РАЗДЕЛ 5. СРЕДСТВА ОПЕРАЦИОННЫХ СИСТЕМ ДЛЯ РАБОТЫ С КОМПЬЮТЕРНЫМИ СЕТЯМИ

Тема 15. Сетевые операционные системы

Операционную систему компьютера часто определяют как взаимосвязанный набор системных программ, который обеспечивает эффективное управление ресурсами компьютера (памятью, процессором, внешними устройствами, файлами и др.), а также предоставляет пользователю удобный интерфейс для работы с аппаратурой компьютера и разработки приложений. Говоря о сетевых ОС, мы, очевидно, должны расширить границы управляемых ресурсов за пределы одного компьютера.

Сетевой операционной системой (ОС) называют операционную систему компьютера, которая помимо управления локальными ресурсами предоставляет пользователям и приложениям возможность эффективного и удобного доступа к информационным и аппаратным ресурсам других компьютеров сети.

Сегодня практически все операционные системы являются сетевыми.

В сетевых ОС удаленный доступ к сетевым ресурсам обеспечивается:

- сетевыми службами;
- средствами транспортировки сообщений по сети (в простейшем случае — сетевыми интерфейсными картами и их драйверами).

Функции сетевых ОС:

- управление каталогами и файлами;
- управление ресурсами;
- коммуникационные функции;
- защита от несанкционированного доступа;
- обеспечение отказоустойчивости;
- управление сетью.

Управление каталогами и файлами является одной из первоочередных функций сетевой операционной системы, обслуживаемых специальной сетевой файловой подсистемой. Пользователь получает от этой подсистемы возможность обращаться к файлам, физически расположенным в сервере или в другой станции данных, применяя привычные для локальной работы языковые средства. При обмене файлами должен быть обеспечен необходимый уровень конфиденциальности обмена (секретности данных).

Управление ресурсами включает запросы и предоставление ресурсов.

Коммуникационные функции обеспечивают адресацию, буферизацию, маршрутизацию.

Защита от несанкционированного доступа возможна на любом из следующих уровней: ограничение доступа в определенное время, и (или) для определенных станций, и (или) определенное число раз; ограничение совокупности доступных конкретному пользователю директорий; ограничение для конкретного пользователя списка возможных действий (например, только чтение файлов); пометка файлов символами типа «только чтение», «скрытность при просмотре списка файлов».

Отказоустойчивость определяется наличием в сети автономного источника питания, отображением или дублированием информации в дисковых накопителях. Отображение заключается в хранении двух копий данных на двух дисках, подключенных к одному контроллеру, а дублирование означает подключение каждого из этих двух дисков к разным контроллерам. Сетевая ОС, реализующая дублирование дисков, обеспечивает более высокий уровень отказоустойчивости.

Дальнейшее повышение отказоустойчивости связано с дублированием серверов.

Структура сетевой ОС

Сетевая операционная система составляет основу любой вычислительной сети. Каждый компьютер в сети в значительной степени автономен, поэтому под сетевой операционной системой в широком смысле понимается совокупность операционных систем отдельных компьютеров, взаимодействующих с целью обмена сообщениями и разделения ресурсов по единым правилам - протоколам. В узком смысле сетевая ОС - это операционная система отдельного компьютера, обеспечивающая ему возможность работать в сети.

Рисунок 46 – Структура сетевой ОС

В сетевой операционной системе отдельной машины можно выделить несколько частей (рис. 46):

- средства управления локальными ресурсами компьютера: функции распределения оперативной памяти между процессами, планирования и диспетчеризации процессов, управления процессорами в мультипроцессорных машинах, управления периферийными устройствами и другие функции управления ресурсами локальных ОС.

- средства предоставления собственных ресурсов и услуг в общее пользование - серверная часть ОС (сервер). Эти средства обеспечивают, например, блокировку файлов и записей, что необходимо для их совместного использования; ведение справочников имен сетевых ресурсов; обработку запросов удаленного доступа к собственной файловой системе и базе данных; управление очередями запросов удаленных пользователей к своим периферийным устройствам.

- средства запроса доступа к удаленным ресурсам и услугам и их использования - клиентская часть ОС (редиректор). Эта часть выполняет распознавание и перенаправление в сеть запросов к удаленным ресурсам от приложений и пользователей, при этом запрос поступает от приложения в локальной форме, а передается в сеть в другой форме, соответствующей требованиям сервера. Клиентская часть также осуществляет прием ответов от серверов и преобразование их в локальный формат, так что для приложения выполнение локальных и удаленных запросов неразличимо.

- коммуникационные средства ОС, с помощью которых происходит обмен сообщениями в сети. Эта часть обеспечивает адресацию и буферизацию сообщений, выбор маршрута передачи сообщения по сети, надежность передачи и т.п., то есть является средством транспортировки сообщений.

В зависимости от функций, возлагаемых на конкретный компьютер, в его операционной системе может отсутствовать либо клиентская, либо серверная части.

На рисунке 47 показано взаимодействие сетевых компонентов. Здесь компьютер 1 выполняет роль "чистого" клиента, а компьютер 2 - роль "чистого" сервера, соответственно на первой машине отсутствует серверная часть, а на второй - клиентская. На рисунке отдельно показан компонент клиентской части - редиректор. Именно редиректор перехватывает все запросы, поступающие от приложений, и анализирует их. Если выдан запрос к ресурсу данного компьютера, то он переадресовывается соответствующей подсистеме локальной ОС, если же это запрос к удаленному ресурсу, то он переправляется в сеть. При этом клиентская часть преобразует запрос из локальной формы в сетевой формат и передает его транспортной подсистеме, которая отвечает за доставку сообщений указанному серверу. Серверная часть операционной системы компьютера 2 принимает запрос, преобразует его и передает для выполнения своей локальной ОС. После того, как результат получен, сервер обращается к транспортной подсистеме и направляет ответ клиенту, выдавшему запрос. Клиентская часть преобразует результат в соответствующий формат и адресует его тому приложению, которое выдало запрос.

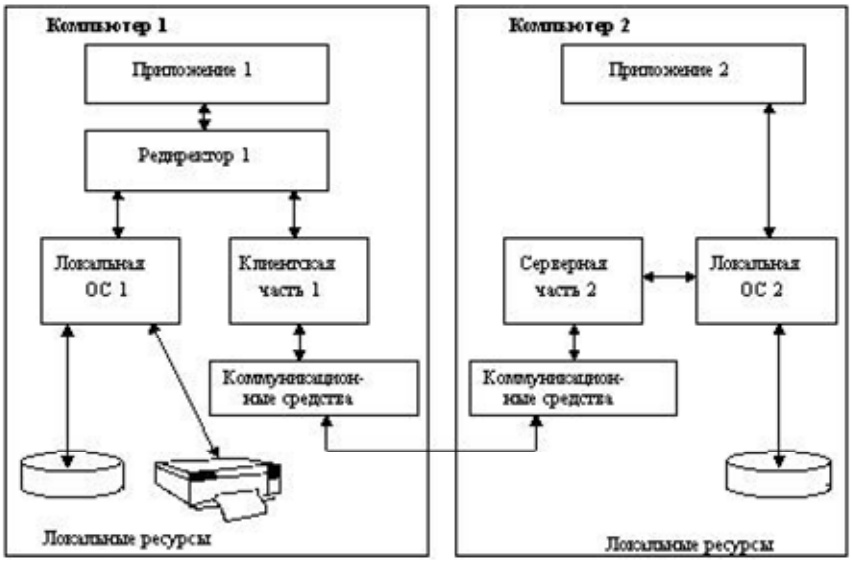


Рисунок 47 – Взаимодействие компонентов операционной системы при взаимодействии компьютеров

На практике сложилось несколько подходов к построению сетевых операционных систем (рис. 48).

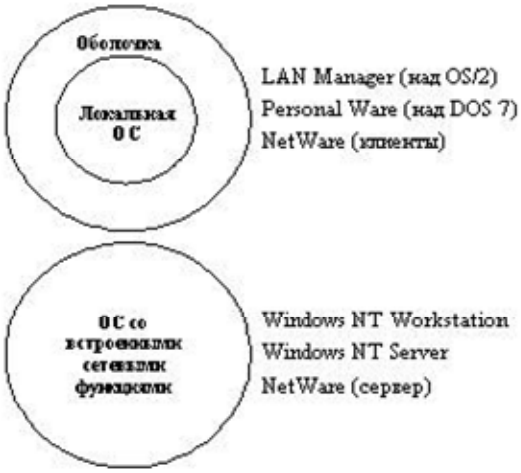


Рисунок 48 – Варианты построения сетевых ОС

Первые сетевые ОС представляли собой совокупность существующей локальной ОС и надстроенной над ней сетевой оболочки. При этом в локальную ОС встраивался минимум сетевых функций, необходимых для работы сетевой оболочки, которая выполняла основные сетевые функции. Примером такого подхода является использование на каждой машине сети операционной системы MS DOS (у которой начиная с ее третьей версии появились такие встроенные функции, как блокировка файлов и записей, необходимые для совместного доступа к файлам).

Принцип построения сетевых ОС в виде сетевой оболочки над локальной ОС используется и в современных ОС, таких, например, как LANtastic или Personal Ware.

Однако более эффективным представляется путь разработки операционных систем, изначально предназначенных для работы в сети. Сетевые функции у ОС такого типа глубоко встроены в основные модули системы, что обеспечивает их логическую стройность, простоту эксплуатации и модификации, а также высокую производительность. Примером такой ОС является система Windows NT фирмы Microsoft, которая за счет встроенности сетевых средств обеспечивает более высокие показатели производительности и защищенности информации по сравнению с сетевой ОС LAN Manager той же фирмы (совместная разработка с IBM), являющейся надстройкой над локальной операционной системой OS/2.

Виды сетевых ОС

Сетевая служба может быть представлена в ОС либо обеими (клиентской и серверной) частями, либо только одной из них.

В нервом случае операционная система, называемая одноранговой, не только позволяет обращаться к ресурсам других компьютеров, но и предоставляет собственные ресурсы в распоряжение пользователей других компьютеров. Например, если на всех компьютерах сети установлены и клиенты, и серверы файловой службы, то все пользователи сети могут совместно применять файлы друг друга. Компьютеры, совмещающие функции клиента и сервера, называют одноранговыми узлами.

Операционная система, которая преимущественно содержит клиентские части сетевых служб, называется клиентской. Клиентские ОС устанавливаются на компьютеры, обращающиеся с запросами к ресурсам других компьютеров сети. За такими компьютерами, также называемыми клиентскими, работают рядовые пользователи. Обычно клиентские компьютеры относятся к классу относительно простых устройств.

К другому типу операционных систем относится серверная ОС — она ориентирована на обработку запросов из сети к ресурсам своего компьютера и включает в себя в основном серверные части сетевых служб. Компьютер с установленной на нем серверной ОС, занимающийся исключительно обслуживанием запросов других компьютеров, называют выделенным сервером сети. За выделенным сервером, как правило, обычные пользователи не работают.

Примеры сетевых ОС

Сегодня практически все ОС являются сетевыми. Наиболее расропстраненные из них:

- Novell NetWare
- Microsoft Windows (95, NT, XP, Vista, Seven)
- Различные UNIX системы, такие как Solaris, FreeBSD
- Различные GNU/Linux системы

– IOS
– ZyNOS компании ZyXEL
– Chrome OS от Google

Операционные системы мейнфреймов

К высшей категории относятся операционные системы мейнфреймов (больших универсальных машин) — компьютеров, занимающих целые залы и до сих пор еще встречающихся в крупных центрах обработки корпоративных данных. Такие компьютеры отличаются от персональных компьютеров по объемам ввода-вывода данных. Мейнфреймы, имеющие тысячи дисков и петабайты данных — весьма обычное явление, а персональный компьютер с таким арсеналом стал бы предметом зависти. Мейнфреймы также находят применение в качестве мощных веб-серверов, серверов крупных интернет-магазинов и серверов, занимающихся межкорпоративными транзакциями.

Операционные системы мейнфреймов ориентированы преимущественно на одновременную обработку множества заданий, большинство из которых требует колоссальных объемов ввода-вывода данных. Обычно они предлагают три вида обслуживания: пакетную обработку, обработку транзакций и работу в режиме разделения времени. Пакетная обработка — это одна из систем обработки стандартных заданий без участия пользователей. В пакетном режиме осуществляется обработка исков в страховых компаниях или отчетов о продажах сети магазинов. Системы обработки транзакций справляются с большим количеством мелких запросов, к примеру обработками чеков в банках или бронированием авиабилетов. Каждая элементарная операция невелика по объему, но система может справляться с сотнями и тысячами операций в секунду. Работа в режиме разделения времени дает возможность множеству удаленных пользователей одновременно запускать на компьютере свои задания, например запросы к большой базе данных. Все эти функции тесно связаны друг с другом, и зачастую операционные системы универсальных машин выполняют их в комплексе. Примером операционной системы универсальных машин может послужить OS/390, наследница OS/360. Однако эти операционные системы постепенно вытесняются вариантами операционной системы UNIX, например Linux.

Тема 16. Команды ОС Windows тестирования сетевых интерфейсов

Существует несколько инструментов для отслеживания и решения проблем, связанных с применением протокола TCP/IP. Этими инструментами являются PING, ARP, IPCONFIG, TRACERT, NBTSTAT и PATHPING. Все они запускаются из командной строки и выдают результаты в формате DOS. В таблице 8.1 перечислены эти инструменты и дано их краткие описания.

PING

Подобно гидролокатору на подводной лодке, команда PING позволяет получать информацию о своих соседях. Правда, тут она применяется в сугубо мирных целях. Она может сообщить вам о том, как долго информационные пакеты идут из вашего компьютера на принимающий компьютер. Она делает это посредством отправки ICMP эхо-сигнала указанному устройству - будь то устройство локальной сети или сервер на другой стороне земного шара.

Таблица 9 – Инструменты для решения проблем протокола TCP/IP

Инструмент командной строки	**Описание**
ARP	Позволяет модифицировать таблицу протокола разрешения адресов.
IPCONFIG	Показывает текущую TCP/IP конфигурацию и позволяет обновлять эти значения.
NBTSTAT	Предоставляет NetBIOS-информацию о TCP/IP-соединениях, перезагружает кэш LMHost и определяет зарегистрированное имя и область действия ID.
PING	Посылает эхо-запрос на указанное устройство.
TRACERT	Перечисляет количество переходов (изменений маршрута) до указанного устройства.
PATHPING	Показывает степень потери информационных пакетов на любом маршрутизаторе или ссылке.

Если вы тестируете пинг-запросом устройство своей локальной сети, то устройство откликнется практически мгновенно. В этом случае вы узнаете, что оба компьютера работают нормально. При возникновении проблем следует выполнить следующие шаги.

1. Протестируйте пингом-запросом адрес локальной перемычки. Если этот адрес ответит, то на локальном компьютере имеется конфигурация протокола TCP/IP.

Ping 127.0.0.1

2. Протестируйте локальный IP-адрес и убедитесь, что нет конкуренции с другим устройством в сети.

Ping IP_адрес

3. Протестируйте IP-адрес шлюза по умолчанию. Так вы проверите возможность добраться до ближайшего маршрутизатора, который позволяет общаться с компьютерами в другой подсети.

Ping IP_адрес шлюза

4. Протестируйте пингом-запросом адрес указанного вами устройства в другой подсети. Так вы проверите возможность установки связи с устройством другой подсети.

Ping IP_адрес узла

5. Протестируйте пингом-запросом то же самое устройство, применив полное имя его домена. Если попытка закончится провалом, но шаг 4 работает, то это проблема разрешения имени. На этом этапе следует убедиться, что DNS-серверы доступны, таблицы Hosts и LMHosts точны, а WINS (если используется) правильно сконфигурирован.

Ping IP_имя узла

Инструмент PING используется следующим образом:

Ping [-t] [-a] [-n] [-l] [-f] [-I TTL] [-v TOS] [-r] [-s] [-j список узлов] [-k список узлов] [-w] список адресатов

Аргументы PING включают в себя следующее.

- -t Поддерживает пингование, пока не будет остановлен нажатием клавиш CTRL+C.
- -n Посылает эхо-сигнал определенное (указанное) количество раз и прекращает тестирование.
- -l Посылает пакет с указанным количеством битов.
- -f Устанавливает флаг Don't Fragment (Не фрагментировать). Это значит, что пакеты не будут разбиваться на части сетевыми устройствами.
- -w Устанавливает время простоя (мс). Время простоя по умолчанию равно 750 мс.

ARP

Протокол разрешения адресов (Address Resolution Protocol, ARP) позволяет компьютерам создавать соединения на физическом уровне. Независимо от того, используете ли вы NetBIOS или TCP/IP имена компьютеров в своей сети, они должны быть конвертированы в MAC-имена сетевой карты компьютера. Когда одна рабочая станция пытается установить связь с другой, она должна транслировать сигнал в соответствии с протоколом ARP, чтобы выяснить MAC-адрес. После того как Windows XP Professional компьютер определит MAC-адрес, он использует его для установки связи с устройством. Эта конверсия IP в MAC хранится в ARP-таблице компьютера.

Команда ARP позволяет просматривать и редактировать таблицу ARP. Этот инструмент полезен при решении проблем, связанных с разрешениями имен. Команда ARP записывается следующим образом.

ARP -s inet_addr eth_addr [if_addr]

ARP -d inet_addr [if_addr]

ARP -a [inet_addr] [-N if_addr]

В приведенных примерах атрибуты работают следующим образом.

- -s Добавляет IP-адрес (inet_addr) или Ethernet MAC адрес (eth_addr) в таблицу ARP. IP-адрес имеет стандартный четырехоктетный формат, в то время как Ethernet-адрес записывается шестью шестнадцатеричными значениями, разделенными тире.
- -d Удаляет указанный IP-адрес из таблицы.
- -a Выводит на экран текущую ARP-таблицу. Если вы включили в нее IP-адрес, то будет представлена только таблица переводов IP-адреса в MAC-адрес для данного компьютера.

Аргумент [if_addr] указывает IP-адрес, отличный от данного по умолчанию. Если вы хотите посмотреть на таблицу ARP компьютера, которым вы пользуетесь, то введите в командную строку arp-a.

IPCONFIG

Инструмент IPCONFIG хорошо подходит для начала поисков источника проблемы, связанной с применением протокола TCP/IP. Команда записывается следующим образом.

Ipconfig [/all | /release [adapter] | /renew [adapter]]

При использовании без аргументов IPCONFIG представляет только основные настройки TCP/IP, включая IP-адрес, маску подсети и шлюз по умолчанию для каждой карты сетевого адаптера. Однако, добавив аргументы, можно повысить полезность IPCONFIG. Аргументы включают в себя следующее.

- /all Показывает основную и дополнительную информацию, такую как сроки окончания аренды и службы разрешения имен.
- /release Выдает IP-адрес указанному адаптеру, если адаптер использовал DHCP.
- /renew Обновляет IP-адрес для указанного адаптера, если адаптер использовал DHCP.

Примечание. Ввод ipconfig? в командную строку сгенерирует полный список аргументов.

Использование инструмента IPCONFIG может дать огромное количество информации о TCP/IP-соединениях и их конфигурациях. Всегда полезно проверять маску подсети. Убедитесь в том, что она не записана как 0.0.0.0, что указывает на конфликт с другим устройством подсети.

TRACERT

Инструмент Trace Route (TRACERT) применяется для отслеживания перемещения пакета данных от устройства к устройству. Он работает посредством передачи пакета со значением времени жизни (TTL), равным 1. Обычно маршрутизаторы сокращают значение TTL на 1 и затем отправляют пакет дальше по пути следования. Если маршрутизатор получает TTL со значением 0, то он возвращает пакет отправителю как просроченный. Это позволяет узнать кое-что о маршрутизаторе. Инструмент TRACERT выполняет это действие для первого маршрутизатора на пути следования пакета, добавляет 1 к TTL и затем отправляет

новый пакет. Следующий пакет доходит до второго маршрутизатора и становится просроченным. Этот маршрутизатор возвращает пакет вместе с информацией о самом себе. Процесс повторяется, пока пакет не дойдет до нужного устройства, или пока количество переходов не достигнет максимального значения.

