системы предоставляют Вам возможность получить сообщение о доставке, часто такие уведомления означают лишь то, что почтовый сервер получателя (а не обязательно сам пользователь) получил сообщение.

Перечисленные проблемы безопасности службы электронной почты создают соответствующие угрозы, которые можно классифицировать следующим образом.

Случайные ошибки. Можно допустить ошибку при работе с электронной почтой. Письмо может быть случайно послано. Простое нажатие клавиши или щелчок мышкой могут послать письмо по неправильному адресу. Почтовые сообщения могут храниться годами, поэтому случайная ошибка может нанести вред через много времени. Архивы писем могут возрасти до такой степени, что система будет аварийно завершаться. Неправильно настроенная программа чтения групп новостей может привести к посылке сообщения не в те группы. Ошибки в списках рассылки могут привести к долгому блужданию писем между почтовыми серверами, причём число писем может увеличиться до такой степени, что почтовые сервера аварийно завершатся. Когда почтовая система организации присоединена к Интернету, последствия ошибок могут оказаться в тысячу раз хуже.

Фальшивые адреса отправителя. Основные протоколы электронной почты (SMTP, POP, IMAP) обычно не осуществляют надежной аутентификации, что позволяет легко создать письма с фальшивыми адресами. Ни один из этих протоколов не использует криптографию, которая могла бы гарантировать конфиденциальность электронных писем. Хотя существуют расширения этих протоколов, решение использовать их должно быть явно принято как составная часть политики администрации почтового сервера. Некоторые такие расширения используют уже имеющиеся средства аутентификации, а другие позволяют клиенту и серверу согласовать тип аутентификации, который будет использоваться в данном соединении. Адресу отправителя в электронной почте Интернета нельзя доверять, так как отправитель может указать фальшивый обратный адрес или заголовок может быть модифицирован в ходе передачи письма, или отправитель может сам соединиться с SMTP-портом на машине, от имени которой он хочет отправить письмо, и ввести текст письма.

Перехват письма. Заголовки и содержимое электронных писем передаются в чистом виде. В результате содержимое сообщения может быть прочитано или изменено в процессе передачи его по Internet. Заголовок может быть модифицирован, чтобы скрыть или изменить отправителя или для того, чтобы перенаправить сообщение.

Почтовые бомбы. Почтовая бомба - это атака с помощью электронной почты. Атакуемая система переполняется письмами до тех пор, пока она не выйдет из строя. Как это может случиться, зависит от типа почтового сервера и того, как он сконфигурирован. Некоторые провайдеры Internet дают временные логины любому для тестирования подключения к Internet, и эти логины могут быть использованы для начала подобных атак.

Типовые варианты выхода почтового сервера из строя:

- Почтовые сообщения принимаются до тех пор, пока диск, где они размещаются, не переполнится. Следующие письма не принимаются. Если этот диск также основной системный диск, то вся система может аварийно завершиться.

- Входная очередь переполняется сообщениями, которые нужно обработать и передать дальше, до тех пор, пока не будет достигнут предельный размер очереди. Последующие сообщения не попадут в очередь.

- У некоторых почтовых систем можно установить максимальное число почтовых сообщений или максимальный общий размер сообщений, которые пользователь может принять за один раз. Последующие сообщения будут отвергнуты или уничтожены.

- Может быть превышена квота диска для данного пользователя. Это помешает принять последующие письма, и может помешать ему выполнять другие действия. Восстановление может оказаться трудным для пользователя, так как ему может понадобиться дополнительное дисковое пространство для удаления писем. Большой размер почтового ящика может сделать трудным для системного администратора получение системных предупреждений и сообщений об ошибках.

- Посылка почтовых бомб в список рассылки может привести к тому, что его члены могут отписаться от списка.

Вредоносные программы. Вредоносное программное обеспечение, которое пересылается вместе с электронным сообщением может нанести непоправимый ущерб серверам, рабочим станциям и находящейся в них информации – исказить или уничтожить данные, блокировать работу приложений и операционной системы в целом.

Фишинговые ссылки. Фишинговые ссылки, которые в огромных количествах рассылаются в электронных сообщениях, также являются серьезной угрозой для организации, владеющей конфиденциальной информацией. Переход по фишинговым ссылкам на хакерские сайты гро-

зит тем, что на компьютеры пользователей будут незаметно установлены программы, позволяющие получить злоумышленникам доступ к ценной персональной информации, логинам и паролям от корпоративных ресурсов.