Синтаксис команды TRACERT следующий.

Tracert [-d] [-h maximum_hops] [-j host-list] [-w timeout] имя конечного устройства

Некоторые аргументы команды TRACERT описаны ниже.

- -d Препятствует разрешению адреса именам хостов.
- -h maximum_hops Устанавливает верхнюю границу общего числа переходов, необходимых для нахождения нужной рабочей станции.
- -j host-list Устанавливает свободный начальный маршрутизатор для всего списка хостов.
- -w timeout Устанавливает время простоя (мс) для каждого перехода.

Вы можете применять команду TRACERT, просто вводя tracert и адрес конечного устройства.

Этот инструмент полезен, если вы не можете запустить ни одной утилиты из пакета протоколов TCP/IP. После того как вы убедились в том, что TCP/IP установлен, но нельзя использовать команды PING или TRACERT, следует удалить и заново проинсталлировать протокол TCP/IP, который мог повредиться.

NBTSTAT

Инструмент NBTSTAT помогает в решении проблем, связанных с разрешением NetBIOS-имен в TCP/IP-соединениях. Он показывает статистику протокола и текущие TCP/IP-соединения, используя NetBT (NetBIOS поверх TCP/IP). Когда сеть функционирует нормально, NetBT разрешает присваивать NetBIOS-имена IP-адресам.

Команда NBTSTAT имеет следующий синтаксис.

Nbtstat [-a Удаленное имя] [-A IP-адрес] [-c] [-n] [-r] [-R] [-s] [-S] [интервал]

Некоторые аргументы NBTSTAT означают следующее.

- -n Показывает имена, зарегистрированные локально системой, в которой используется сервер или службы переадресации.
- -c Перечисляет переводы имени в IP-адрес, которые находятся в кэше системы.
- -R Заставляет систему очищать кэш и перезагружать его из файла Lmhosts (автоматически перезагружаются только те элементы Lmhosts файла, которые имеют обозначение #PRE).
- -a "имя" Возвращает таблицу NetBIOS-имен компьютера, а также MAC-адрес его сетевой карты.
- -s Перечисляет текущие NetBIOS-сессии, их статус и основные статистические данные.

Примечание. Для получения более подробной информации о NBTSTAT введите nbtstat? в окне команд.

PATHPING

Инструмент PATHPING является комбинацией инструментов PING и TRACERT. Этот инструмент в упорядоченном режиме посылает информационные пакеты на каждый маршрутизатор по пути к месту назначения. Затем он рассчитывает результаты на основании пакетов, возвращенных каждым маршрутизатором. Так как PATHPING показывает степень потери пакетов в любом маршрутизаторе или соединении, администратор может определить, какие именно маршрутизаторы и соединения вызывают проблемы в работе сети.

Команда PATHPING записывается следующим образом.

Pathping [-n] [-h maximum_hops] [-g host-list] [-p period] [-q num_queries] [-w timeout] [-T] [-R] target_name

Некоторые аргументы PATHPING включают в себя следующее.

- -n Не разрешает присваивать адреса именам хостов.
- -h maximum_hops Указывает максимальное количество изменений маршрута, необходимое для нахождения конечного пункта. Настройка по умолчанию предусматривает 30 переходов.
- -p period Указывает время (мс) между двумя передачами пинг-сигнала. По умолчанию равно 250 мс.
- -q num_queries Указывает количество запросов, посланных на каждый компьютер во время прохождения маршрута. Значение по умолчанию - 100.
- -w timeout Указывает время (мс), отводимое на ожидание ответа. По умолчанию - 3000 мс (или 3 с).

Тема 17. Команды ОС Unix конфигурирования и тестирования сетевых интерфейсов

Настройка сетевых интерфейсов

Интерфейсом с точки зрения ОС является устройство, через которое система получает и передает IP-пакеты. Роль интерфейса локальной сети может выполнять одно (или несколько) из следующих устройств: Ethernet-карта, ISDN-адаптер или модем, подключенный к последовательному порту. Каждое устройство (не весь компьютер!) имеет свой IP-адрес. Для выхода в локальные сети используется, как правило, Ethernet-карта, что и будет предполагаться в настоящем разделе.

Расположение конфигурационных файлов

Отметим сразу, что все приводимые ниже команды можно выполнять из командной строки, но тогда придется повторять эти операции при каждом перезапуске компьютера. Поэтому может быть удобнее записать их в один из

инициализационных файлов, автоматически запускаемых при старте системы. В разных дистрибутивах процесс загрузки организован по-разному. В "Linux NET-3-HOWTO" приводится следующая таблица:

Таблица 10 – Расположение конфигурационных файлов в основных дистрибутивах

Дистрибутив	Настройка интерфейса и маршрутизации	Запуск демонов
Debian	/etc/init.d/network	/etc/init.d/netbase /etc/init.d/netstd_init /etc/init.d/netstd_nfs /etc/init.d/netstd_misc
Slackware	/etc/rc.d/rc.inet1	/etc/rc.d/rc.inet2
RedHat	/etc/sysconfig/network-scripts/ifup-<ifname>	/etc/rc.d/init.d/network

Обратите внимание, что дистрибутивы Debian и Red Hat содержат отдельный каталог для скриптов запуска системных сервисов (хотя сами файлы настроек находятся в других местах, например, в дистрибутиве Red Hat они хранятся в каталоге /etc/sysconfig). Для понимания процесса загрузки ознакомьтесь с содержимым файла /etc/inittab и документацией по процессу init.

Команда ifconfig

После подключения драйверов вы должны настроить те интерфейсы, которые вы предполагаете использовать. Настройка интерфейса заключается в присвоении IP-адресов сетевому устройству и установке нужных значений для других параметров сетевого подключения. Наиболее часто для этого используется программа ifconfig (ее название происходит от "interface configuration").

Запустите ее без аргументов (или с единственным аргументом -a) и вы узнаете, какие параметры установлены в данный момент для активных сетевых интерфейсов (в частности, для сетевой карты). Кстати, имеет смысл выполнить эту команду еще до подключения модулей: а вдруг у вас поддержка интерфейсов встроена в ядро и необходимые настройки сделаны в процессе инсталляции системы. Тогда вы в ответ можете получить информацию о параметрах вашей Ethernet-карты и так называемого "кольцевого интерфейса" или "обратной петли" - Local Loopback (интерфейс Ethernet при единственной сетевой карте обозначается как eth0, а кольцевой интерфейс - как lo). Если же по этой команде вы ничего не получите, то надо переходить к подключению модулей и настройке, и начинать надо с кольцевого интерфейса.

Настройка локального интерфейса lo

Этот интерфейс используется для связи программ IP-клиентов с IP-серверами, запущенными на той же машине, так что его необходимо настроить даже в том случае, если вы вообще не подключаете никаких сетевых устройств.

Локальный интерфейс настраивается очень просто: командой

```
[root]# /sbin/ifconfig lo 127.0.0.1
```

Теперь, чтобы проверить работоспособность протоколов TCP/IP на вашей машине, дайте команду:

```
[root]# ping 127.0.0.1
```

Настройка интерфейса платы Ethernet локальной сети (eth0)

Для того чтобы ваш компьютер вошел в сеть с IP-адресом, полученным вами у администратора (пусть для примера это будет адрес 192.168.0.15), вы должны запустить команду ifconfig примерно следующим образом:

```
[root]# /sbin/ifconfig eth0 192.168.0.15 netmask
255.255.255.0 up
```

Если не указывать маску подсети, то по умолчанию устанавливается маска подсети 255.0.0.0.

В некоторых случаях необходимо бывает изменить адрес прерывания, используемого сетевой картой, порта ввода-вывода или типа соединения, используемого в сети. Это можно сделать, выполнив следующую команду:

```
root# /sbin/ifconfig eth0 irq 5 io_addr 220 media
10baseT
```

Не все устройства (платы) поддерживают динамическое изменение этих параметров (т. е. может потребоваться переустановить переключатели на плате).

Интерфейс для последовательного порта

Последовательный порт используется для подключения модема, через который осуществляется соединение с сетью по телефонной линии. Для настройки интерфейса этого типа тоже можно использовать программу ifconfig. Однако, такие программы как pppd и dip, используемые для соединения с сетью по модему, способны автоматически конфигурировать сетевой интерфейс, поэтому обычно для этого случая применять ifconfig не требуется.

Настройка маршрутизации

Правила маршрутизации определяют, куда отправлять IP-пакеты. Данные маршрутизации хранятся в одной из таблиц ядра. Вести таблицы маршрутизации можно статически или динамически. Статический маршрут - это маршрут, который задается явно с помощью команды route. Динамическая маршрутизация выполняется процессом-демоном (routed или gated), который ведет и модифицирует таблицу маршрутизации на основе сообщений от других компьютеров сети. Для выполнения динамической маршрутизации разработаны специальные протоколы: RIP, OSPF, IGRP, EGP, BGP и т. д.

Динамическая маршрутизация необходима в том случае, если у вас сложная, постоянно меняющаяся структура сети и одна и та же машина может быть доступна по различным интерфейсам (например, через разные Ethernet или SLIP интерфейсы). Маршруты, заданные статически, обычно не меняются, даже если используется динамическая маршрутизация.

Для персонального компьютера, подключаемого к локальной сети, в большинстве ситуаций бывает достаточно статической маршрутизации командой route. Прежде чем пытаться настраивать маршруты, просмотрите таблицу маршрутизации ядра с помощью команды netstat -n -r. Вы должны увидеть что-то вроде следующего

[root]# netstat –nr

Kernel IP routing table

Destination	Gateway	Genmask	Flags	MSS	Window	irtt	Iface
10.72.128.101	0.0.0.0	255.255.255.255	UH	0	0	0	eth0
10.72.128.0	0.0.0.0	255.255.255.0	U	0	0	0	eth0
127.0.0.0	0.0.0.0	255.0.0.0	U	0	0	0	lo
0.0.0.0	10.72.128.254	0.0.0.0	UG	0	0	0	eth0

Если таблица пуста, то вы увидите только заголовки столбцов. Тогда надо использовать route. С помощью команды route можно добавить или удалить один (за один раз) статический маршрут. Вот ее формат:

```
[root]# /sbin/route [-f] операция [-тип] адресат шлюз [dev] интерфейс
```

Здесь аргумент операция может принимать одно из двух значений: add (маршрут добавляется) или delete (маршрут удаляется). Аргумент адресат может быть IP-адресом машины, IP-адресом сети или ключевым словом default . Аргумент шлюз - это IP-адрес компьютера, на который следует пересылать пакет (этот компьютер должен иметь прямую связь с вашим компьютером). Команда

```
[root]# /sbin/route -f
```

удаляет из таблицы данные обо всех шлюзах. Необязательный аргумент тип принимает значения net или host . В первом случае в поле адресата указывается адрес сети, а во втором - адрес конкретного компьютера (хоста).

Как правило, бывает необходимо настроить маршрутизацию по упоминавшимся выше трем интерфейсам:

- локальный интерфейс (lo),
- интерфейс для платы Ethetnet (eth0),
- интерфейс для последовательного порта (PPP или SLIP).

Локальный интерфейс поддерживает сеть с IP-номером 127.0.0.1. Поэтому для маршрутизации пакетов с адресом 127.... используется команда:

```
[root]# /sbin/route add -net 127.0.0.1 lo
```

Если у вас для связи с локальной сетью используется одна плата Ethernet, и все машины находятся в этой сети (сетевая маска 255.255.255.0), то для настройки маршрутизации достаточно вызвать:

```
[root]# /sbin/route add -net 192.168.36.0 netmask 255.255.255.0 eth0
```

Если же вы имеете насколько интерфейсов, то вам надо определиться с сетевой маской и вызвать команду route для каждого интерфейса.

Поскольку очень часто IP-пакеты с вашего компьютера могут отправляться не в одну единственную сеть, а в разные сети (например, при просмотре разных сайтов в Интернете), то в принципе надо было бы задать очень много маршрутов. Очевидно, что сделать это было бы очень сложно, точнее просто невозможно. Поэтому решение проблемы маршрутизации пакетов перекладывают на плечи специальных компьютеров - маршрутизаторов, а на обычных компьютерах задают маршрут по умолчанию, который используется для отправки всех пакетов, не указанных явно в таблице маршрутизации. С помощью маршрута по умолчанию вы говорите ядру "а все остальное отправляй туда". Маршрут по умолчанию настраивается следующей командой:

```
[root]# /sbin/route add default gw 192.168.1.1 eth0
```

Опция gw указывает программе route, что следующий аргумент - это IP-адрес или имя маршрутизатора, на который надо отправлять все пакеты, соответствующие этой строке таблицы маршрутизации.

После настройки маршрутизации можно проверить, что у вас получилось. Для этого снова дайте команду

```
[root]# netstat -nr
```

Если вывод команды выглядит так, как это было показано выше, но не содержит строки, которая в графе Destination содержит 0.0.0.0 , а в графе Gateway указывает на маршрут, используемый для соединений по умолчанию, то вы, вероятно, не задали этот маршрут.

Настройка службы имен

С помощью команды ifconfig вы задали IP-адрес вашего компьютера, но он еще не знает своего имени (при инсталляции системы он получил обезличенное имя localhost). Существует команда hostname, которая позволяет установить (и узнать действующее в данный момент) имя компьютера и имя домена.

Однако установить только имя и только этой командой еще недостаточно, поскольку эта команда меняет имя только на текущий сеанс работы. Поэтому обычно эта команда вызывается в одном из инициализационных файлов, например, /etc/rc.d/rc или /etc/rc.d/rc.local. Вы можете попытаться найти ее там, чтобы изменить должным образом имя компьютера, которое задается в качестве параметра команды hostname. В таком случае требуется перезагрузиться для того чтобы изменения вступили в силу.

Другой способ изменения имени компьютера или домена состоит в том, что эти имена прописываются в файле /etc/sysconfig/network в виде двух строчек примерно следующего вида:

```
HOSTNAME="new_host_name.localdomain.upperdomain"
DOMAINNAME=localdomain.upperdomain
```

Тогда в процессе инициализации системы эти имена будут восстанавливаться, потому что файл /etc/sysconfig/network вызывается из /etc/rc.d/rc.sysinit.

Кроме того, имя компьютера должно быть прописано в файле /etc/hosts, который связывает имя компьютера с его IP-адресом. Каждая строка файла /etc/hosts должна начинаться с IP-адреса, за которым следует имя данного узла. Следом за именем можно записать произвольное число псевдонимов этого узла.

Даже если ваш компьютер не подключен к сети, в файле /etc/hosts должна быть прописана хотя бы одна строка следующего вида.

```
127.0.0.1 localhost localhost.localdomain
```

Если же ваш компьютер подключен к TCP/IP сети, то в этом файле дополнительно нужно прописать строку вида

```
192.168.0.15 host_name host_name.localdomain
```

Файл /etc/hosts используется в механизмах разрешения имен. В больших сетях трудно было бы поддерживать в актуальном состоянии файлы /etc/hosts на всех компьютерах, если бы это был основной инструмент для определения IP-адресов по именам. Поэтому обычно для разрешения имен используются серверы DNS. Однако файл /etc/hosts все равно необходим, хотя бы для обращения к серверу DNS. Поэтому в нем имеет смысл указать IP-адреса и соответствующие имена шлюзов и серверов DNS и NIS. А чтобы все приложения использовали этот файл при разрешении имен, должен иметься файл /etc/hosts.conf, содержащий строку

```
order hosts,bind
```

которая говорит, что при разрешении имен сначала должен использоваться файл /etc/hosts, а затем должно происходить обращение к серверу DNS. В большинстве случаев в файле /etc/hosts.conf достаточно иметь две строки:

```
order hosts,bind
multi on
```

Эти параметры указывают системе преобразования имен, что надо просмотреть файл /etc/hosts перед тем, как посылать запрос к серверу, и что следует возвращать все найденные в /etc/hosts адреса для данного имени, а не только первый.

Но настройка механизма разрешения имен не ограничивается редактированием файлов /etc/hosts и /etc/hosts.conf. Необходимо еще указать компьютеру имена серверов DNS. Они прописываются в файле /etc/resolv.conf. Этот файл имеет весьма простой формат. Это текстовый файл, каждая строка которого задает один из параметров системы преобразования имен. Как правило, используются три ключевых слова-параметра:

- domain - задает имя локального домена.
- search - задает список имен доменов, которые будут добавляться к имени машины, если вы не укажете явно имени домена. Это позволяет ограничить область поиска и избежать некоторых ошибок (например, вы ищете компьютер linux.msk.ru, а механизм разрешения имен выведет вас на linux.spb.ru).

- nameserver - этот параметр, который вы можете указывать несколько раз, задает IP-адрес сервера преобразования имен, на который ваша машина будет посылать запросы. Повторяя этот параметр, вы можете задать несколько серверов.

Если вы не собираетесь заводить поддержку сервиса имен для своей сети (что является довольно сложной организационной и технической проблемой), и доверяете ведение своих имен администратору локальной сети или вашему IP-провайдеру, то вам достаточно задать файл /etc/resolv.conf примерно следующего вида:

```
domain abcd.ru
search abcd.ru xyz.edu.ru
nameserver 192.168.10.1
nameserver 192.168.12.1
```

В этом примере машина находится в домене abcd.ru. Если вы зададите имя машины, не указывая домена, например "pc1" , то система преобразования имен попытается сначала найти машину "pc1.abcd.ru" , а в случае неудачи - "pc1.xyz.edu.ru" . Для преобразования имен ваша машина будет обращаться к серверам по адресам "192.168.10.1" и "192.168.12.1".

Тестирование сетевого соединения

Чтобы проверить, соединяется ли ваш компьютер с сетью, попробуйте дать команду ping, указав ей в качестве параметра IP-адрес одного из компьютеров сети. Пусть, например, вам известно (узнайте реальный номер и имя у администратора сети), что в сети есть компьютер с IP-адресом 192.168.0.2 и именем pc1. Тогда вы должны дать команду:

```
[user]$ ping 192.168.0.2
```

или (тут вы одновременно проверяете и работу службы DNS)

[user]$ ping pc1

Если соединение с сетью установлено, должны появиться и периодически обновляться строчки примерно такого вида:

```
64 bytes from 192.168.0.2: icmp_seq=0 ttl=32 time=1.2 ms
64 bytes from 192.168.0.2: icmp_seq=1 ttl=32 time=1.0 ms
64 bytes from 192.168.0.2: icmp_seq=2 ttl=32 time=1.0 ms
64 bytes from 192.168.0.2: icmp_seq=3 ttl=32 time=1.0 ms
64 bytes from 192.168.0.2: icmp_seq=4 ttl=32 time=1.1 ms
```

Это означает, что сетевое соединение работает. Для того чтобы прервать тестирование сети, нажмите комбинацию клавиш <Ctrl>+<C>.

Программы telnet и rlogin

Для того чтобы воспользоваться программой telnet, вам необходимо знать имя или IP-адрес удаленного компьютера, работающего под управлением ОС типа UNIX, на котором для вас открыт пользовательский бюджет. Предположим для

примера, что на компьютере linux2 имеется пользователь user5, пароль которого вам известен. В таком случае вы можете дать команду

```
[user]$ telnet linux2
```

Если программе удалось подключиться к указанному компьютеру, на экране появится сообщение "Connected to server linux2" и приглашение к входу в систему, как если бы вы сидели за терминалом компьютера linux2. Вводите имя (user5) и пароль, и вы будете работать на этом компьютере.

Команда rlogin может быть использована для выхода на удаленный компьютер вполне аналогично команде telnet, хотя лучше сразу указать в командной строке имя пользователя:

```
[user]$ rlogin -l user5 linux2
```

Завершив работу, не забудьте закрыть сессию (командой exit). После этого программа telnet (или rlogin) докладывает, что сессия закрыта, и вы возвращаетесь к командной строке локальной оболочки.

Когда вы работаете с программой telnet, вы полностью работаете на удаленном компьютере: команды выполняются в его оперативной памяти, вы видите (по команде ls) каталоги и файлы на дисках удаленного компьютера и т. д. Только вывод результатов осуществляется на ваш монитор. В рамках программы telnet невозможно, например, открыть для просмотра файл, расположенный на локальном диске. Ваш компьютер выполняет только роль удаленного терминала. Если же вы хотите организовать обмен файлами между вашим компьютером и удаленным, можно воспользоваться программой ftp.

Программа ftp

Программа ftp - это пользовательский интерфейс к стандартному протоколу передачи файлов по Интернету - File Transfer Protocol. Программа позволяет передавать файлы на удаленный компьютер и получать файлы с удаленного компьютера. Однако, введя команду ftp, вы запускаете только клиентскую программу. Для того чтобы получить доступ к файлам удаленного компьютера, на нем должен быть запущен ftp-сервер. Кроме того, необходимо знать либо имя и пароль пользователя, либо ftp-сервер должен разрешать анонимный доступ. Предположим, что эти условия выполнены и вы запустили программу ftp (без параметров). Вы увидите приглашение интерпретатора команд этой программы:

```
ftp >
```

Если ввести знак вопроса, программа выдаст перечень возможных команд. Первая команда, которую нужно в этом случае ввести, - команда open, после которой надо указать сетевое имя компьютера, на котором запущен ftp-сервер. Если анонимный доступ к этому серверу разрешен, то вы получите запрос на ввод имени и пароля пользователя. По команде pwd можно узнать имя текущего каталога на удаленном компьютере, а по команде dir - вывести список файлов и подкаталогов этого каталога. Команда cd имя_каталога используется для смены текущего каталога на удаленном компьютере.

В любой момент вы можете повторно ввести команду ? или ее эквивалент help , чтобы получить подсказку по возможным командам. Для получения более подробной подсказки по конкретной команде надо ввести имя интересующей вас команды после help или ?, например, так:

```
ftp > help dir
```

Если вы хотите выполнить какую-то команду на локальном компьютере (например, выяснить имя текущего каталога), надо дать соответствующую команду, перед которой поставить восклицательный знак:

```
ftp >! pwd
```

! - это команда интерпретатора, вызывающая новый экземпляр оболочки shell локального компьютера. Первый аргумент, следующий за !, должен быть командой оболочки, а все остальные аргументы - аргументами вызываемой команды. Для смены текущего каталога на локальном компьютере имеется специальная команда lcd (очень полезная, поскольку часто до запуска ftp забываешь перейти в тот каталог, куда хочешь скопировать файл с удаленного компьютера; не выходить же из-за этого из программы ftp).

Для пересылки файла на удаленный компьютер используется команда

```
ftp > put имя_файла
```

(или ее синоним send), а для копирования файла с удаленного компьютера в текущий каталог на локальном диске - команда

```
ftp > get имя_файла
```

В принципе этих двух команд вполне достаточно для организации обмена файлами с удаленным компьютером, но как же неудобно ими пользоваться! Приходится набирать полностью имена всех пересылаемых файлов. Поэтому испытываешь воистину большое облегчение, когда узнаешь, что существуют такие команды как mput и mget. Они позволяют задать шаблон имени пересылаемых файлов, и будут дополнительно переспрашивать, надо ли пересылать каждый конкретный файл. Благодаря этому можно (самый крайний случай) заказать пересылку всех файлов:

```
ftp > mget *
```

а потом либо подтверждать пересылку очередного файла, либо отказываться. Конечно, когда файлов в каталоге очень много, то и это окажется утомительной процедурой, но ведь можно задать более разумный шаблон! Так что думайте, как облегчить себе работу.

Перед тем, как начать пересылку файлов, следует еще выполнить одну из команд, определяющих режим пересылки: ascii или binary . По умолчанию программа использует режим "ascii", и это вполне допустимо при пересылке текстовых файлов, но если вы собираетесь передать или получить исполняемый файл, то необходимо задать режим "binary". Процесс пересылки файлов можно прервать с помощью комбинации клавиш <Ctrl>+<C>.

Пока вы находитесь в программе ftp, вы можете выполнить некоторые операции с файлами и каталогами на удаленном компьютере (конечно, для этого надо иметь соответствующие права). По команде

```
ftp > rename from_name to_name
```

осуществляется переименование файла или каталога; команда

```
ftp > mkdir name
```

создает каталог, а

```
ftp > delete name
```

удаляет файл или каталог. Еще одна интересная команда - system, позволяет выяснить тип операционной системы на удаленном компьютере. Ну, и наконец, команда close (или disconnect) позволяет завершить сеанс работы с удаленным компьютером, не выходя из программы ftp (т. е. предполагается, что после этого вы снова дадите команду open, например, для соединения с другим компьютером). Если же вы хотите вообще выйти из программы, то надо дать команду bye.

Виртуальные терминалы и интерфейс командной строки NetWork Simulator'a.

Виртуальные устройства в NET-Simulator управляются при помощи интерфейса командной строки из виртуальных терминалов. Терминал устройства можно открыть двойным кликом на значке устройства или через контекстное меню. Поддерживается история команд, клавиши вверх/вниз позволяют просматривать историю команд.

Список команд доступных на данном устройстве можно посмотреть командой help. Сочетание клавиш Ctrl+L очищает терминал. Краткая справка по любой команде выводится при вызове команды с опцией -h.

Справочник команд:

- help
- route
- ifconfig
- ping
- arp
- mactable

help — выводит список доступных команд.

help [-h]Опции Описание

-h Краткая справка.

route — позволяет управлять таблицей маршрутизации устройств поддерживающих протокол IP4.

route [-h] [{-add|-del} <target> [-netmask <address>] [-gw <address>] [-metric <M>] [-dev <If>]]Опции Описание

-h Краткая справка.

target Адрес назначения. Назначением может быть подсеть или отдельный узел в зависимости от значения маски подсети. Если маска равна 255.255.255.255 или отсутствует совсем назначением будет узел, иначе назначением будет сеть.

-add Добавляет новый маршрут в таблицу маршрутизации.

-del Удаляет маршрут из таблицы маршрутизации.

-dev <If> Принудительно присоединяет маршрут к определенному интерфейсу. If — имя интерфейса.

-gw <address> Направляет пакеты по этому маршруту через заданный шлюз. address — адрес шлюза.

-netmask <address> Маска подсети используемая совместно с адресом назначения при добавлении маршрута. address — маска. Если маска не задана явно подразумевается 255.255.255.255.

-metric <M> Метрика используемая в данном маршруте. M — целое число большее или равное нулю.

Если route вызывается без параметров, то команда выводит на экран таблицу маршрутизации:

```
=>route
IP routing table
Destination     Gateway      Netmask         Flags  Metric  Iface
10.0.0.0        *            255.0.0.0       U      1       eth0
11.0.0.0        10.0.0.10    255.0.0.0       UG     1       eth0
192.168.120.1   10.0.0.10    255.255.255.255 UGH    1       eth0
```

Если маршрут не использует шлюз, вместо адреса шлюза выводиться *. Flags может содержать значение: U — маршрут активен, G — маршрут использует шлюз, H — назначением является узел.

Примеры:

```
=>route -add 192.168.120.0 -netmask 255.255.255.0 -dev eth0
=>route
IP routing table
Destination     Gateway        Netmask         Flags  Metric  Iface
192.168.120.0   *              255.255.255.0   U      1       eth0
=>
=>route -add 192.168.121.10 -gw 192.168.120.10
=>route
```

```
IP routing table
Destination     Gateway         Netmask         Flags   Metric  Iface
192.168.120.0   *               255.255.255.0   U       1       eth0
192.168.121.10  192.168.120.1   255.255.255.255 UGH     1       eth0
=>
```

ifconfig — конфигурирует сетевые интерфейсы.

ifconfig [-h] [-a] [<interface>] [<address>] [-broadcast <address>] [-netmask <address>] [-up|-down] ОпцииОписание

-h Краткая справка.

-a Показывать информацию о всех интерфейсах. Если данная опция отсутствует выводится информация только об активных интерфейсах.

interface Конфигурировать или показать информацию только о заданном интерфейсе.

address IP-адрес присваиваемый интерфейсу.

-broadcast <address> Широковещательный адрес присваиваемый интерфейсу. address — широковещательный адрес.

-netmask <address> Маска подсети используемая совместно с адресом. address — маска. Если маска не задана явно, маска принимается равной стандартным значения для стандартных классов подсетей A, B и C.

-up Активирует интерфейс. При активизации интерфейса для него автоматически добавляется соответствующий маршрут в таблице маршрутизации.

-down Деактивирует интерфейс. При деактивации интерфейса соответствующий маршрут автоматически удаляется из таблицы маршрутизации.

Если ifconfig вызывается без параметров, то команда выводит на экран данные о состоянии всех активных интерфейсов:

```
=>ifconfig
eth0    Link encap:Ethernet  HWaddr 0:0:0:0:CF:0
        inet addr:192.168.120.1  Bcast:192.168.120.255  Mask:255.255.255.0
        UP
        RX packets:23 errors:0 dropped:0
        TX packets:23 errors:0 dropped:0
        RX bytes:0 TX bytes:0
```

HWaddr — уникальный 6-ти байтовый адрес интерфейса, аналогичный MAC-адресу в Ethernet сетях. Назначается автоматически.

Примеры:

```
=>ifconfig eth0 192.168.120.1 -up
```

```
=>ifconfig
eth0    Link encap:Ethernet  HWaddr 0:0:0:0:CF:0
        inet addr:192.168.120.1  Bcast:192.168.120.255  Mask:255.255.255.0
        UP
        RX packets:0 errors:0 dropped:0
        TX packets:0 errors:0 dropped:0
        RX bytes:0 TX bytes:0
```

ping — использует ICMP протокол что бы проверить достижимость интерфейса удаленного узла. ping посылает удаленному узлу ICMP ECHO_REQUEST и ожидает в течении определенного промежутка времени ICMP ECHO_RESPONSE. В случае получения ответа выводит данные о прохождении ICMP-пакета по сети.

ping [-h] [-i <interval>] [-t <ttl>] <destination> Опции Описание

-h Краткая справка.

-i <interval>Задает частоту ICMP-запросов. interval — интервал между запросами в секундах. По умолчанию отсылается один пакет в секунду.

-t <ttl> Задает значение атрибута Time to Live в генерируемых IP-пакетах. ttl — целое число 0-255. По умолчанию TTL равно 64.

destination IP-адрес исследуемого узла

Примеры:

```
=>ping 192.168.120.1
PING 192.168.120.1
64 bytes from 192.168.120.1: icmp_seq=0 ttl=62 time=477 ms
64 bytes from 192.168.120.1: icmp_seq=1 ttl=62 time=435 ms
64 bytes from 192.168.120.1: icmp_seq=2 ttl=62 time=234 ms
64 bytes from 192.168.120.1: icmp_seq=3 ttl=62 time=48 ms
64 bytes from 192.168.120.1: icmp_seq=4 ttl=62 time=87 ms
64 bytes from 192.168.120.1: icmp_seq=5 ttl=62 time=56 ms
```

ping выводит результат исследования удаленного узла в следующем формате: 64 bytes from 192.168.120.1 — размер полученного ответа и адрес источника ответа. В NET-Simulator размер пакета имеет условное значение и всегда равен 64B. icmp_seq=0 — номер пакета. Каждый запрос содержит свой номер, как правило формируется инкрементно. ping выводит номер пакета из каждого полученного ответа. ttl=62 — значение TTL из полученного ответа. time=48 ms — время прохождения пакетом полного маршрута (туда и обратно, round-trip time) в миллисекундах.

arp — показывает ARP-таблицу устройства. Кроме того опция -r позволяет сформировать запрос для определения MAC-адреса по явно заданному IP-адресу. Эта функция обычно отсутствует в реальных устройствах, в NET-Simulator она добавлена для наглядности при изучении протоколов канального и сетевого уровня.

arp [-h] [-r <IP-address> <interface>]Опции Описание

-h Краткая справка.

-r <IP-address> <interface> Прежде чем вывести ARP-таблицу предпринимает попытку найти MAC-адрес по явно заданному IP-адресу. IP-address IP-адрес для которого определяется MAC-адрес. interface имя интерфейса в сети подсоединенной к которому будет происходить поиск.

Если arp вызывается без параметров, то команда выводит на экран ARP-таблицу:

```
=>arp
Address         HWaddress       iface
10.0.0.10       0:0:0:0:BC:0    eth0
10.0.0.11       0:0:0:0:1F:2    eth0
```

Примеры:

```
 =>arp -r 192.168.120.12 eth1
Address         HWaddress       iface
10.0.0.10       0:0:0:0:BC:0    eth0
10.0.0.11       0:0:0:0:1F:2    eth0
192.168.120.12   0:0:0:0:12:1    eth1
```

mactable — показывает таблицу MAC-адресов коммутаторов второго уровня.

mactable [-h]Опции Описание

-h Краткая справка.

Примеры:

```
=>mactable
MACAddress      port
0:0:0:0:B3:0    0
0:0:0:0:2F:2    0
0:0:0:0:03:0    3
```

где port — номер порта на коммутаторе. Нумерация портов идет по порядку начиная с нуля.

РАЗДЕЛ 6. ГЛОБАЛЬНЫЕ СЕТИ

Тема 18. Основные принципы построения глобальных сетей

Структура глобальных сетей

Глобальные сети *(WAN, Wide Area Networks) позволяют организовать взаимодействие между компьютерами на больших расстояниях*. В идеале глобальная компьютерная сеть должна передавать данные абонентов любых типов, которые есть на предприятии и нуждаются в удаленном обмене информацией. Для этого глобальная сеть должна предоставлять целый комплекс услуг: передачу пакетов локальных сетей, обмен факсами, передачу телефонных разговоров, обмен видеоизображениями и т.д.

Из приведенного выше перечня услуг глобальных сетей видно, что они используются в основном как транзитный транспортный механизм, предоставляющий только услуги трех нижних уровней модели OSI. В последнее время, однако, в связи с развитием сети Internet, где представлен самый широкий спектр услуг протоколов верхнего уровня (WWW, News и д.р.), к сетевым ресурсам глобальных сетей предъявляет новые требования. В Наблюдается сближения технологий глобальных и локальных сетей на разных уровнях модели OSI - от транспортных до прикладных.

В общем случае глобальная сеть строится с помощью каналов связи, которые соединяются коммутаторами глобальной сети. Такие коммутаторы называются также *центрами коммутации пакетов (ЦАП)*. Отметим, что в зависимости от технологий передачи данных в глобальных сетях пакеты могут называться *кадры, ячейки*. Коммутаторы устанавливаются в тех географических пунктах, в которых требуется ответвление или слияние потоков данных конечных абонентов или магистральных каналов, переносящих данные многих абонентов. Выбор места установки коммутаторов определяется многими факторами: наличием квалифицированного персонала по обслуживанию в данной местности, надежностью местной сети, наличием необходимых линий абонентов, стоимостью и т.д. Абоненты сети подключаются к коммутаторам также по *выделенным каналам связи*, которые имееют более низкую пропускную способность, чем *магистральные каналы*. Для подключения конечных пользователей допускается использование *коммутируемых* (не выделенных) каналов, т.е. каналов телефонных сетей. Такие каналы имеют значительно низшее качество связи из- высокого уровня шумов. Конечные узлы глобальной сети более разнообразны, чем конечные узлы локальной сети. На рис 1. показан пример построения глобальной сети. Как видно из рисукна к глобальной сети могут подключаться как отдельные домашние компьютеры, так и целые локальные сети, телефонные станции, а также друигие устройства,

требующие каналов связи. При этом для совмещения передачи компьютерных и голосовых данных используются специальные устройсва, называемые *мультиплесорами*. Обычно предача голоса имеет более высокий приоритет.

При передаче данных через глобальную сеть маршрутизаторы и коммутаторы работают с той же логикой, что и в локальных сетях. В этом случае последние называются *удаленными коммутаторами*. Маршрутизаторы принимают решение о пересылке пакетов на основании номера сети какого либо протокола сетевого уровня (например IP) . Следует при использовании предприятием глобальной сети четко выяснить перечень предоставляемых глобальной сетью услуг, а также определить интерфейс взаимодействия сети предприятия с глобальной сетью, чтобы его оборудование и программное обеспечение корректно сопрягалось с соответствующим оборудованием и программным обеспечением глобальной сети. Протоколы взаимодействия глобальной сети с абонентами называются *интерфейсы пользователь- сеть (User-to-Network Interface, UNI)*, а протоколы взаимодействия коммутаторов внутри глобальной сети называются *интерфейсами сеть – сеть (Network- to Network Interface, NNI*). Аппаратура (устройства), вырабатывающие данные для передачи в глобальную сеть называются устройствами *DTE (Data Terminal Equipment)*- порт маршрутизатора, модем домашнего пользователя и т.д. Так как с этих устройств данные передаются в глобальную сеть по каналу связи, имеющему определенный стандарт, то устройства DTE оснащаются устройствами, называемыми DCE (Data Circuit terminating Equipment). Между этими устройствами существует свой интерфейс (стандарт), наиболее популярный из известных – RS-232 C/ V245. Он представляет собой 25 контактный разъем, где на каждый контакт должен поступать в соответствии со стандартом определенный сигнал (питание, земля, передающий сигнал, принимаемый сигнал, сигнал синхронизации и т.д.). Скорость передачи данных до 115 200 бит/ с. Указанный интерфейс, например, реализован во всех компьютерах в виде COM – порта.

Кроме этого существуют другие интерфейсы – RS 449, V35, X 21, HSSI (High – Speed Serial Interface) Некоторые из них могут поддерживать скорость до 10 Мбит на расстоянии до 10 метров.

Типы глобальных сетей

При построения глобальной сети необходимо учитывать множество различных факторов, но главными из них является выбор типа глобальной сети, или, другими словами выбор метода организации каналов передачи информации.

В общем случае выделяют три типа глобальных сетей. Это сети с использованием:

- выделенных каналов
- коммутации каналов
- коммутации пакетов

Отметим особенности каждого из типов глобальных сетей.

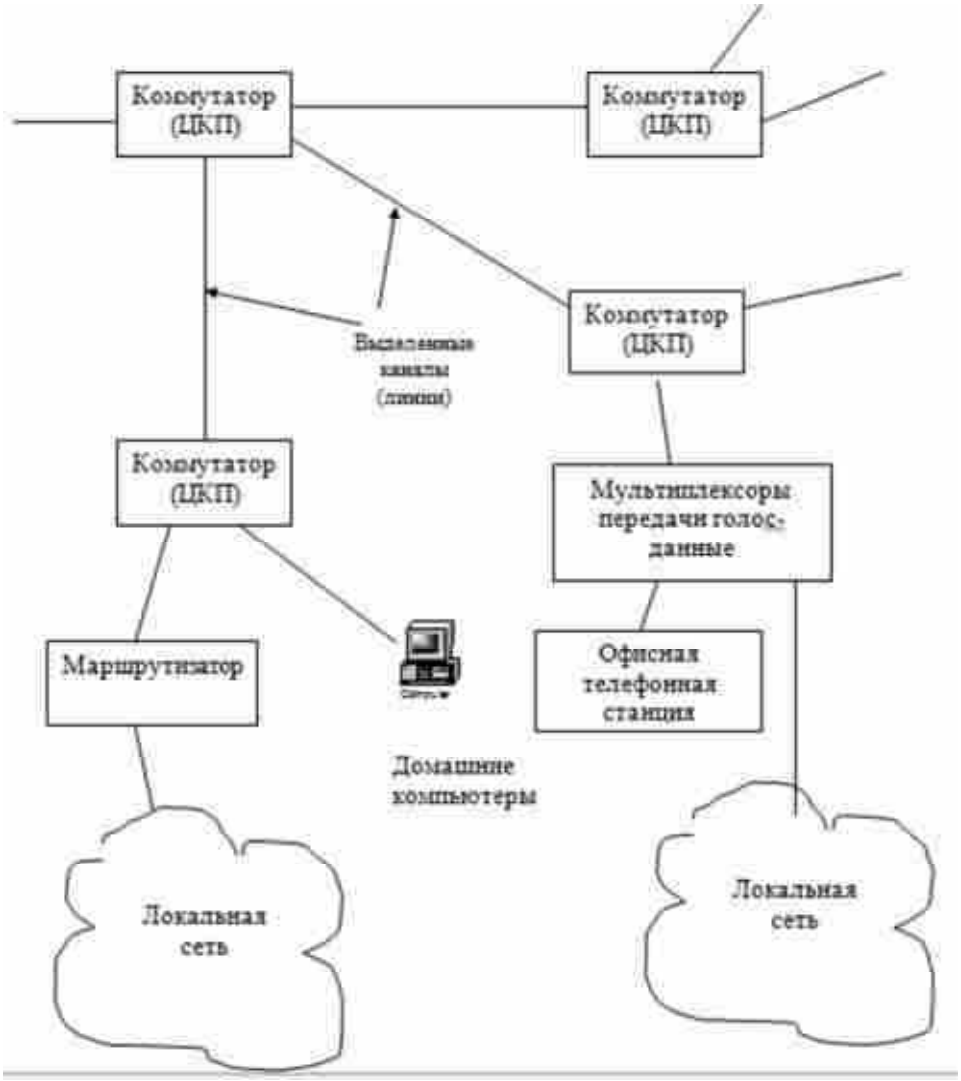


Рисунок 49 – Структура глобальной компьютерной сети

Выделенные каналы

Выделенные каналы можно получить у телекоммуникационных компаний (в Республике Беларусь, например, у Белтелеком), которые владеют каналами дальней связи и сдают их в аренду.