Спам. Спам (англ. *spam*) – массовая рассылка коммерческой, политической и иной рекламы (информации) или иного вида сообщений лицам, не выражавшим желания их получать. Такие сообщения не только серьезно загружают память, но и ежедневно отвлекают сотрудников от выполнения служебных обязанностей.

Через электронную почту распространяется самый большой поток спама. В настоящее время доля вирусов и спама в общем трафике электронной почты составляет по разным оценкам от 70 до 95 процентов.

Массовая рассылка спама имеет низкую себестоимость для отправителя в расчёте на сообщение. Однако огромное количество бесполезных сообщений наносит очевидный вред получателям. В первую очередь речь идёт о времени, потраченном впустую на отсеивание ненужной почты и выискивании среди неё отдельных нужных писем. Очень часто интернет-трафик стоит дорого, и пользователю приходится платить за очевидно ненужные письма. Кроме того провайдерам приходится тратить ресурсы на избыточное оборудование и системы защиты от спама (избыточное оборудование, избыточная ёмкость каналов, специальное программное обеспечение для распознавания спама). Спам также наносит вред репутации компании, если спам используется в недобросовестной конкуренции и «чёрном» пиаре.

По данным Лаборатории Касперского, в феврале 2010 г. почтовый спам в рунете распределился по тематике следующим образом: 18,9 % – образование,15,7 % – отдых и путешествия, 15,5 % – медикаменты, товары/услуги для здоровья, 9,2 % – компьютерное мошенничество, 6,5 % – компьютеры и интернет, 5,2 % – реплики элитных товаров, 4,1 % – реклама спамерских услуг, 2,7 % – для взрослых, 2,2 % – недвижимость, 2,2 % – юридические услуги, 1,9 % – личные финансы, 1,4 % – полиграфия.

Сбор e-mail адресов для рассылки спама осуществляется с помощью специального робота, используя веб-страницы, конференции Usenet, списки рассылки, электронные доски объявлений, гостевые книги, чаты и другое. Такая программа-робот способна собрать за час тысячи адресов и создать из них базу данных для дальнейшей рассылки по ним спама.

6.4. Защита электронных писем и почтовых систем

Защита от случайных ошибок. Вот некоторые из способов предотвратить случайные ошибки:

- учить пользователей как правильно работать с электронной почтой и что делать, если они совершили ошибку;

- конфигурировать программы электронной почты так, чтобы стандартные действия пользователя, использующие установки по умолчанию, были бы самыми безопасными;

- использовать программы, которые строго реализуют протоколы и соглашения Internet.

Защита от фальшивых адресов. От этого можно защититься с помощью использования шифрования для присоединения к письмам электронных подписей. Одним популярным методом является использование шифрования с открытыми ключами. Однонаправленная хэш-функция письма шифруется, используя секретный ключ отправителя. Получатель использует открытый ключ отправителя для расшифровки хэш-функции и сравнивает его с хэш-функцией, рассчитанной по полученному сообщению. Это гарантирует, что сообщение на самом деле написано отправителем, и не было изменено в пути. Популярные коммерческие программы используют алгоритмы RC2, RC4, или RC5 фирмы RSA.

Защита от перехвата. От него можно защититься с помощью шифрования содержимого сообщения или канала, по которому он передаётся. Если канал связи зашифрован, то системные администраторы на обоих его концах все-таки могут читать или изменять сообщения. Было предложено много различных схем шифрования электронной почты, но ни одна из них не стала массовой. Одним из самых популярных приложений является PGP (*Pretty Good Privacy* - компьютерная программа, также библиотека функций, позволяющая выполнять операции шифрования и цифровой подписи сообщений, файлов и другой информации, представленной в электронном виде). В прошлом использование PGP было проблематичным, так как в ней использовалось шифрование, подпадавшее под запрет на экспорт из США. Коммерческая версия PGP включает в себя независимо компилируемые программные модули, динамически подключаемые к основным популярным почтовым программам, что делает её особенно удобной для включения в письмо электронной подписи и шифрования письма клиентом. Последние версии PGP используют лицензированную версию алгоритма шифрования с открытыми ключами RSA.