Использовать выделенные линии можно двумя способами:

- территориальная сеть строится с их помощью для соединения между собой коммутаторов (как на рис. 49);
- соединение между собой только объединяемых локальных сетей или конечных абонентов другого типа.

Второй случай является более простым и предпочтительным, т.к. так как отсутствуют протоколы глобальных сетей. Иногда второй способ называется «услуги выделенных каналов», так как в нем действительно больше ничего не используется из технологий собственно глобальных сетей. Выделенные каналы

активно применяются сегодня для связи между крупными локальными сетями, так эта услуга гарантирует пропускную способность арендуемого канала. При большом, однако, количестве географически удаленных точек и интенсивном трафике использование выделенных каналов приводит к высоким затратам за счет большого числа арендуемых каналов.

Глобальные сети с коммутацией каналов

Сети с коммутацией каналов в настоящее время используют каналы двух типов: традиционные аналоговые телефонные каналы и цифровые каналы с интеграцией услуг ISDN (будет рассмотрена в следующей лекции). Преимуществом сетей с коммутацией каналов является их широчайшая распространенность – обычные телефонные сети. Последнее время сети ISDN также стали использоваться в нашей республике.

Основным недостатком аналоговых телефонных сетей является низкое качество составного канала из-за перекрестных частотных помех Цифровые каналы связи лишены указанных недостатков, так как по каналу передается сигналы в специальном цифровом кодировании. Сети с коммутацией каналов имеют тот недостаток, что пользователь платит не за объем переданной или полученной информации, а за время подключения. Одна для работы дома телефонные каналы связи являются единственной возможностью выхода в глобальную компьютерную сеть.

Глобальные сети с коммутацией пакетов.

Принцип коммутации пакетов был рассмотрен в предыдущих лекциях. К таким сетям относятся в настоящее время такие технологии как X25, frame relay, SDMS и ATM, которые будут подробнее рассмотрены ниже.

Магистральные сети и сети доступа

Территориальные глобальные сети можно разделить на две большие категории:

- Магистральные сети
- Сети доступа

Магистральные сети используются для образования связей между крупными локальными сетями, принадлежащим большим подразделениям. Они должны обеспечить высокую пропускную способность, т.к. на магистрали объединяются потоки большого количества сетей. Кроме этого они должны обеспечивать высокий коэффициент готовности, т.к. через них может проходить очень важная оперативная информация. Обычно в качестве магистральных сетей используются цифровые выделенные каналы со скоростями от 2 до 622 Мбит / с и используются технологии сетей frame relay, ATM, X25 или TCP/ IP сети. Для обеспечения высокой готовности магистрали используется смешанная избыточная топология.

Под сетями доступа понимаются территориальные сети, необходимые для связи небольших локальных сетей и отдельных удаленных компьютеров с

центральной сетью предприятия. Вопросам удаленного доступа в последнее время уделяется особенно важное значения, т.к. быстрый доступ к корпоративной информации из любой географической точки является в настоящее время важным фактором для своевременного принятия управленческих решений. В качестве удаленных узлов могут быть также банкоматы или кассовые аппараты. К сетям доступа предъявляются требования, существенно отличающиеся от требований к магистральным сетям. Так как точек удаленного доступа может быть много, то в этом случае сеть доступа должна иметь очень разветвленную структуру, которая может быть использована сотрудниками как дома, так и в командировках. Кроме этого стоимость удаленного доступа должна быть не высокой, чтобы экономически оправдать большое число удаленных пользователей. В качестве сетей доступа обычно применяют телефонные аналоговые сети, сети ISDN, иногда сети frame relay. Для таких сетей используются каналы со скоростью 64 Кбит/с – 2 Мбит/с.

Программные и аппаратные средства , которые обеспечивают подключение компьютеров или локальных сетей удаленных пользователей к глобальной сети называются *средствами удаленного доступа* . Обычно на клиентской стороне это модем и соответствующее программное обеспечение.

Организацию массового удаленного доступа со стороны сети обеспечивает обычно *сервер удаленного доступа (Remote Access Server, RAS).* Такие сервера имеют много низкоскоростных портов для подключения пользователей через аналоговые телефонные сети или ISDN.

На рис. 50 показана структура глобальной сети, объединяющая в корпоративную сеть отдельные локальные сети и удаленных пользователей.

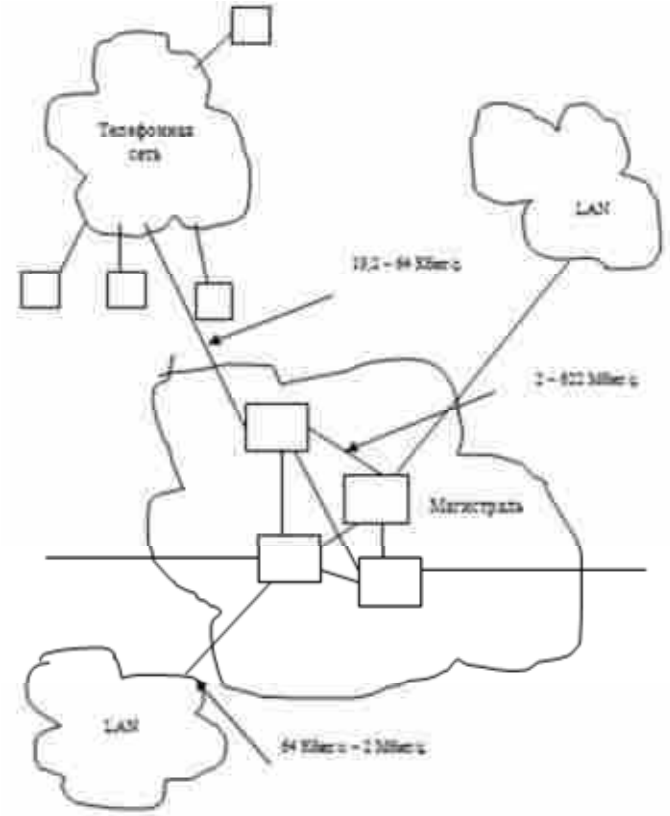


Рисунок 50 – Объединение локальных сетей в глобальную сеть

Глобальные сети на основе выделенных каналов

Выделенный канал – это канал с фиксированной полосой пропускания или фиксированной пропускной способностью, постоянно соединяющий двух абонентов. Абонентами могут быть как отдельные устройства (компьютеры или терминалы), так целые сети.

Выделенные каналы делятся на *аналоговые и цифровые*..

Аналоговые выделенные каналы**.**

Аналоговые выделенные каналы (линии) могут быть 2-х проводные или 4-проводные. В 4- х проводных линиях два провода используются на прием, а два на передачу, что значительно увеличивает пропускную способность канала.

Аналоговые выделенные каналы делятся на *нагруженные и ненагруженные*.

Первую группу составляют линии , проходящие через аппаратуру телефонных станций, и работают на тональных частотах от 3,1 КГц до 108 КГц.

Вторая группа выделенных аналоговых линий – это линии, которые не проходят через аппаратуру уплотнения телефонных станций. Такие линии обладают широкой полосой пропускания (до 1 МГц).

Для передачи информации по аналоговым линиям связи используется частотное разделение каналов, где каждый канал имеет собственную частотную полосу. Поэтому недостатком таких линий связи является влияние каналов друг на друга, т.е. наличие перекрестных частотных помех.

Для передачи данных по выделенным нагруженным аналоговым линиям связи используются специальные устройства, называемые *модемами*. Модемы преобразуют цифровой сигнал в аналоговый с помощью методов аналоговой модуляции. В зависимости от режимов работы различные модемы обеспечивают различные скорости передачи данных : от 1200 бит/с до 33,6 Кбит/с.

Цифровые выделенные каналы

Цифровые выделенные линии представляют собой постоянные линии связи, работающие на принципе разделения каналов по времени. Исторически существует для таких линий две технологии передачи данных: более ранняя PDH (Plesiochronic Digital Hierarchy) – почти синхронная (плезио) иерархия и SDH (Synchronous Digital Hierarchy) – синхронная иерархия. В США синхронная иерархия реализована в стандарте SONET.

Основной принцип технологии *PDH* заключается в объединении на более высоком уровне в один канал несколько более низкоскоростных каналов. В начале (60-е годы 20-го столетия) была разработана аппаратура низшего уровня (ее назвали

аппаратура Т1), которая объединила в цифровом виде на постоянной основе 24 абонента. Каждый абонентский канал образовывал поток данных скоростью 64 Кбит/ с. Затем на более высоком уровне четыре канала Т1 объединялись в более скоростной канал Т2, передающий данные со скоростью 6,312 Мбит/с. В свою очередь семь каналов Т2 объединялись в канал Т3, имеющий скорость 44,736 Мбит/с. АппаратураТ1,Т2,Т3 образует при взаимодействии иерархическую сеть (Рис.51).

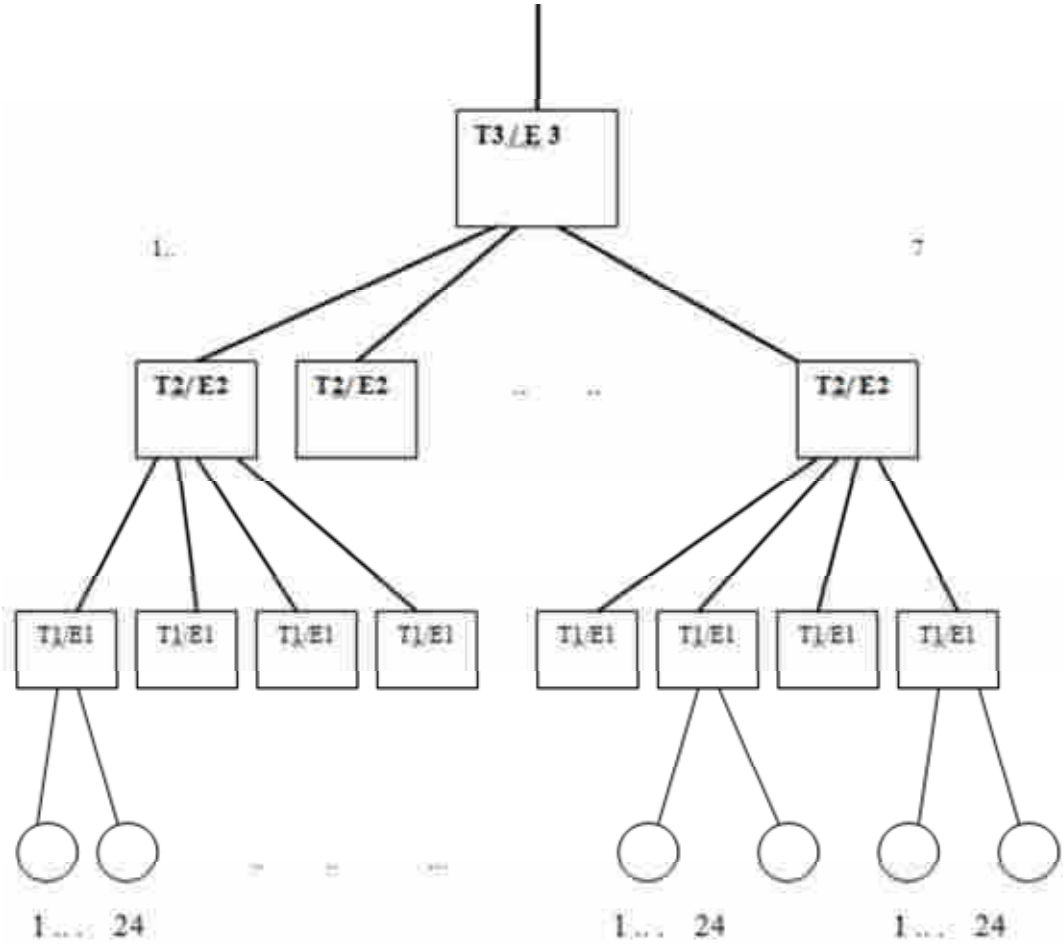


Рисунок 51 – Иерархическая сеть PDH

Глобальные сети с коммутацией каналов

Глобальные сети с коммутацией каналов используют услуги телефонных сетей. Телефонные сети делятся на *аналоговые и цифровые* в зависимость от способов коммутации (мультиплексирования) абонентских и магистральных каналов. Аналоговые телефонные сети принимают данные от абонентов в аналоговой форме, а мультиплексирование и коммутацию осуществляют как аналоговым, так и цифровым методами. В цифровых сетях информация от абонентов поступает в цифровом виде, и используются цифровые методы коммутации.

Аналоговые телефонные сети

Наиболее популярными аналоговыми коммутируемыми каналами являются обычные телефонные сети. Такие сети в настоящее время малопригодны для

построения магистралей, так как их максимальная пропускная способность составляет 56 Кбит/с, да и то в случае использования цифровых коммутаторов. В общем случае средняя пропускная способность аналоговых телефонных сетей составляет 9600 бит/с. В настоящее время аналоговые телефонные сети используются для организации индивидуального удаленного доступа, например, подключения с домашнего компьютера к сети Интернет. Соединение локальных сетей с помощью аналоговых коммутируемых каналов является экономически невыгодным из- за низкой пропускной способности и необходимости оплачивать не количество передаваемой информации, а время соединения. Обычно такие каналы для соединения локальных сетей рекомендуется использовать только для передачи сводок, имеющих небольшие объемы. Для подключения к физическим линиям связи используются как модемы, используемые только для коммутируемых каналов, так и модемы универсальные, применяемые также и на выделенных каналах. Последние стоят дороже. В отличие от модемов для выделенных каналов в модемах для коммутируемых каналов существует функция набора телефонного номера.

Цифровые телефонные сети

К первым цифровым телефонным сетям относятся так называемые службы Switched 56 (коммутируемые каналы 56 Кбит/с) и цифровые сети с интегральными услугами ISDN (Intergrated Services Digital Network). В настоящее время в мире наблюдается тенденция вытеснения службы Switched 56 сетями ISDN.

Сети ISDN

В сетях ISDN данные обрабатываются в цифровом виде. Первоначально сети создавались для передачи голоса, но они могут также использоваться и для передачи компьютерных данных.

В сетях ISDN используются (интегрируются) несколько видов служб: *выделенные цифровые канал; коммутируемая телефонная сеть общего пользования:, сеть передачи данных с коммутацией пакетов, сеть передачи данных с трансляцией кадров (frame relay); средства контроля и управления сетью*. Стандарты ISDN описывают также ряд других услуг прикладного уровня: факсимильную связь на скорости 64 Кбит/ с, телексную связь на скорости 9600 Бит/с, видеотелекс на скорости 9600 Бит/с и некоторые другие. Базовой скоростью сети является скорость 64 Кбит/ с. Сеть поддерживает два типа пользовательского интерфейса: начальный интерфейс BRI (Basic Interface Interface) и основной интерфейс (Primay Rate Interface, PRI).

*Начальный интерфейс **BRI*** предоставляет пользователю два канала по 64 Кбит/с для передачи данных (каналы типа B) и один канал с пропускной способностью 16 Кбит/с для передачи управляющей информации (канал типа D). Все каналы работают в полнодуплексном режиме, что позволяет получить суммарную скорость передачи данных 144 Кбит/с.

Основной интерфейс PRI предназначен для пользователей с повышенными требованиями к пропускной способности. Интерфейс PRI поддерживает либо схему каналов 30 B+ D, либо схему 23 B + D. B обеих схемах канал D обеспечивает скорость 64 Кбит/ с. Первый вариант предназначен для Европы, второй для Северной Америки и Японии, соответствующими скоростями 2,0248 Мбит/с и 1,544 Мбит/с.

В настоящее время сети ISDN используются в основном как телефонные цифровые сети высокого качества. Их преимущество, по сравнению с традиционными телефонными сетями, является то, что они позволяют организовать одновременно несколько цифровых каналов через один телефонный провод и объединить различные транспортные и прикладные службы. В качестве магистралей указанные сети не используются из- за отсутствия скоростной службы коммутации пакетов и невысокие скорости каналов.

В сетях с коммутацией каналов в основном используются пакеты небольшого фиксированного размера.

Тема 19. Глобальные сети с коммутацией пакетов

Для глобальных сетей с выделенными или коммутируемыми каналами основные проблемы были сосредоточены на физическом и канальном уровне, так как на сетевом уровне работали сетевые протоколы IP или IPX, с помощью которых происходило объединение локальных сетей.

Для глобальных сетей с коммутацией пакетов используется другая оригинальная техника маршрутизация пакетов, использующая понятие *«виртуального канала»*.

Техника виртуальных каналов используется во всех сетях с коммутацией пакетов, кроме сетей TCP/IP.

Виртуальный канал устанавливается между абонентами сети перед тем, как начать передачу пакета с помощью посылки в сеть специального пакета – запрос на установление соединения (Call Request), который содержит адрес узла назначения.

Существуют два типа виртуального канала*: коммутируемый виртуальный канал (SVC- Switched Virtual Circuit) и постоянный виртуальный канал (Permanent Virtual Circuit).*

При создании *коммутируемого виртуального канала* коммутаторы сети настраиваются по запросу абонента (динамически), а создание постоянного виртуального канала происходит заранее, причем коммутаторы сети настраиваются вручную администратором сети. Принцип работы виртуального канала состоит в том, что маршрутизация пакетов между коммутаторами сети на основании таблиц маршрутизации происходит только *один раз* – при создании виртуального канала. После создания виртуального канала пакеты предаются с помощью так называемых

номеров или идентификаторов виртуальных каналов (VCI- Virtual Channel Identifier).

После прокладки виртуального канала через глобальную сеть коммутаторы больше не используют для пакетов этого соединения таблицу маршрутизации, а продвигают пакеты на основании номеров виртуальных каналов. Поскольку таблицы коммутации портов значительно меньше таблиц маршрутизации (в них содержаться только текущие соединения), то и пересылка пакетов осуществляется с большей скоростью.

Режим постоянного виртуального канала (PVC) является особенностью технологии маршрутизации в глобальных сетях. Отметим, что в сетях TCP/IP такого режима нет. Режим PVC является наиболее эффективным с точки зрения производительности сети, так большую часть работ по маршрутизации пакетов администратор сети выполняет на этапе подготовки.

Техника виртуальных каналов имеет свои достоинства и недостатки по сравнению с техникой IP- маршрутизации. Маршрутизация пакетов без предварительного установления соединения (IP- адресация) эффективна для кратковременных потоков данных. Кроме этого, если имеются дополнительные параллельные линии связи, пакеты могут продвигаться по ним, что увеличивает надежность сети. При использовании же виртуальных каналов очень эффективно передаются долговременные потоки, так как для установления виртуального канала требуется дополнительное время (5- 10 мс), что для кратковременных пакетов является неприемлемым.

Рассмотрим наиболее популярные сети с коммутацией пакетов.

Сети X.25

Технология сетей X.25 - самая старая из стандартных технологий построения территориальных сетей с коммутацией пакетов. До использования Интернет в коммерческих целях сети X.25 были единственными доступными для коммерческих целей сетями. Сети X.25 хорошо работают на ненадежных и зашумленных линиях связи за счет установления виртуального соединения и коррекции ошибок на двух уровнях- канальном и сетевом. Стандарт X.25 был разработан в 1974 году. Сети X.25 хорошо подходят для передачи трафика низкой интенсивности, например, для подключения удаленного терминала (например, банкомат, касса). Среди особенностей сети X.25 является то, что она может работать только с одним протоколом канального уровня, который называется LAB- B, т.е. в отличие от IP-сетей не может объединять разнородные локальные сети.

Сеть X.25 состоит из коммутаторов, соединенных высокоскоростными выделенными каналами.

Для адресации сетей X.25 и их соединения между собой используется международная нумерация, называемая IDN – International Data Numbers. Адреса имеют разную длину, которая может доходить до 14 десятичных знаков. Первые

четыре цифры IDN называются кодом идентификации сети (DNIC- Data Network Identification Code). DNIC поделен на две части: первая (три цифры) определяют страну нахождения сети, а вторая номер сети X.25 внутри страны. Для нумерации сети внутри страны остается только одна цифра, что позволяет иметь внутри страны только 10 сетей X.25. Если в стране требуется более чем 10 сетей X.25, то стране присваивается дополнительный номер. Остальные цифры используются для адресации пользователей внутри сети.

Коммутаторы сетей X.25 представляют собой более простые и дешевые устройства, чем маршрутизаторы сетей TCP/IP. Это связано с тем, что коммутаторы не выполняют операций преобразований различных форматов канального уровня (как IP- маршрутизатор для объединения локальных сетей различных технологий) и не определяют оптимальный путь прохождения пакетов. В отличие от коммутаторов локальных сетей коммутаторы сетей X.25 обмениваются информацией подтверждения о получении кадров и организуют повторную передачу утерянного кадра.

Отметим, что сети X.25 были разработаны для низкоскоростных линий связи (1200 – 9600 бит/с) с высоким уровнем помех, которые еще широко распространены в нашей республике.

Сети frame relay

Технология framerelay начинает занимать в территориальных сетях с коммутацией пакетов ту же нишу, которую заняла в локальных сетях технология Ethernet. Обе технологии предоставляют только базовый транспортный сервис, доставляя кадры в узел назначения без гарантий, дейтаграммным способом. Однако, если кадры теряются, то сеть framerelay, как и сеть Ethernet, не предпринимает никаких усилий для их восстановления. Поэтому полезная пропускная способность сервисов верхнего уровня в сетях framerelay зависит от качества каналов и методов восстановления пакетов протоколами верхнего уровня, расположенными над протоколом framerelay. Если каналы качественные, то кадры будут теряться и искажаться редко, так что скорость восстановления пакетов протоколом TCP или NCP будет вполне приемлема. Если же кадры искажаются и теряются часто, то полезная пропускная способность в сети framerelay может упасть в десятки раз, так, как это происходит в сетях Ethernet при плохом состоянии кабельной системы.

Поэтому сети framerelay неразрывно связаны с оптоволоконными кабелями, по крайней мере на магистральных каналах "коммутатор - коммутатор".

На оптоволоконных линиях связи они обеспечивают передачу данных со скоростью до 2 Мбит/с. Технология Frame relay использует для передачи данных технику виртуальных соединений, аналогичную техники сетей X.25.

В отличие от сетей X.25 и сетей TCP/ IP, однако, в сетях frame relay пользователь может заказать у владельца сети необходимый уровень качества обслуживания, что включает: CIR – согласованная информационная скорость, с

которой сеть будет передавать данные; Вс- максимальное количество байтов, которое сеть будет передавать сеть от этого пользователя за интервал времени Т; Ве- максимальное количество байтов, которое сеть будет передавать сверх установленного значения Вс за интервал времени Т.

Важной особенностью сетей frame relay является так же то, что производители оборудования стремятся поддержать передачу голоса. Магистральные коммутаторы сети frame relay передают голосовые кадры в первую очередь.

Отметим, что использование виртуальных каналов для построения сети имеет недостаток при большом количестве точек доступа к сети необходимо строить большое количество виртуальных каналов, который необходимо оплачивать отдельно. В сетях TCP/IP оплачивается количество точек доступа, а не количество связей между ними.

Тем не менее сети frame relay можно успешно использовать для объединения локальных сетей.

Технология ATM

Технология ATM (ATM- Asynchronous Transfer Mode) является самой современной иперспективной технологией глобальных сетей и разрабатывается как единый транспорт для с интеграцией всех возможных услуг. По замыслу разработчиков технология ATM сможет обеспечить следующие возможности:

- передачу в рамках одной сети компьютерного и мультимедийного (голос, видео) трафика, причем каждому виду трафика качество обслуживания будет соответствовать его потребностям;
- иерархию скоростей передачи данных от десятков Мбит/с до нескольких Гбит/с с гарантированной пропускной способностью для приложений;
- общие транспортные протоколы для локальных и глобальных сетей;
- сохранение существующих физических каналов связи и протоколов;
- взаимодействие с протоколами локальных и глобальных сетей: IP, ISDN, Ethernet, Token Ring и др.

Технология ATM совмещает в себе две технологии- коммутации пакетов и коммутации каналов. От первой технологии она заимствует передачу данных в виде адресуемых пакетов, а от второй использование пакетов небольшого фиксированного размера в результате чего задержки в сети становятся более предсказуемыми. Основные стандарты технологии ATM были приняты в 1993 году и работы по их разработке активно продолжаются.

Трафик компьютерных сетей имеет ярко выраженный асинхронный характер и пульсирующий характер, т.к. каждый компьютер посылает свою информацию в непредсказуемые заранее (случайные) моменты времени. Трафик компьютерной сети очень чувствителен к потерям данных, так как их необходимо восстанавливать за счет повторной передачи.

Мультимедийный трафик, передающий голос или изображение, наоборот характеризуется низкой пульсацией и высокой чувствительностью к задержкам передачи данных. Кроме этого указанные трафики имеют различные размеры предаваемых пакетов. Пакеты, содержащие компьютерные данные, могут иметь пакет длиной до 4500 байт, при передаче которого через коммутатор может произойти значительная задержка пакетов с голосовыми данными. Поэтому в технологии ATM любой вид трафика передается пакетами фиксированной и очень маленькой длины в 53 байта. Пакеты ATM называются ячейками (cell). Поле данных ячейки занимает 48 байт, а заголовок 5 байт.

Кроме стандартизации и выбора одного и того же размера ячейки для любого вида трафика в технологии ATM реализуется важнейшее требование, предъявляемое к компьютерным сетям- *заказ пропускной способности и качества обслуживания*. (частично реализованного в сетях frame relay).

Магистраль ATM обеспечивает большие скорости передачи данных и может обеспечить связь между отдельными городами или даже странами. Массовое применению технологии ATM в локальных сетях сдерживается ее высокой стоимостью по сравнению с инвестициями в существующие технологии локальных сетей.

Тема 20. Сеть Интернет

Основные определения

24 октября 1995 года Федеральный сетевой совет (FNC), США, единодушно одобрил резолюцию, определяющую термин "Интернет" Это определение разрабатывалось при участии специалистов в области сетей и в области прав на интеллектуальную собственность.

Интернет — это глобальная информационная система, которая:

1. логически взаимосвязана пространством глобальных уникальных адресов, основанных на Интернет-протоколе (IP) или на последующих расширениях или преемниках IP;
2. способна поддерживать коммуникации с использованием семейства Протокола управления передачей, который называется Интернет-протоколом (TCP/IP) или его последующих расширений/преемников и/или других IP-совместимых протоколов;
3. обеспечивает, использует или делает доступной, на общественной или частной основе, высокоуровневые сервисы, надстроенные над описанной здесь коммуникационной и иной связанной с ней инфраструктурой.

Как видно из определения, в основе сети Интернет лежит использование протокола сетевого уровня, IP- протокола, над которым должны работать протоколы более высокого уровня, в первую очередь TCP – протокол.

Следует отметить, что революционизирующее влияние Интернет на мир компьютеров и коммуникаций не имеет исторических аналогов. Изобретение телеграфа, телефона, радио и компьютера подготовило почву для происходящей ныне их беспрецедентной интеграции. Интернет одновременно является и средством общемирового вещания, и механизмом распространения информации, и средой для сотрудничества и общения людей и компьютеров, охватывающей весь земной шар.

Интернет представляет собой один из наиболее успешных примеров того, какую пользу могут принести долгосрочные вложения и поддержка исследований и разработки информационной инфраструктуры. Начиная с ранних исследований в области пакетной коммутации, правительства различных стран, промышленность и академическая наука оставались партнерами в развитии и развертывании этой новой сетевой технологии.

В историческом развитии сети Интернет можно выделить четыре различных аспекта:

- технологическая эволюция исследований по пакетной коммутации;
- развитие методов и средств эксплуатации и управления глобальной и сложной сетевой инфраструктурой;
- социальный аспект, приведший к образованию широкого сообщества пользователей;
- коммерциализация, характеризуемая чрезвычайно эффективным превращением результатов исследований в развернутую, широко доступную информационную систему

Зарождение Интернет

У истоков создания сети Интернет стояла группа ученых и инженеров Управления перспективных исследований и разработок Министерства обороны США – DARPA (Defence Advanced Research Agency), созданная в 1962 году под руководством Дж. Ликлайдера. Этим ученым впервые была сформулирована концепция «галактической сети», объединяющая огромное количество компьютеров, и с помощью которой каждый пользователь сможет быстро получить доступ к данным и программам, расположенным на любом компьютере. Эта концепция очень близка по духу современному состоянию Интернет. Одновременно появились работы Леонарда Клейнрока по теории коммутации пакетов (пакетной коммутации), в которых теоретически обосновывалось возможность создания компьютерных сетей на основе пакетной коммутации. В дальнейшем проведенные эксперименты показали, что компьютеры с разделением

времени могут успешно работать вместе, выполняя программы и осуществляя выборку на удаленной машине. Стало ясно и то, что телефонная система того времени с коммутацией соединений абсолютно непригодна для создания компьютерной сети.

В 1967 году появился проект первой компьютерной сети ARPANET, а в 1968 были доработана структура и спецификации этой сети, которая должна была работать по технологии коммутации пакетов. После разработки первого коммутатора пакетов компанией BBN, который назывался тогда интерфейсным процессором, появилась возможность провести соединения с их помощью нескольких компьютеров, находящихся на большом расстоянии друг от друга. В сентябре 1969 один из коммутаторов был установлен в Калифорнийском университете, к нему был подключен компьютер, а второй коммутатор с подключенным компьютером разместили в Стэнфордском исследовательском институте. Через месяц было послано первое компьютерное сообщение из Колифорнийского университета, которое было успешно принято в Стэнфорде. Двумя следующими узлами ARPANET стали университет города Санта- Барбара и Университет штат Юта.

Таким образом, к концу 1969 года первые четыре компьютера были объединены в первоначальную конфигурацию ARPANET В последующие годы число компьютеров, подключенных к ARPANET, быстро росло.

Одновременно велись работы по созданию функционально полного протокола межкомпьютерного взаимодействия и другого сетевого программного обеспечения. В декабре 1970 года Сетевая рабочая группа (Network Working Group, NWG) завершила работу над первой версией протокола, получившего название Протокол управления сетью (Network Control Prtocol, NCP). После того, как в 1971- 1972 годах этот протокол был реализован на всех узлах ARPANET, пользователи сети смогли приступить к разработке приложений работающих над этим протоколом.

В марте 1972 года появилось *первое такое приложение – электронная почта*. Создателем программы электронной почты стал сотрудник вышеупомянутой компании BBN Рэй Томлисон (Ray Tomlinson), он же предложил использовать значок @ («собака»). Для своего времени электронная почта стала тем, же чем в наши дни является служба WWW- исключительно мощным катализатором роста всех видов межперсональных потоков данных.

Концепция объединения сетей

Интернет основывается на идее существования множества независимых сетей почти произвольной архитектуры, начиная от ARPANET. Интернет в современном понимании воплощает ключевой технический принцип *открытости сетевой архитектуры*. При подобном подходе архитектура и техническая реализация отдельных сетей не навязываются извне - они могут свободно выбираться

поставщиком сетевых услуг при сохранении возможности объединения с другими сетями посредством сетевого уровня.

Открытая сетевая архитектура подразумевает, что отдельные сети могут проектироваться и разрабатываться независимо, со своими уникальными интерфейсами, предоставляемыми пользователям и/или другим поставщикам сетевых услуг, включая услуги Интернет. При проектировании каждой сети могут быть приняты во внимание специфика окружения и особые требования пользователей. Вообще говоря, не накладывается никаких ограничений на типы объединяемых сетей или их территориальный масштаб.

Как уже указывалось выше, в сети ARPANET использовался протокол NCP.

Однако NCP не содержал средств для адресации сетей и отдельных машин. В обеспечении сквозной надежности протокол NCP полагался на хорошие линии связи.. Если какие-то пакеты терялись, протокол и поддерживаемые им приложения должны были остановиться. В модели NCP отсутствовало сквозное управление ошибками, поскольку ARPANET должна была являться единственной существующей сетью, причем настолько надежной, что от компьютеров не требовалось умение реагировать на ошибки. Таким образом, протокол NCP не соответствовал требованиям открытой сетевой архитектуры и требовал серьезной доработки.

Сотрудник DARPA Роберт Канн в 1972 году предложил разработать новую версию протокола, удовлетворяющую требованиям окружения с открытой сетевой архитектурой.

Этот протокол позднее будет назван ***Transmission Control Protocol/ Internet Protocol (TCP/IP — Протокол управления передачей/Межсетевой протокол)***.

В то время как NCP действовал в как драйвер устройства, новинка должна была в большей мере напоминать коммуникационный протокол.

В основе разработки нового протокола лежали четыре принципа:

- каждая сеть должна сохранять свою индивидуальность. При подключении к Интернет сети не должны подвергаться внутренним переделкам;

- передача пакетов должна идти по принципу "максимум возможного". Если пакет не прибыл в пункт назначения, источник должен вскоре повторно передать его;

- для связывания сетей должны использоваться черные ящики; позднее их назовут *шлюзами и маршрутизаторами*.

На локальном уровне не должно существовать глобальной системы управления.

Самыми первыми результатами по реализации указанных принципов стало:

- Общение между двумя компьютерами логически должно представляться как обмен непрерывными последовательностями байт. Для идентификации байта используется его позиция в последовательности.

- Управление потоком данных осуществляется на основе механизмов подтверждений. Получатель может выбирать, когда посылать подтверждение, распространяющееся на все полученные к этому моменту пакеты.

В публикациях того времени по объединению сетей (начало 70 –х годов) первоначально описывался один протокол, названный TCP. Он предоставлял все услуги по транспортировке и перенаправлению данных в Интернет. Планировалось, что протокол TCP будет поддерживать целый диапазон транспортных сервисов. Затем, однако, протокол TCP был раздел на два протокола — простой IP, обслуживающий только адресацию и перенаправление отдельных пакетов, и отдельный TCP, имеющий дело с такими аспектами, как управление потоком данных и нейтрализация потери пакетов. Для приложений, не нуждавшихся в услугах TCP, была добавлена альтернатива — Пользовательский дэйтаграммный протокол (User Datagram Protocol, UDP), открывающий прямой доступ к базовым сервисам уровня IP с приложений верхнего уровня.

Ключевая концепция создания Интернет состояла в том, что объединение сетей проектировалось не для какого-то одного приложения, но как универсальная инфраструктура, над которой могут быть надстроены новые приложения. Основой этих приложений являлся протокол TCP / IP.

Широкое распространение в 1980-е годы локальных сетей, персональных компьютеров и рабочих станций дало толчок бурному росту Интернет. Технология Ethernet, разработанная в 1973 году фирмой Xerox PARC, в наши дни является доминирующей сетевой технологией в Интернет, а персональные компьютеры и рабочие станции стали доминирующими компьютерами.

Рост Интернет вызвал важные изменения и в вопросах управления. Чтобы сделать сеть более дружественной для человека, компьютерам были присвоены имена, делающие ненужным запоминание числовых адресов. Пол Мокапетрис (Paul Mockapetris) из Института информатики Университета Южной Калифорнии придумал доменную систему имен (Domain Name System, DNS). DNS позволила создать масштабируемый распределенный механизм для отображения иерархических имен компьютеров (например, www.acm.org) в Интернет-адреса.

Еще одной особенностью, вызванной ростом Интернет, стало внесение изменений в программное обеспечение. Протокол TCP/IP стал встраиваться в существующие операционные системы Unix.

В целом стратегия встраивания протоколов Интернет TCP/IP в самую распространенную операционную систему, явилась одним из ключевых элементов успешного и повсеместного распространения Интернет.

Протокол TCP/IP был принят в качестве военного стандарта в 1980 году. Это позволило военным начать использование технологической базы Интернет и, в

конце концов, привело к разделению на военное и гражданское Интернет-сообщества. К 1983 году ARPANET использовало значительное число военных исследовательских, разрабатывающих и эксплуатирующих организаций.

Кроме этого к 1985 году технологии Интернет поддерживались широкими кругами исследователей и разработчиков. Интернет начинали использовать для повседневных компьютерных коммуникаций люди самых разных категорий. Особую популярность завоевала электронная почта, работавшая на разных платформах. Совместимость различных почтовых систем продемонстрировала выгоды массовых электронных коммуникаций между людьми.

Громадным шагом в развитии Интернет стала разработка в 1989 году Тимом Бернерсом-Ли гипертекстовой среды, а также разработка им первого Web-браузера, который назывался World Wide Web.

17 мая 1991 года на вычислительных системах Европейской физической лаборатории CERN (European Organization for Nuclear research) была установлена окончательная версия первого в мире Web-сервера.

Создание инфраструктуры Интернет

К середине 1970-х годов компьютерные сети начали расти, как грибы после дождя. Министерство энергетики США сначала создало сеть MFENet в интересах исследователей термоядерного синтеза с магнитным удержанием, затем специалисты в области физики высоких энергий получили сеть HEPNet, для астрофизиков из NASA построили сеть SPAN, национальный научный фонд (NSF), США, развернул сеть CSNET, объединившую специалистов по информатике из академических и промышленных кругов. Указанные сети должны были использоваться замкнутым сообществом специалистов; как правило, этим работа сетей и ограничивалась. Особой потребности в совместимости сетей не было; соответственно, не было и самой совместимости. Важным шагом по объединению сетей стало в 1985 году важное решение об обязательном использовании в NSFNet протокола TCP/IP.

Размах сети NSFNet, воспринимаемой уже как сеть Интернет Размеры ее финансирования составили 200 миллионов долларов за период с 1986 по 1995 год. В сочетании с качеством TCP/IP протоколов это привело к тому, что в начале 90-х семейство TCP/IP вытеснило или значительно потеснило во всем мире большинство других протоколов глобальных компьютерных сетей. К 1990 году окончательно разукомплектовали сеть ARPANET, которая не могла уже конкурировать с новыми технологиями Интернет. Протокол IP уверенно становился доминирующим сервисом транспортировки данных в глобальной информационной инфраструктуре.

Использование масок в IP- адресации

Основной недостаток использования классов IP- адресов напрямую состоит в том, что если организация имеет несколько сетевых номеров, то все компьютеры вне сети имеют доступ к этим адресам и сеть организации становится прозрачной.

Для устранения указанного недостатка адресное пространство сети разбивается на более мелкие непересекающиеся пространства – подсети (subnet). С каждой из подсетей можно работать как с обычной TCP/IP – сетью.

Разбивка адресного пространства на подсети осуществляется с помощью *масок***.**

Маска- это число, которое используется в паре с IP- адресом; двоичная запись маски содержит единицы в тех разрядах, которые должны в IP- адресах интерпретироваться как номер сети. Единицы в маске должны представлять непрерывную последовательность.