PGP использует алгоритмы шифрования RSA, IDEA и MD5. PGP поддерживает компрессию, передаваемых данных, их секретность, электронную подпись и средства управления доступа к ключам. Схема работы PGP показана на 6.2, где D_A, D_B - личные (закрытые) ключи A и B соответственно, а E_A, E_B - их открытые ключи. Отметим, что секретный ключ для IDEA строится автоматически по ходу работы PGP на стороне A и называется ключом сессии - K_M, который затем шифруется алгоритмом RSA с открытым ключом пользователя B. Так же следует обратить внимание на то, что медленный алгоритм RSA используется для шифро-

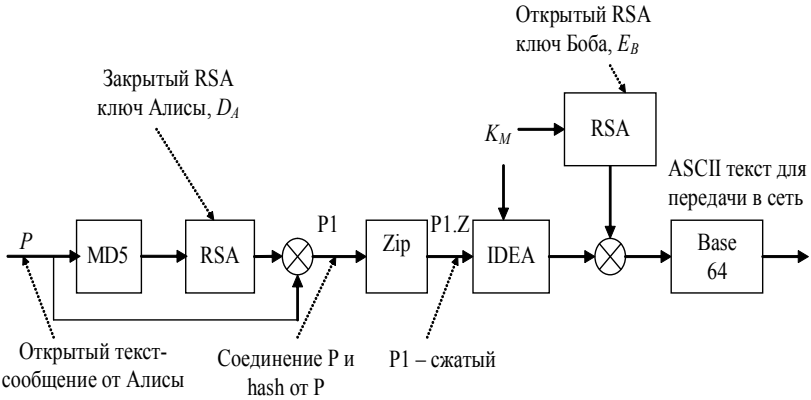

Рис. 6.2. Схема работы программы PGP

вания коротких фрагментов текста: 128 бит MD5 и 128 бит IDEA ключа.

Защита от почтовых бомб. Для защиты от атак типа «почтовые бомбы» используются межсетевые экраны, предписывающие пропускать или запрещать электронные сообщения, поступающие с определённых адресов.

Защита от вредоносных программ. Для устранения угроз от вредоносного программ защита электронной почты использует надежные антивирусные программы, позволяющие сканировать почтовые сообщения.

Защита от фишинговых ссылок. Для противодействия угрозам, связанным с фишинговыми ссылками служба электронной почты использует программы, позволяющие сканировать почтовые сообщения и оповещать о наличии фишинговых ссылок в полученном письме.

Защита от спама. В целях снижения потока спама защита электронной почты использует специальные программные фильтры и антиспам-системы.

Одним часто используемым средством защиты, применяемым некоторыми пользователями Usenet, является конфигурирование своих клиентов для чтения новостей таким образом, что в поле Reply-To (обратный адрес) письма, посылаемого ими в группу новостей, помещается фальшивый адрес, а реальный адрес помещается в сигнатуре или в теле сообщения. Таким образом программы сбора почтовых адресов, собирающие адреса из поля Reply-To, окажутся бесполезными.

Защита писем, почтовых серверов и программ должна соответствовать важности информации, передаваемой по корпоративным сетям. Как правило, должно осуществляться централизованное управление сервисами электронной почты. Должна быть разработана политика, в которой указывался бы нужный уровень защиты.

Примеры политик безопасности для электронной почты

Низкий риск

Пользователь. Использование служб электронной почты для целей, явно противоречащих интересам организации или противоречащих политикам безопасности организации - явно запрещено, так же как и чрезмерное использование её в личных целях. Использование адресов организации в письмах-пирамидах запрещено.

Организация предоставляет своим сотрудникам электронную почту для выполнения ими своих обязанностей. Ограниченное использование её в личных целях разрешается, если оно не угрожает организации.

Использование электронной почты таким образом, что это помогает получать личную коммерческую выгоду, запрещено.

Менеджер. Все сотрудники должны иметь адреса электронной почты. Справочники электронных адресов должны быть доступны для общего доступа.

Если организация обеспечивает доступ к электронной почте внешних пользователей, таких как консультанты, контрактные служащие или партнеры, они должны прочитать политику доступа к электронной почте и расписаться за это.

Содержимое почтовых сообщений считается конфиденциальным, за исключением случая проведения расследований органами внутренних дел.