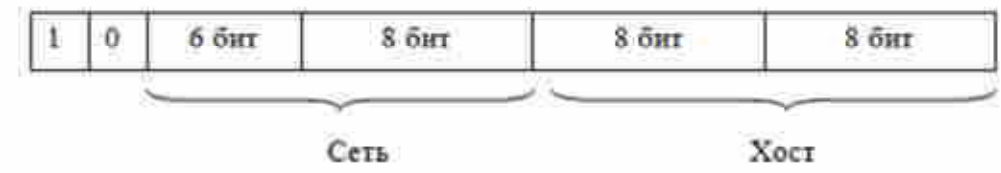
Для стандартных классов маски имеют следующие значения:

- Класс А – 11111111.00000000.00000000.00000000 (255.0.0.0)
- Класс В - 11111111.11111111.00000000.00000000 (255.255.0.0)
- Класс С - 11111111.11111111.11111111.00000000 (255.255.255.0)

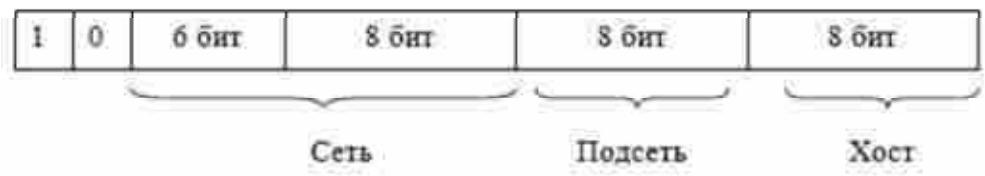
Рассмотрим, каким образом маска преобразует IP- адреса.

Пусть организация получила один IP- адрес класса В. Как известно, для сетей класса В первые два байта являются номером сети, а два остальные байта определяют номер узла. Для организации подсетей и их нумерации используются разряды байтов номеров узлов. В самом простом случае для нумерации подсетей используется первый байт номера узла.

Адрес до преобразования выглядел следующим образом:



После организации подсети IP- адрес стал выглядеть:

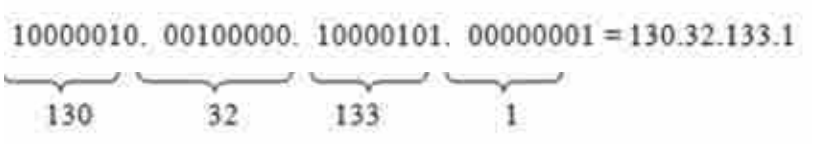


Задавая в третьем байте номера подсети, можно разбивать сеть на отдельные подсети и присваивать номера узлов внутри подсети. В этом случае нумерация

узлов внутри подсетей является локальным для организации и не видна во внешней сети. Все компьютеры вне организации видят одну большую IP- сеть и они должны поддерживать только маршруты доступа к шлюзам, соединяющим сеть организации с внешним миром.

Пример

IP- адрес сети класса В задан в виде:



а) Маска не используется. В этом случае номером сети являются первые два байта и определяют сеть 130.32.0.0, а номер узла равен 0.0.132.1

б) Используется маска:

11111111.11111111.10000000.00000000 = 255.255.128.0

В этом случае наложение маски на IP- адрес дает новое число, интерпретируемое как номер сети:

10000010. 00100000. 10000000. 00000000 = 130.32.128.0

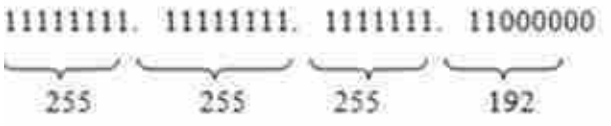
Номер узла в этой сети становится 0.0.5.1

Как видно из примера, снабжая IP-адреса маской, можно отказаться от понятий классов адресов и сделать более гибкой систему адресации сетей.

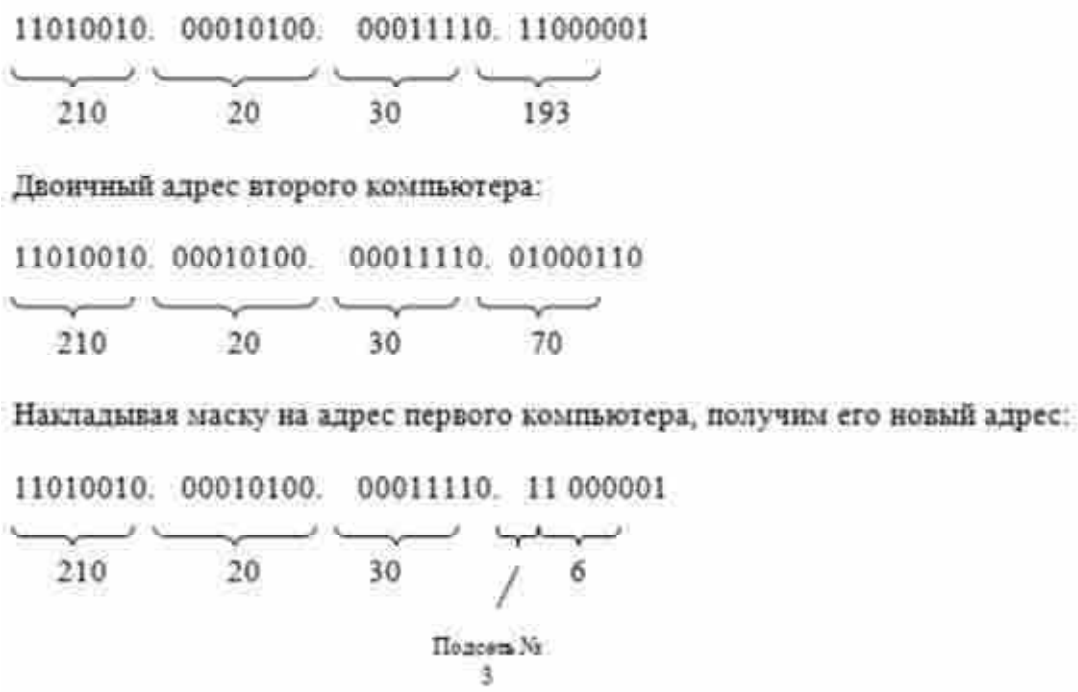
Пример

Пусть в сети работают два компьютера, имеющие два соответствующие IP-адреса: 210.20.30.193 и 210.20.30.70. Для разделения указанных компьютеров в две разные подсети используем маску 255.255.255.192

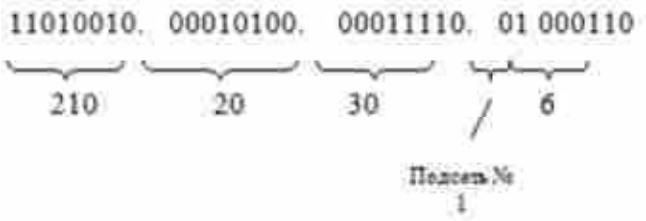
В двоичной форме маска имеет вид:



Двоичный адрес первого компьютера:



Накладывая маску на адрес второго компьютера, получим его новый адрес:



Таким образом, сеть с помощью маски разбилась на две подсети, номер второго компьютера в подсети стал равным шести.

Следует отметить, что в настоящее время наблюдается дефицит IP- адресов, выделяемых организацией InterNIC. Очень трудно получить адрес класса B и практически невозможно стать обладателем адреса класса A. Если же IP- сеть создана для работы в автономном режиме, без связи с Интернет, то администратор сети сам произвольно назначает номер. Но даже в этой ситуации в стандартах Интернет определены несколько диапазонов адресов, не рекомендуемых для использования в локальных сетях. Эти адреса не обрабатываются маршрутизаторами Интернет ни при каких условиях. Для сетей класса A – это сеть 10.0.0.0, в классе B- это диапазон из 16 номеров сетей 172.16.0.0 – 172.31.0.0, в классе C – это диапазон из 255 сетей – 192.168.0.0 – 192.168.255.0.

Для разрешения проблемы дефицита адресов осуществляется переход на новую версию IP- протокола- протокол IPv6, в котором резко расширяется адресное пространство за счет 16- байтных адресов.

Протокол IPv6, как развитие транспортных средств IP- протокола

Указанный протокол решает принципиальную проблему нехватки IP-адресов посредством использования 128- разрядных адресов вместо 32 – разрядных адресов,

благодаря чему адресное пространство расширяется в 296 раз. Результатом этого будет то, что любой житель Земли может получить в сове распоряжение несколько IP- адресов, новое количество адресов позволит подключить к сети свыше 1 квадрильона компьютеров в 1 триллионе сетей.

Адреса в IPv6 – протоколе разделяются на три типа: *обычные, групповые и нечеткие.*

Пакет с обычным адресом передается конкретному адресату, в то время как пакет с групповым адресом доставляется всем членам группы. Пакет с нечетким адресом доставляется только ближайшему члену данной группы.

В IPv6 128 разрядные адреса записываются в виде восьми 16- разрядных целых чисел, разделенных двоеточие. Каждое число представлено шестнадцатеричными цифрами, разделенными двоеточиями. Другими словами, необходимо вводить 32 шестнадцатеричные цифры для задания IP- адреса. IPv6 – адрес может выглядеть так: 501A:0000:0000:0000:00FC:ABCD:3F1F:3D5A.

Переход от традиционных IP- адресов к IPv6 – адресам займет ни один год и старая адресация будет постепенно замещаться новыми программными продуктами и оборудованием, использующим IPv6- протокол.

Среди других новых свойств IPv6 – протокола можно отметить также более рациональную структуру формата заголовка пакета, увеличение производительности маршрутизаторов, работающих с этим протоколом, возможность маркировки потока данных, если их необходимо обрабатывать особым образом, аутентификацию дейтаграмм и д.р.

Система доменов DNS

Выше было установлено, что для обращения к хостам используются 32-разрядные IP- адреса. Поскольку при работе в сети Интернет использовать цифровую адресацию сетей крайне неудобно, то вместо цифр используются символьные имена, называемыми *доменными именами.* Доменом называется группа компьютеров, объединенных одним именем. Символьные имена дают пользователю возможность лучше ориентироваться в Интернет, поскольку запомнить имя всегда проще, чем цифровой адрес.

На заре создания Интернет соответствия между именами хостов и их IP-адресами были размещены в единственном файле, который назывался Hosts.txt, который размещался на компьютере в центре InterNIC. Этот файл передавался по всем хостам еще совсем тогда крохотной сети. Стремительный рост Интернет заставил выработать новую концепцию механизма разрешения имен. С этой целью была разработана специальная система DNS (Domain Name System), для реализации которой был создан специальный сетевой протокол DNS. Начальные попытки создать единую копию целой базы данных имен и адресов оказались тщетными из-за громадного объема информации. Было принято решение строить распределенную

базу данных, а для увеличения производительности использовать механизм локального кэширования (сохранения в локальной базе данных). Доступ к распределенной базе данных не зависит ни от аппаратной платформы хоста, ни от коммутационной системы. Доступ к базе данных должны иметь все пользователи Интернет. Администрирование базы данных DNS возлагается на каждую организацию, которая подключается к Интернет. Организация должна инсталлировать свой собственный компьютер -сервер разрешения имен и ту часть распределенной базы данных, содержащей информацию о домене хостов данной организации. Сервер должен обслуживать хосты внутри организации и предоставлять доступ к базе данных этой организации извне.

Структура баз данных в системе DNS имеет иерархический вид, аналогичный иерархии файлов, принятой во многих файловых системах. Дерево имен начинается с корня, затем следует старшая символьная часть имени, вторая часть имени и т.д. Младшая часть имени соответствует конечному узлу сети. Все имена разделяются точками, причем иерархия задается справа налево, например, www.bseu.minsk.by

По имени можно получить информацию о профиле организации или ее местоположении. Шесть доменов высшего уровня определены следующим образом:

- gov – правительственные организации;
- mil – военные организации;
- edu – образовательные организации;
- com - коммерческие организации;
- org- общественные организации;
- net – организации, предоставляющие сетевые услуги, как правило, региональные сетевые организации.

Кроме того, все страны мира имеют свое собственное символьное имя, обозначающий домен верхнего уровня этой страны. Например, de – Германия, us – США, ru- Россия, by – Беларусь и т.д. Таким образом, адрес www.cdo.bseu.minsk.by означает, что компьютер дистанционного образования cdo находится в группе компьютеров (в домене) Белорусского государственного экономического университета bseu, в домене minsk в Республике Беларусь. Графически DNS можно представить в виде дерева, как на рисунке 52.

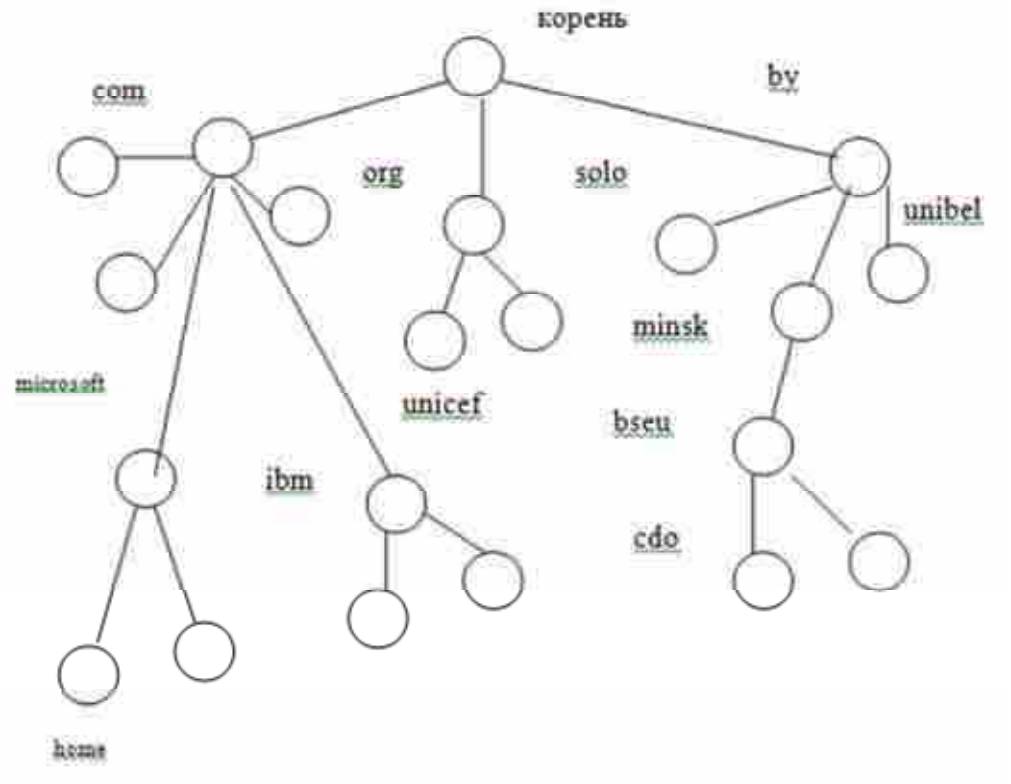


Рисунок 52 – Дерево системы DNS

DNS имеет три основные компоненты:

- Пространство имен домена (domain name space) и записи базы данных DNS (resource records).Они определяют структуру имен «дерева» и данных, связанных с этими именами. Запрос по данному имени возвратит IP- адрес хоста.
- Сервера имен (name servers). Сервера имен – это специальные компьютеры со специальными серверными программами, обрабатывающие информацию имен и данных имен. Сервер управляет всей информацией подчиненной ему области имен и данных домена. При обращении за информацией, который данный сервер не обслуживает, он должен или переправить запрос серверу, обслуживающему эту информацию, или стоящему на следующей ступени иерархии. Сервер, в распоряжении которого находится определенная часть информации об именах, является владельцем (authority) имен домена, а граница владения называется зоной (zone). Зоны строятся не на основе принадлежности какой-либо части данных к определенной организации, а распределяются автоматически серверами имен и должны обеспечить полную адресацию хостов.
- Программы разрешения имен (resolves). Эти программы возвращают информацию, хранящуюся в базе данных имен домена по запросу пользователя. Пользователь взаимодействует с пространством имен через указанные программы. Как правило эти программы реализуются в виде системного модуля, напрямую связанного с пользовательской программой, поэтому не требуется ни какого дополнительного протокола обмена.

Основным предназначением системы имен доменов является обеспечение механизма именования ресурсов. Этот механизм должен эффективно работать с различными хостами, сетями, семействами протоколов и типами организаций. Описанная выше структура DNS позволяет решать проблему адресации отдельных модулей изолировано, и, тем самым, создает универсальную модульную архитектуру.

Пользователь взаимодействует с пространством имен через программы разрешения. Для работы программ разрешения необходимо обращаться к серверам имен на других хостах, что может давать задержки от миллисекунд до нескольких секунд. Поэтому одной из важнейших свойств программ разрешения имен является возможность устранения сетевых задержек ответов. При этом используется механизм *кэширования результатов запросов имен*. Этот механизм ускоряет процесс определения имен, так в КЭШ-памяти накапливается информация о всех предыдущих именах , к которым обращалась программа.

Наиболее упрощенный и распространенный принцип работы такой программы с серверами имен показан на рисунке 53.

Рисунок 53 – Принцип работы с серверами имен

Программа пользователя запрашивает имя хоста и передает этот запрос программе разрешения имен. В первую очередь программа разрешения имен обращается за необходимым IP- адресом в собственную КЭШ- память. Если требуемого имени а КЭШ – памяти не находится, программа разрешения имен обращается к удаленному серверу имен, В случае нахождения необходимого имени, программа возвращает пользователю требуемый IP- адрес, одновременно записывая его в КЭШ- память.

Система DNS требует, чтобы доступ к информации определенной зоны мог быть осуществлен с нескольких серверов доменов. Существует механизм предоставления пользователям различных доменов совместного использования информации путем установления *доверительных отношений* между доменами. При

этом доверительные отношения могут быть как *двухсторонними*, так и *односторонними*.

При двухсторонних доверительных отношениях пользователь любого из двух доменов имеет доступ к информации, находящихся на соседнем домене.

При односторонних доверительных отношениях пользователь, находящийся в доверяемом домене, имеет доступ к серверам доменам доверителя, но не наоборот.

РАЗДЕЛ 7. БЕЗОПАСНОСТЬ КОМПЬЮТЕРНЫХ СЕТЕЙ

Тема 21. Защита информации в локальных и глобальных сетях.

Одной из наиболее очевидных причин нарушения системы защиты является умышленный несанкционированный доступ (НСД) к конфиденциальной информации со стороны нелегальных пользователей и последующие нежелательные манипуляции с этой информацией. Защита информации – это комплекс мероприятий, проводимых с целью предотвращения утечки, хищения, утраты, несанкционированного уничтожения, искажения, модификации (подделки), несанкционированного копирования, блокирования информации и т.п. Поскольку утрата информации может происходить по сугубо техническим, объективным и неумышленным причинам, под это определение попадают также и мероприятия, связанные с повышением надежности сервера из-за отказов или сбоев в работе винчестеров, недостатков в используемом программном обеспечении и т.д.

Следует заметить, что наряду с термином "защита информации" (применительно к компьютерным сетям) широко используется, как правило, в близком значении, термин "компьютерная безопасность".

Переход от работы на персональных компьютерах к работе в сети усложняет защиту информации по следующим причинам:

большое число пользователей в сети и их переменный состав. Защита на уровне имени и пароля пользователя недостаточна для предотвращения входа в сеть посторонних лиц;

значительная протяженность сети и наличие многих потенциальных каналов проникновения в сеть;

уже отмеченные недостатки в аппаратном и программном обеспечении, которые зачастую обнаруживаются не на предпродажном этапе, называемом бета-тестированием, а в процессе эксплуатации. В том числе неидеальны встроенные средства защиты информации даже в таких известных и "мощных" сетевых ОС, как Windows NT или NetWare.

Остроту проблемы, связанной с большой протяженностью сети для одного из ее сегментов на коаксиальном кабеле, иллюстрирует рис. 9.1. В сети имеется много физических мест и каналов несанкционированного доступа к информации в сети. Каждое устройство в сети является потенциальным источником электромагнитного излучения из-за того, что соответствующие поля, особенно на высоких частотах, экранированы неидеально. Система заземления вместе с кабельной системой и сетью электропитания может служить каналом доступа к информации в сети, в том числе на участках, находящихся вне зоны контролируемого доступа и потому особенно уязвимых. Кроме электромагнитного излучения, потенциальную угрозу представляет бесконтактное электромагнитное воздействие на кабельную систему. Безусловно, в случае использования проводных соединений типа коаксиальных кабелей или витых пар, называемых часто медными кабелями, возможно и

непосредственное физическое подключение к кабельной системе. Если пароли для входа в сеть стали известны или подобраны, становится возможным несанкционированный вход в сеть с файл-сервера или с одной из рабочих станций. Наконец возможна утечка информации по каналам, находящимся вне сети:

– хранилище носителей информации,

– элементы строительных конструкций и окна помещений, которые образуют каналы утечки конфиденциальной информации за счет так называемого микрофонного эффекта,

– телефонные, радио-, а также иные проводные и беспроводные каналы (в том числе каналы мобильной связи).

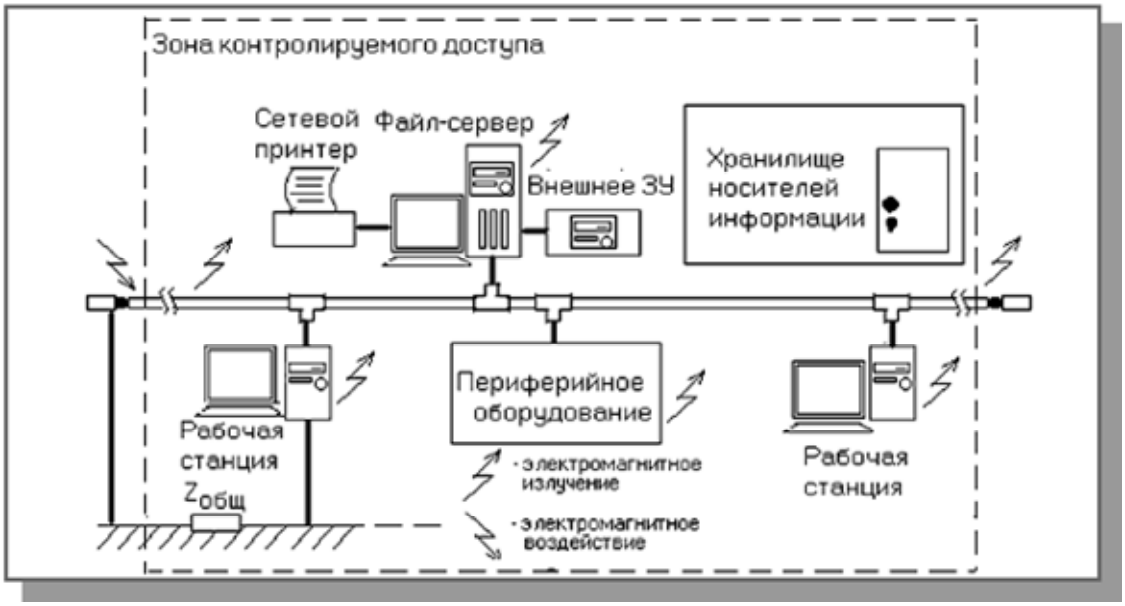


Рисунок 54 – Места и каналы возможного несанкционированного доступа к информации в компьютерной сети

Любые дополнительные соединения с другими сегментами или подключение к Интернет порождают новые проблемы. Атаки на локальную сеть через подключение к Интернету для того, чтобы получить доступ к конфиденциальной информации, в последнее время получили широкое распространение, что связано с недостатками встроенной системы защиты информации в протоколах TCP/IP. Сетевые атаки через Интернет могут быть классифицированы следующим образом:

1. Сниффер пакетов (sniffer – в данном случае в смысле фильтрация) – прикладная программа, которая использует сетевую карту, работающую в режиме promiscuous (не делающий различия) mode (в этом режиме все пакеты, полученные по физическим каналам, сетевой адаптер отправляет приложению для обработки).

2. IP-спуфинг (spoof – обман, мистификация) – происходит, когда хакер, находящийся внутри корпорации или вне ее, выдает себя за санкционированного пользователя.

3. Отказ в обслуживании (Denial of Service – DoS). Атака DoS делает сеть недоступной для обычного использования за счет превышения допустимых пределов функционирования сети, операционной системы или приложения.

4. Парольные атаки – попытка подбора пароля легального пользователя для входа в сеть.

5. Атаки типа Man-in-the-Middle – непосредственный доступ к пакетам, передаваемым по сети.

6. Атаки на уровне приложений.

7. Сетевая разведка – сбор информации о сети с помощью общедоступных данных и приложений.

8. Злоупотребление доверием внутри сети.

9. Несанкционированный доступ (НСД), который не может считаться отдельным типом атаки, так как большинство сетевых атак проводятся ради получения несанкционированного доступа.

10. Вирусы и приложения типа "троянский конь".

Классификация средств защиты информации

Защита информации в сети на рис. 53. может быть улучшена за счет использования специальных генераторов шума, маскирующих побочные электромагнитные излучения и наводки, помехоподавляющих сетевых фильтров, устройств зашумления сети питания, скремблеров (шифраторов телефонных переговоров), подавителей работы сотовых телефонов и т.д. Кардинальным решением является переход к соединениям на основе оптоволокна, свободным от влияния электромагнитных полей и позволяющим обнаружить факт несанкционированного подключения.

В целом средства обеспечения защиты информации в части предотвращения преднамеренных действий в зависимости от способа реализации можно разделить на группы:

Технические (аппаратные) средства. Это различные по типу устройства (механические, электромеханические, электронные и др.), которые аппаратными средствами решают задачи защиты информации. Они либо препятствуют физическому проникновению, либо, если проникновение все же состоялось, доступу к информации, в том числе с помощью ее маскировки. Первую часть задачи решают замки, решетки на окнах, защитная сигнализация и др. Вторую – упоминавшиеся выше генераторы шума, сетевые фильтры, сканирующие радиоприемники и множество других устройств, "перекрывающих" потенциальные каналы утечки информации или позволяющих их обнаружить. Преимущества технических средств связаны с их надежностью, независимостью от субъективных факторов, высокой устойчивостью к модификации. Слабые стороны – недостаточная гибкость, относительно большие объем и масса, высокая стоимость.

Программные средства включают программы для идентификации пользователей, контроля доступа, шифрования информации, удаления остаточной

(рабочей) информации типа временных файлов, тестового контроля системы защиты и др. Преимущества программных средств – универсальность, гибкость, надежность, простота установки, способность к модификации и развитию. Недостатки – ограниченная функциональность сети, использование части ресурсов файл-сервера и рабочих станций, высокая чувствительность к случайным или преднамеренным изменениям, возможная зависимость от типов компьютеров (их аппаратных средств).

Смешанные аппаратно-программные средства реализуют те же функции, что аппаратные и программные средства в отдельности, и имеют промежуточные свойства.

Организационные средства складываются из организационно-технических (подготовка помещений с компьютерами, прокладка кабельной системы с учетом требований ограничения доступа к ней и др.) и организационно-правовых (национальные законодательства и правила работы, устанавливаемые руководством конкретного предприятия). Преимущества организационных средств состоят в том, что они позволяют решать множество разнородных проблем, просты в реализации, быстро реагируют на нежелательные действия в сети, имеют неограниченные возможности модификации и развития. Недостатки – высокая зависимость от субъективных факторов, в том числе от общей организации работы в конкретном подразделении.

По степени распространения и доступности выделяются программные средства, поэтому далее они рассматриваются более подробно (см. "Стандартные методы шифрования и криптографические системы" и "Программные средства защиты информации"). Другие средства применяются в тех случаях, когда требуется обеспечить дополнительный уровень защиты информации.

Шифрование данных представляет собой разновидность программных средств защиты информации и имеет особое значение на практике как единственная надежная защита информации, передаваемой по протяженным последовательным линиям, от утечки. Шифрование образует последний, практически непреодолимый "рубеж" защиты от НСД. Понятие "шифрование" часто употребляется в связи с более общим понятием криптографии. Криптография включает способы и средства обеспечения конфиденциальности информации (в том числе с помощью шифрования) и аутентификации. Конфиденциальность – защищенность информации от ознакомления с ее содержанием со стороны лиц, не имеющих права доступа к ней. В свою очередь аутентификация представляет собой установление подлинности различных аспектов информационного взаимодействия: сеанса связи, сторон (идентификация), содержания (имитозащита) и источника (установление авторства с помощью цифровой подписи).

Число используемых программ шифрования ограничено, причем часть из них являются стандартами де-факто или де-юре. Однако даже если алгоритм шифрования не представляет собой секрета, произвести дешифрование

(расшифрование) без знания закрытого ключа чрезвычайно сложно. Это свойство в современных программах шифрования обеспечивается в процессе многоступенчатого преобразования исходной открытой информации (plain text в англоязычной литературе) с использованием ключа (или двух ключей – по одному для шифрования и дешифрования). В конечном счете, любой сложный метод (алгоритм) шифрования представляет собой комбинацию относительно простых методов.

Классические алгоритмы шифрования данных

Имеются следующие "классические" методы шифрования:

- подстановка (простая – одноалфавитная, многоалфавитная однопетлевая, многоалфавитная многопетлевая);
- перестановка (простая, усложненная);
- гаммирование (смешивание с короткой, длинной или неограниченной маской).

Устойчивость каждого из перечисленных методов к дешифрованию без знания ключа характеризуется количественно с помощью показателя Sк, представляющего собой минимальный объем зашифрованного текста, который может быть дешифрован посредством статистического анализа.

Подстановка предполагает использование альтернативного алфавита (или нескольких) вместо исходного.

Стандартные методы шифрования (национальные или международные) для повышения степени устойчивости к дешифрованию реализуют несколько этапов (шагов) шифрования, на каждом из которых используются различные "классические" методы шифрования в соответствии с выбранным ключом (или ключами). Существуют две принципиально различные группы стандартных методов шифрования:

- шифрование с применением одних и тех же ключей (шифров) при шифровании и дешифровании (симметричное шифрование или системы с закрытыми ключами – private-key systems);
- шифрование с использованием открытых ключей для шифрования и закрытых – для дешифрования (несимметричное шифрование или системы с открытыми ключами – public-key systems).

Строгое математическое описание алгоритмов стандартных методов шифрования слишком сложно. Для пользователей важны в первую очередь "потребительские" свойства различных методов (степень устойчивости к дешифрованию, скорость шифрования и дешифрования, порядок и удобство распространения ключей), которые и рассматриваются ниже.

Для дальнейшего повышения устойчивости к дешифрованию могут применяться последовательно несколько стандартных методов или один метод шифрования (но с разными ключами).

Стандартные методы шифрования и криптографические системы

Стандарт шифрования США DES (Data Encryption Standard – стандарт шифрования данных) относится к группе методов симметричного шифрования и действует с 1976 г. Число шагов – 16. Длина ключа – 64 бита, из которых 8 бит – проверочные разряды четности/нечетности. Долгое время степень устойчивости к дешифрованию этого метода считалась достаточной, однако в настоящее время он устарел. Вместо DES предлагается "тройной DES" – 3DES, в котором алгоритм DES используется 3 раза, обычно в последовательности "шифрование – дешифрование – шифрование" с тремя разными ключами на каждом этапе.

Надежным считается алгоритм IDEA (International Data Encryption Algorithm), разработанный в Швейцарии и имеющий длину ключа 128 бит.

Отечественный ГОСТ28147-89 – это аналог DES, но с длиной ключа 256 бит, так что его степень устойчивости к дешифрованию изначально существенно выше. Важно также и то, что в данном случае предусматривается целая система защиты, которая преодолевает "родовой" недостаток симметричных методов шифрования – возможность подмены сообщений. Такие усовершенствования, как имитовставки, хэш-функции и электронные цифровые подписи позволяют "авторизовать" передаваемые сообщения.

К достоинствам симметричных методов шифрования относится высокая скорость шифрования и дешифрования, к недостаткам – малая степень защиты в случае, если ключ стал доступен третьему лицу.

Довольно популярны, особенно при использовании электронной почты в Интернет, несимметричные методы шифрования или системы с открытыми ключами – public-key systems. К этой группе методов относится, в частности, PGP (Pretty Good Privacy – достаточно хорошая секретность). Каждый пользователь имеет пару ключей. Открытые ключи предназначены для шифрования и свободно рассылаются по сети, но не позволяют произвести дешифрование. Для этого нужны секретные (закрытые) ключи. Принцип шифрования в данном случае основывается на использовании так называемых односторонних функций. Прямая функция x >> f(x) легко вычисляется на основании открытого алгоритма (ключа). Обратное преобразование. f(x) >> x без знания закрытого ключа затруднено и должно занимать довольно длительное время, которое и определяет степень "трудновычислимости" односторонней функции.

Идея системы с открытыми ключами может быть пояснена следующим образом (табл. 11). Для шифрования сообщений можно взять обычную телефонную книгу, в которой имена абонентов расположены в алфавитном порядке и предшествуют телефонным номерам. У пользователя имеется возможность выбора соответствия между символом в исходном тексте и именем абонента, то есть это многоалфавитная система. Ее степень устойчивости к дешифрованию выше. Легальный пользователь имеет "обратный" телефонный справочник, в котором в первом столбце располагаются телефонные номера по возрастанию, и легко

производит дешифрование. Если же такового нет, то пользователю предстоит утомительное и многократное просматривание доступного прямого справочника в поисках нужных телефонных номеров. Это и есть практическая реализация трудно-вычислимой функции. Сам по себе метод шифрования на основе телефонных справочников вряд ли перспективен хотя бы из-за того, что никто не мешает потенциальному взломщику составить "обратный" телефонный справочник. Однако в используемых на практике методах шифрования данной группы в смысле надежности защиты все обстоит благополучно.

Таблица 11 - Пример шифрования в системе с открытыми ключами

Исходное слово	Выбранное имя абонента	Зашифрованное сообщение (телефонные номера)
S	Scott	3541920
A	Adleman	4002132
U	Ullman	7384502
N	Nivat	5768115
A	Aho	7721443

Другая известная система с открытыми ключами – RSA.

Несимметричные методы шифрования имеют преимущества и недостатки, обратные тем, которыми обладают симметричные методы. В частности, в несимметричных методах с помощью посылки и анализа специальных служебных сообщений может быть реализована процедура аутентификации (проверки легальности источника информации) и целостности (отсутствия подмены) данных. При этом выполняются операции шифрования и дешифрования с участием открытых ключей и секретного ключа данного пользователя. Таким образом, несимметричные системы можно с достаточным основанием отнести к полноценным криптографическим системам. В отличие от симметричных методов шифрования, проблема рассылки ключей в несимметричных методах решается проще – пары ключей (открытый и закрытый) генерируются "на месте" с помощью специальных программ. Для рассылки открытых ключей используются такие технологии как LDAP (Lightweight Directory Access Protocol – протокол облегченного доступа к справочнику). Рассылаемые ключи могут быть предварительно зашифрованы с помощью одного из симметричных методов шифрования.

Традиционные и обязательные для современных криптографических систем способы обеспечения аутентификации и проверки целостности получаемых данных (хэш-функции и цифровые подписи), которые реализуются непосредственными участниками обмена, не являются единственно возможными. Распространен также способ, осуществляемый с участием сторонней организации, которой доверяют все участники обменов. Речь идет об использовании так называемых цифровых

сертификатов – посылаемых по сети сообщений с цифровой подписью, удостоверяющей подлинность открытых ключей.

Программные средства защиты информации

Встроенные средства защиты информации в сетевых ОС доступны, но не всегда, как уже отмечалось, могут полностью решить возникающие на практике проблемы. Например, сетевые ОС NetWare 3.x, 4.x позволяют осуществить надежную "эшелонированную" защиту данных от аппаратных сбоев и повреждений. Система SFT (System Fault Tolerance – система устойчивости к отказам) компании Novell включает три основные уровня:

SFT Level I предусматривает, в частности, создание дополнительных копий FAT и Directory Entries Tables, немедленную верификацию каждого вновь записанного на файловый сервер блока данных, а также резервирование на каждом жестком диске около 2% от объема диска. При обнаружении сбоя данные перенаправляются в зарезервированную область диска, а сбойный блок помечается как "плохой" и в дальнейшем не используется.

SFT Level II содержит дополнительные возможности создания "зеркальных" дисков, а также дублирования дисковых контроллеров, источников питания и интерфейсных кабелей.

SFT Level III позволяет применять в локальной сети дублированные серверы, один из которых является "главным", а второй, содержащий копию всей информации, вступает в работу в случае выхода "главного" сервера из строя.

Система контроля и ограничения прав доступа в сетях NetWare (защита от несанкционированного доступа) также содержит несколько уровней:

- уровень начального доступа (включает имя и пароль пользователя, систему учетных ограничений – таких как явное разрешение или запрещение работы, допустимое время работы в сети, место на жестком диске, занимаемое личными файлами данного пользователя, и т.д.);
- уровень прав пользователей (ограничения на выполнение отдельных операций и/или на работу данного пользователя, как члена подразделения, в определенных частях файловой системы сети);
- уровень атрибутов каталогов и файлов (ограничения на выполнение отдельных операций, в том числе удаления, редактирования или создания, идущие со стороны файловой системы и касающиеся всех пользователей, пытающихся работать с данными каталогами или файлами);
- уровень консоли файл-сервера (блокирование клавиатуры файл-сервера на время отсутствия сетевого администратора до ввода им специального пароля).

Однако полагаться на эту часть системы защиты информации в ОС NetWare можно не всегда. Свидетельством тому являются многочисленные инструкции в Интернете и готовые доступные программы, позволяющие взломать те или иные элементы защиты от несанкционированного доступа.

То же замечание справедливо по отношению к более поздним версиям ОС NetWare (вплоть до последней 6-й версии) и к другим "мощным" сетевым ОС со встроенными средствами защиты информации (Windows NT, UNIX). Дело в том, что защита информации – это только часть тех многочисленных задач, которые решаются сетевыми ОС. Усовершенствование одной из функций в ущерб другим (при понятных разумных ограничениях на объем, занимаемый данной ОС на жестком диске) не может быть магистральным направлением развития таких программных продуктов общего назначения, которыми являются сетевые ОС. В то же время в связи с остротой проблемы защиты информации наблюдается тенденция интеграции (встраивания) отдельных, хорошо зарекомендовавших себя и ставших стандартными средств в сетевые ОС, или разработка собственных "фирменных" аналогов известным программам защиты информации. Так, в сетевой ОС NetWare 4.1 предусмотрена возможность кодирования данных по принципу "открытого ключа" (алгоритм RSA) с формированием электронной подписи для передаваемых по сети пакетов.

Специализированные программные средства защиты информации от несанкционированного доступа обладают в целом лучшими возможностями и характеристиками, чем встроенные средства сетевых ОС. Кроме программ шифрования и криптографических систем, существует много других доступных внешних средств защиты информации. Из наиболее часто упоминаемых решений следует отметить следующие две системы, позволяющие ограничить и контролировать информационные потоки.

Firewalls – брандмауэры (дословно firewall – огненная стена). Между локальной и глобальной сетями создаются специальные промежуточные серверы, которые инспектируют и фильтруют весь проходящий через них трафик сетевого/транспортного уровней. Это позволяет резко снизить угрозу несанкционированного доступа извне в корпоративные сети, но не устраняет эту опасность полностью. Более защищенная разновидность метода – это способ маскарада (masquerading), когда весь исходящий из локальной сети трафик посылается от имени firewall-сервера, делая локальную сеть практически невидимой.

Proxy-servers (proxy – доверенность, доверенное лицо). Весь трафик сетевого/транспортного уровней между локальной и глобальной сетями запрещается полностью – маршрутизация как таковая отсутствует, а обращения из локальной сети в глобальную происходят через специальные серверы-посредники. Очевидно, что при этом обращения из глобальной сети в локальную становятся невозможными в принципе. Этот метод не дает достаточной защиты против атак на более высоких уровнях – например, на уровне приложения (вирусы, код Java и JavaScript).

Тема 22. Безопасность ЛВС при взаимодействии с Интернет

В наше время в деятельности любого коммерческого предприятия очень большую важность имеет защита информации. Информация сегодня – ценные ресурс, от которого зависит как функционирование предприятия в целом, так и его конкурентоспособность. Угроз безопасности информационных ресурсов предприятия много – это и компьютерные вирусы, которые могут уничтожить важные данные, и промышленный шпионаж со стороны конкурентов преследующих своей целью получение незаконного доступа к информации представляющей коммерческую тайну, и много другое. Поэтому особое место приобретает деятельность по защите информации, по обеспечению информационной безопасности.