Сотрудник отдела автоматизации. POP-сервер должен быть сконфигурирован так, чтобы исключать использование незашифрованных паролей с локальных машин.

Средний риск

Пользователь. Электронная почта предоставляется сотрудникам организации только для выполнения ими своих служебных обязанностей. Использование её в личных целях запрещено.

Конфиденциальная информация или информация, являющаяся собственностью организации, не может быть послана с помощью электронной почты. Могут использоваться только утвержденные почтовые программы.Нельзя устанавливать анонимные ремэйлеры.

Служащим запрещено использовать анонимные ремэйлеры.

Менеджер. Конфиденциальная информация или информация, являющаяся собственностью организации, не может быть послана с помощью электронной почты.

Если будет установлено, что сотрудник неправильно использует электронную почту с умыслом, он будет наказан.

Сотрудник отдела автоматизации

Почтовая система должна обеспечивать только один внешний электронный адрес для каждого сотрудника. Этот адрес не должен содержать имени внутренней системы или должности.

Должен вестись локальный архив MIME-совместимых программ для просмотра специальных форматов и быть доступен для внутреннего использования.

Высокий риск

Пользователь. Электронная почта предоставляется сотрудникам организации только для выполнения своих служебных обязанностей. Использование её в личных целях запрещено.

Все электронные письма, создаваемые и хранимые на компьютерах организации, являются собственностью организации и не считаются персональными.

Организация оставляет за собой право получить доступ к электронной почте сотрудников, если на то будут веские причины. Содержимое электронного письма не может быть раскрыто, кроме как с целью обеспечения безопасности или по требованию правоохранительных органов.

Пользователи не должны позволять кому-либо посылать письма, используя их идентификаторы. Это касается их начальников, секретарей, ассистентов или других сослуживцев.

Организация оставляет за собой право осуществлять наблюдение за почтовыми отправлениями сотрудников. Электронные письма могут быть прочитаны организацией, даже если они были удалены и отправителем, и получателем. Такие сообщения могут использоваться для обоснования наказания.

Менеджер. Справочники электронных адресов сотрудников не могут быть сделаны доступными всем.

Если с помощью электронного письма должна быть послана конфиденциальная информация или информация, являющаяся собственностью организации, она должна быть зашифрована так, чтобы её мог прочитать только тот, кому она предназначена, с использованием утверждённых в организации программ и алгоритмов.

Никто из посетителей, контрактников или временных служащих не имеет права использовать электронную почту организации.

Должно использоваться шифрование все информации, классифицированной как критическая или коммерческая тайна, при передаче её через открытые сети, такие как Internet.

Выходящие сообщения могут быть выборочно проверены, чтобы гарантировать соблюдение политики.

Сотрудник отдела автоматизации. Входящие письма должны проверяться на вирусы или другие РПС.

Почтовые серверы должны быть сконфигурированы так, чтобы отвергать письма, адресованные не на компьютеры организации.

Журналы почтовых серверов должны проверяться на предмет выявления использования неутвержденных почтовых клиентов сотрудниками организации, и о таких случаях должно докладываться.

Почтовые клиенты должны быть сконфигурированы так, чтобы каждое сообщение подписывалось с помощью цифровой подписи отправителя.

Выводы

Структура электронного сообщения состоит из двух частей: заголовка письма, в котором указывается служебная информация и тела письма, в котором находится текст письма. Системой доставки электронной почты используется заголовок, так как в нем содержится необходимая информация о сообщении: адрес назначения, приоритет, секретность, требование об уведомлении и т.д.

Основными элементами службы электронной почты являются: *почтовый сервер*, обеспечивающий приём и передачу электронных писем пользователей, *почтовый ящик* для хранения сообщений электронной почты, *адрес* почтового ящика пользователя, *почтовый клиент* для работы с почтовыми серверами, *почтовый протокол* для пересылки почтового сообщения от клиента на почтовый сервер и загрузки почты с почтового сервера адресату.

Распространённые на настоящий момент протоколы электронной почты - SMTP POP3, IMAP, MAPI. Протокол SMTP необходим для передачи сообщения от пользователя на почтовый сервер. Протоколы POP и IMAP - для загрузки (приёма) электронной почты с сервера хранения на машину пользователя.

Основными угрозами безопасности электронной почты являются: случайные ошибки, фальшивые адреса отправителя, перехват письма, почтовые бомбы, вредоносные программы, фишинговые ссылки, спам.

Безопасность электронной почты обеспечивается следующими мероприятиями: не допускать случайные ошибки, использовать технологию электронных подписей, шифрования сообщений или канала, межсетевые экраны, антивирусные и программы, сканирующие сообщение, программные фильтры и антиспам-системы.

Защита писем, почтовых серверов и программ должна соответствовать важности информации, передаваемой по корпоративным сетям. Должна быть разработана политика, в которой указывался бы нужный уровень защиты.

ЗАКЛЮЧЕНИЕ

Предотвращать необходимо не только несанкционированный доступ к информации с целью её раскрытия или нарушения её целостности, но и попытки проникновения с целью нарушения работоспособности этих систем. Защищать необходимо все компоненты систем: оборудование, программы, данные и персонал.

Все усилия по обеспечению внутренней безопасности систем должны фокусироваться на создании надежных и удобных механизмов принуждения всех её законных пользователей и обслуживающего персонала к безусловному соблюдению требований политики безопасности, т.е. установленной в организации дисциплины прямого или косвенного доступа к ресурсам и информации.

Одним из важнейших аспектов проблемы обеспечения безопасности компьютерных систем является выявление, анализ и классификация возможных путей реализации угроз безопасности, т.е. возможных каналов несанкционированного доступа к системе с целью нарушения её работоспособности или доступа к критической информации, а также оценка реальности реализации угроз безопасности и наносимого при этом ущерба.

Все известные меры защиты компьютерных систем подразделяются на: законодательные, морально-этические, административные, физические и технические (аппаратурные и программные). Все они имеют свои достоинства и недостатки.

Наилучшие результаты достигаются при системном подходе к вопросам безопасности компьютерных систем и комплексном использовании различных методов и средств их защиты на всех этапах жизненного цикла систем.

Основными универсальными механизмами противодействия угрозам безопасности, реализуемыми в конкретных средствах защиты, являются:

- идентификация (именование и опознавание), аутентификация (подтверждение подлинности) и авторизация (присвоение полномочий) субъектов;
- контроль (разграничение) доступа к ресурсам системы;
- регистрация и анализ событий, происходящих в системе;
- контроль целостности ресурсов системы.

ЛИТЕРАТУРА

1. Конеев И.Р., Беляев А.В.. Информационная безопасность предприятия. – СПб: БХВ-Санкт-Петербург, 2003.
2. Вертузаев М.С., Юрченко О.М.. Защита информации в компьютерных системах от несанкционированного доступа. – М.: ДМК Пресс, 2002.
3. Петраков А.В. Основы практической защиты информации. Учебное пособие для вузов. – М.: Радио и связь, 2001.
4. Лукацкий А.. Обнаружение атак. –СПб.: БХВ-Санкт-Петербург, 2001.
5. Домарев В.В. Защита информации и безопасность компьютерных систем. – Киев: “DiaSoft”, 1999.
6. Соколов А.В., Шаньгин В.Ф. Защита информации в распределенных корпоративных сетях и системах. – М.: ДМК Пресс, 2002.
7. Ховард М., Лебланк Д. Защищенный код. /Пер. с англ. – М.: Издательско-торговый дом «Русская редакция», 2003.
8. Медведковский И.Д., Семьянов Б.В., Леонов Д.Г., Лукацкий А.В. Атака из Internet, 2002.
9. Зима В., Молдовян А., Молдовян Н. Безопасность глобальных сетевых технологий. – СПб: БХВ-Санкт-Петербург, 2002.

СОДЕРЖАНИЕ

CONTENTS

Учебное издание

Татьяна Михайловна Татарникова

ЗАЩИЩЁННЫЕ КОРПОРАТИВНЫЕ СЕТИ

Раздел: «Задачи по защите информации»

Учебное пособие

Редактор *О.С. Крайнова*
Компьютерная вёрстка *К.П. Ерёмин*

ЛР № 020309 от 30.12.96

Подписано в печать 21.08.2012. Формат 60x90 1/16. Гарнитура “Таймс”
Печать цифровая. Усл. печ. л. 7,1. Тираж 300 экз. Заказ № 112
РГГМУ, 195196, Санкт-Петербург, Малоохтинский пр. 98.
Отпечатано в ЦОП РГГМУ