Информационная безопасность (англ. «Information security») – защищенность информации и соответствующей инфраструктуры от случайных или преднамеренных воздействий сопровождающихся нанесением ущерба владельцам или пользователям информации. Информационная безопасность – обеспечение конфиденциальности, целостности и доступности информации. Цель защиты информации – минимизация потерь, вызванных нарушением целостности или конфиденциальности данных, а также их недоступности для потребителей.

Основные типы угроз информационной безопасности: 1. Угрозы конфиденциальности – несанкционированный доступ к данным (например, получение посторонними лицами сведений о состоянии счетов клиентов банка). 2. Угрозы целостности – несанкционированная модификация, дополнение или уничтожение данных (например, внесение изменений в бухгалтерские проводки с целью хищения денежных средств). 3. Угрозы доступности – ограничение или блокирование доступа к данным (например, невозможность подключится к серверу с базой данных в результате DDoS-атаки).

Источники угроз: 1. Внутренние: а) ошибки пользователей и сисадминов; б) ошибки в работе ПО; в) сбои в работе компьютерного оборудования; г) нарушение сотрудниками компании регламентов по работе с информацией. 2. Внешние угрозы: а) несанкционированный доступ к информации со стороны заинтересованных организаций и отдельных лица (промышленный шпионаж конкурентов, сбор информации спецслужбами, атаки хакеров и т.п.); б) компьютерные вирусы и иные вредоносные программы; в) стихийные бедствия и техногенные катастрофы (например, ураган может нарушить работу телекоммуникационной сети, а пожар уничтожить сервера с важной информацией).

Методы обеспечения безопасности информации в ИС:

Препятствие - физическое преграждение пути злоумышленнику к защищаемой информации (например, коммерчески важная информация хранится на сервере внутри здания компании, доступ в которое имеют только ее сотрудники).

Управление доступом – регулирование использования информации и доступа к ней за счет системы идентификации пользователей, их опознавания, проверки полномочий и т.д. (например, когда доступ в отдел или на этаж с компьютерами, на которых хранится секретная информация, возможен только по специальной карточке-пропуску. Или когда каждому сотруднику выдается персональный логин и пароль для доступа к базе данных предприятия с разными уровнями привилегий).

Криптография – шифрование информации с помощью специальных алгоритмов (например, шифрование данных при их пересылке по Интернету; или использование электронной цифровой подписи).

Противодействие атакам вредоносных программ (англ. «malware») – предполагает использование внешних накопителей информации только от проверенных источников, антивирусных программ, брандмауэров, регулярное выполнение резервного копирования важных данных и т.д. (вредоносных программ очень много и они делятся на ряд классов: вирусы, эксплойты, логические бомбы, трояны, сетевые черви и т.п.).

Регламентация – создание условий по обработке, передаче и хранению информации, в наибольшей степени обеспечивающих ее защиту (специальные нормы и стандарты для персонала по работе с информацией, например, предписывающие в определенные числа делать резервную копию электронной документации, запрещающие использование собственных флеш-накопителей и т.д.).

Принуждение – установление правил по работе с информацией, нарушение которых карается материальной, административной или даже уголовной ответственностью (штрафы, закон «О коммерческой тайне» и т.п.).

Побуждение – призыв к персоналу не нарушать установленные порядки по работе с информацией, т.к. это противоречит сложившимся моральным и этическим нормам (например, Кодекс профессионального поведения членов «Ассоциации пользователей ЭВМ США»).

Средства защиты информации:

Технические (аппаратные) средства – сигнализация, решетки на окнах, генераторы помех воспрепятствования передаче данных по радиоканалам, электронные ключи и т.д.

Программные средства – программы-шифровальщики данных, антивирусы, системы аутентификации пользователей и т.п.

Смешанные средства – комбинация аппаратных и программных средств. Организационные средства – правила работы, регламенты, законодательные акты в сфере защиты информации, подготовка помещений с компьютерной техникой и прокладка сетевых кабелей с учетом требований по ограничению доступа к информации и пр.

Использование межсетевых экранов для защиты локальных сетей

Одним из самых популярных методов защиты локальной сети от атак извне является использование межсетевого экрана (МСЭ). Межсетевой экран(firewall,

брандмауэр) представляет собой программную или программно-аппаратную систему, которая устанавливается на границе охраняемой вычислительной сети и осуществляет фильтрацию сетевого трафика в обе стороны, разрешая или запрещая прохождение определенных пакетов внутрь локальной сети (в периметр безопасности) или из нее в зависимости от выбранной политики безопасности. Однако, только фильтрацией пакетов задачи современных МСЭ не ограничиваются, они выполняют также множество дополнительных действий:

- кэширование информации, когда часть полученной из внешней сети информации временно сохраняется на локальных запоминающих устройствах, что позволяет экономить время и потребляемый трафик при повторном обращении к той же информации;
- трансляция адресов, позволяющая использовать для внешних коммуникаций компьютеров локальной сети только один IP-адрес – адрес самого брандмауэра, внутренние адреса локальной сети могут быть любыми;
- переадресация, позволяющая отправлять пакеты, приходящие на некоторый IP-адрес, на компьютер с другим адресом. Это позволяет, например, распределить нагрузку на Web-сервер.

Существуют два основных типа МСЭ: пакетные фильтры и шлюзы приложений. При этом оба типа могут быть реализованы одновременно в одном брандмауэре. Пакетные фильтры (packet filter) представляют собой сетевые маршрутизаторы, которые принимают решение о том, пропускать или блокироватьпакет на основании информации в его заголовке (рис.55)

Пакетные фильтры работают с информацией в заголовках IP, ICMP, TCP и UDP- пакетов. Правила фильтрации пакетов задаются на основе следующих данных:

- название сетевого интерфейса и направление передачи информации;
- IP-адреса отправителя и получателя;
- протокол более высокого уровня (используется TCP или UDP);
- порт отправителя и получателя для протоколов TCP и UDP;
- опции IP (например, блокировка маршрутизации от источника);
- тип сообщения ICMP.

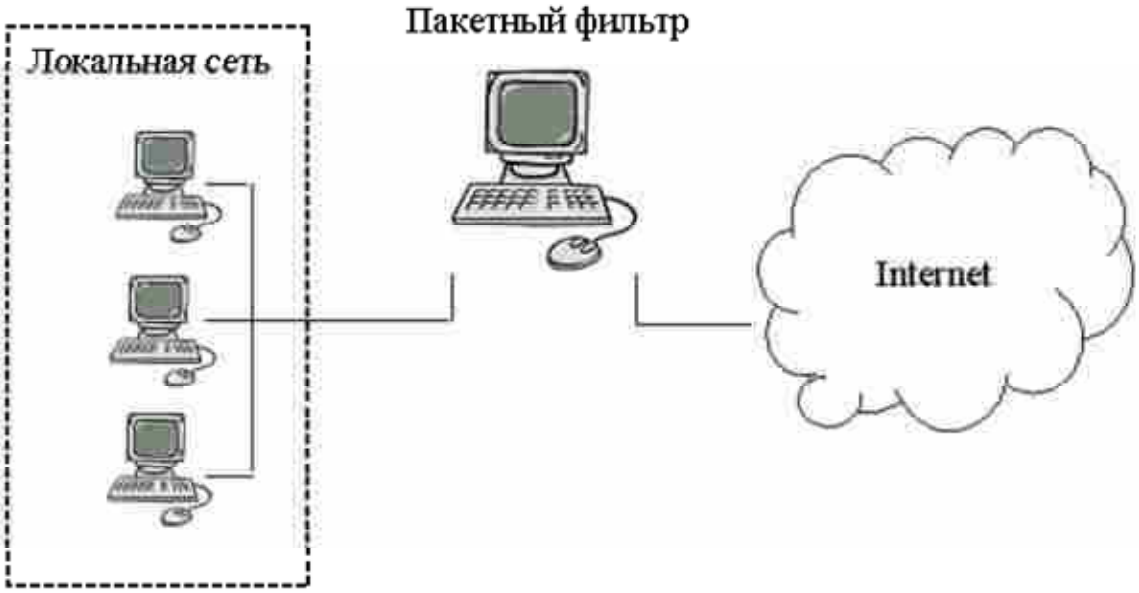


Рисунок 55 - Пакетный фильтр на границе локальной сети

При определении правил фильтрации необходимо придерживаться одной из двух стратегий политики безопасности:

1. Разрешить весь трафик, не запрещенный правилами фильтрации.

2. Запретить весь трафик, не разрешенный правилами фильтрации.

С точки зрения безопасности более предпочтительной является вторая стратегия, согласно которой задаются правила, разрешающие прохождение пакетов определенного типа, прохождение остальных пакетов запрещается. Это связано с тем, что, во-первых, количество запрещенных пакетов обычно гораздо больше, чем количество разрешенных, а во-вторых, со временем могут появиться новые службы, для которых при использовании первой стратегии необходимо будет дописывать запрещающие правила (если доступ к ним, конечно, нежелателен), в то время как вторая стратегия запретит доступ к ним автоматически.

Пакетные фильтры классифицируются на фильтры без памяти и фильтры с памятью (динамические). Первые из них фильтруют информацию только исходя из информации в заголовке рассматриваемого пакета. Динамические же пакеты учитывают при фильтрации текущее состояние соединений, формируя таблицы входящих и исходящих пакетов, и принимают решение на основании информации в нескольких взаимосвязанных пакетах.

Кроме того, пакетные фильтры могут реализовывать также множество дополнительных возможностей. Например, перенаправление пакетов, дублирование пакетов, подсчет трафика, ограничение полосы пропускания, запись пакетов в файл протокола и многое другое. Настройка пакетного фильтра требует от администратора значительной квалификации и понимания принципов работы всех протоколов стека TCP/IP от протоколов прикладного уровня до протоколов сетевого уровня.

Большинство современных операционных систем имеют встроенные пакетные фильтры, например ipfw в Unix или Internet Connection Firewall в Windows. Функции пакетного фильтра могут выполнять и аппаратные маршрутизаторы, например, CISCO PIX 515E.

Главным недостатком пакетных фильтров является невозможность осуществления фильтрации пакетов по содержимому информационной части пакетов, то есть по данным, относящимся к пакетам более высокого уровня. Этот недостаток может быть устранен путем использования шлюзов приложений (application gateway, proxy-server). Proxy-серверы работают на прикладном уровне, обеспечивая работу той или иной сетевой службы. При этом в отличие от пакетных фильтров, которые лишь перенаправляют пакет из одной сети в другую, proxy-серверы принимают запрос от клиента и направляют его во внешнюю сеть от своего имени, разрывая таким образом нормальный сетевой трафик (см. рис.56). Поэтому брандмауэр в виде шлюза приложений может быть реализован на компьютере всего с одним сетевым интерфейсом.

Рисунок 56 – Выполонение запроса через proxy сервер

Поясним схему на рис.56. Клиент формирует запрос на сервер какой-либо службы в Internet (например, запрос на внешний Web-сервер). Запрос поступает на proxy-сервер (конфигурация брандмауэра должна быть такова, чтобы все запросы к какой-либо службе в Internet обязательно поступали на соответствующий proxy-сервер). Приняв запрос от клиента, proxy-сервер проверяет его по заданным правилам фильтрации содержимого пакета и, если запрос не содержит запрещенных параметров, формирует пакет с запросом уже от своего имени (со своим обратным адресом) внешнему серверу. Ответ от внешнего сервера поступает, очевидно, на имя proxy-сервера. Пройдя проверку, аналогичную запросу, ответ может быть принят либо отвергнут. Если ответ принят -ответ направляется на адрес клиента, первоначально сформировавшего запрос.

Фильтрация содержимого может осуществляться по множеству параметров:

– IP-адрес отправителя и получателя;
– запрашиваемый URL;
– наличие вложений (приложения Java, компоненты ActiveX) и т.п.
– время запроса и другие, в зависимости от используемого proxy-сервера.

Важной функцией современных proxy-серверов является трансляция сетевых адресов (Network Address Translation, NAT), которая подразумевает замену адреса клиента в запросе во внешнюю сеть на собственный адрес (или несколько адресов) proxy-сервера. Это позволяет скрыть от посторонних структуру внутренней сети, список используемых в ней адресов. С другой стороны это позволяет иметь на всю локальную сеть лишь один легальный IP-адрес, который должен быть присвоен proxy-серверу. Рабочие станции внутри сети могут иметь любые IP-адреса, в том числе и те, которые запрещено использовать во внешней сети. NAT может быть организована по статической и динамической схеме. При статической трансляции адрес клиента в локальной сети привязывается к конкретному адресу, который транслируется во внешнюю сеть. Динамическая трансляция предполагает наличие диапазона доступных внешних адресов и при каждом запросе клиента proxy-сервер выделяет один из свободных адресов для представления клиента во внешней сети, по окончании транзакции этот адрес возвращается в список свободных и может быть использован в дальнейшем для передачи запроса другого клиента. Развитием идеи NAT стала трансляция адресов портов (Network Address Port Translation, NAPT), когда один и тот же IP-адрес распределяется при трансляции на несколько пользователей и каждому пользователю сопоставляется в отправляемом во внешнюю сеть пакете уникальная комбинация IP-адреса и номера порта отправителя. Иными словами, различным пользователям сети proxy-сервер сопоставляет один и тот же IP-адрес, но присваивает различные номера портов в исходящих запросах. Это стало возможным за счет того, что порт отправителя зачастую не несет в запросе никакой полезной информации и может быть использован для уникальной идентификации клиента локальной сети для proxy-сервера.

Современные proxy-серверы выполняют обычно еще одну важную функцию – кэширование информации. Информация, пришедшая на proxy-сервер, сохраняется на локальных запоминающих устройствах, и при очередном запросе клиента запрашиваемая им информация сначала ищется в локальной памяти, и только если ее там нет - запрос передается во внешнюю сеть. Это позволяет уменьшить объем трафика, потребляемого из внешней сети, а также уменьшить время доступа к информации для конечного клиента.

Рассмотрим свойства наиболее распространенных proxy-серверов.

Microsoft Proxy Server (версия 2.0) представляет собой брандмауэр с расширяемым набором функций и сервер кэширования информации, обеспечивает поддержку протоколов HTTP и gopher, а также поддержку клиентских приложений (например, Telnet и RealAudio) для компьютеров интрасети, использующих

протоколы TCP/IP или IPX/SPX, поддерживает VPN, выполняет функции фильтра пакетов. Работает в среде Windows.

Squid - высокопроизводительный кэширующий proxy-сервер для web-клиентов с поддержкой протоколов FTP, gopher и HTTP, имеющий реализации как под Unix, так и под Windows- платформы.Squid хранит метаданные и особенно часто запрашиваемые объекты в ОЗУ, кэширует DNS-запросы, поддерживает неблокирующие DNS-запросы и реализует негативное кэширование неудачных запросов. Поддерживает протокол ICP (Internet Casing Protocol), позволяющий организовывать нескольким серверам иерархические структуры кэширования.

Помимо перечисленных, на практике могут быть использованы так называемые персональные межсетевые экраны. Они устанавливаются на компьютер пользователя, и все правила безопасности задаются для обмена этого компьютера с внешней сетью. Это позволяет настроить политику безопасности персонально под каждого пользователя непосредственно на его рабочей станции. Примером подобного МСЭ является брандмауэр AtGuard, который включает в себя функции proxy-сервера и локального пакетного фильтра. AtGuard способен блокировать баннеры, файлы cookie, Java -скрипты и апплеты, а также элементы ActiveX. Еще одна особенность AtGuard – способность работать в режиме обучения, когда при каждой попытке подключиться к какому-либо порту запрашивается разрешение на установку соединения, и сделанный пользователем выбор становится правилом для дальнейшей работы программы.

Используемые на практике МСЭ представляют собой интегрированную систему защиты, включающую и пакетный фильтр, и proxy-сервер. Они могут располагаться как на одном, так и на нескольких компьютерах, в связи с чем существует возможность выбора архитектуры используемого МСЭ [8].

Архитектура с использованием в качестве МСЭ компьютера с двумя сетевыми интерфейсами похожа на схему подключения пакетного фильтра (рис. 57), но на МСЭ должна быть отключена возможность маршрутизации пакетов. Это позволяет полностью блокировать трафик во внешнюю сеть на этом компьютере, а все необходимые сервисы должны обеспечиваться proxy-серверами, работающими на двухканальном компьютере. Для обеспечения дополнительной защиты можно поместить маршрутизатор с фильтрацией пакетов между внешней сетью и двухканальным компьютером. Архитектура с экранированным узлом предполагает использование одновременно и пакетного фильтра, и proxy-сервера (рис.5.5).

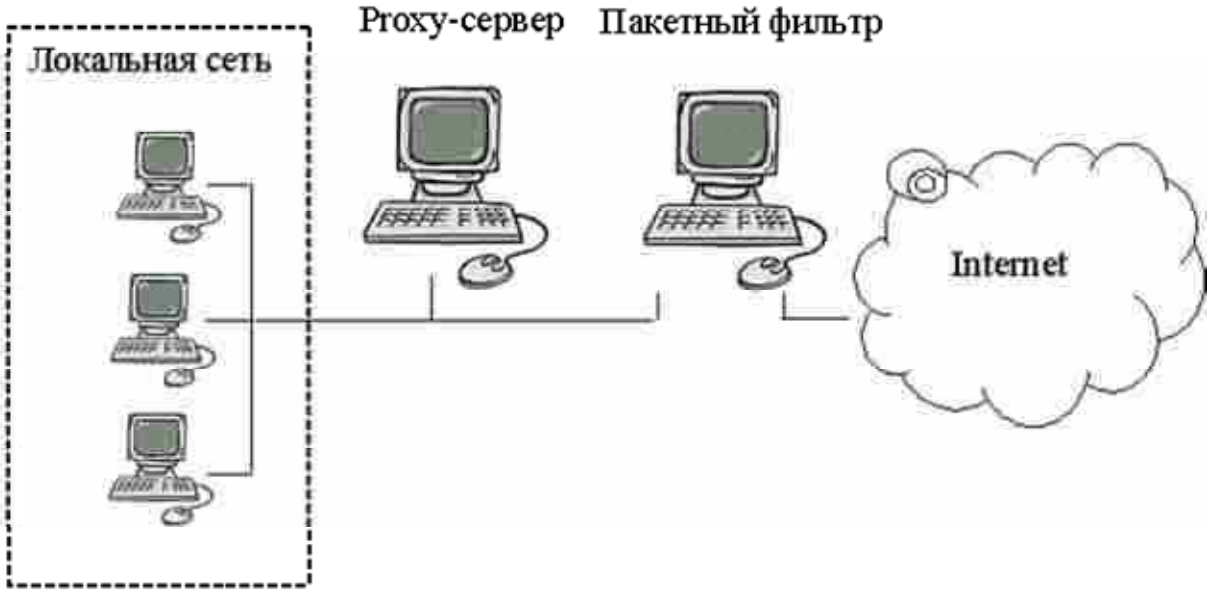


Рисунок 57 – МСЭ с использованием экранированного узла

На границе с внешней сетью устанавливается пакетный фильтр, который должен блокировать потенциально опасные пакеты, чтобы они не достигли прикладного шлюза (proxy-сервера) и локальной сети. Он отвергает или пропускает трафик в соответствии со следующими правилами:

- трафик из внешней сети к прикладному шлюзу пропускается;
- прочий трафик из внешней сети блокируется;
- пакетный фильтр блокирует любой трафик из локальной сети во внешнюю, если он не идет от прикладного шлюза.

Прикладной шлюз должен обеспечивать функции proxy-сервера для всех потенциально опасных служб и для работы ему достаточно одного сетевого интерфейса. Подобная схема подключения брандмауэра отличается большей гибкостью по сравнению с двухканальным МСЭ, поскольку пакетный фильтр может позволить пропустить запросы к надежным сервисам в обход прикладного шлюза. Этими надежными сервисами могут быть те сервисы, для которых нет proxy-сервера, и которым можно доверять в том смысле, что риск использования этих сервисов считается приемлемым.

Развитием концепции изолированного узла стала архитектура с изолированной подсетью. Здесь используется два пакетных фильтра (рис.58)для организации изолированной подсети, которую еще называют демилитаризованной зоной (DMZ).

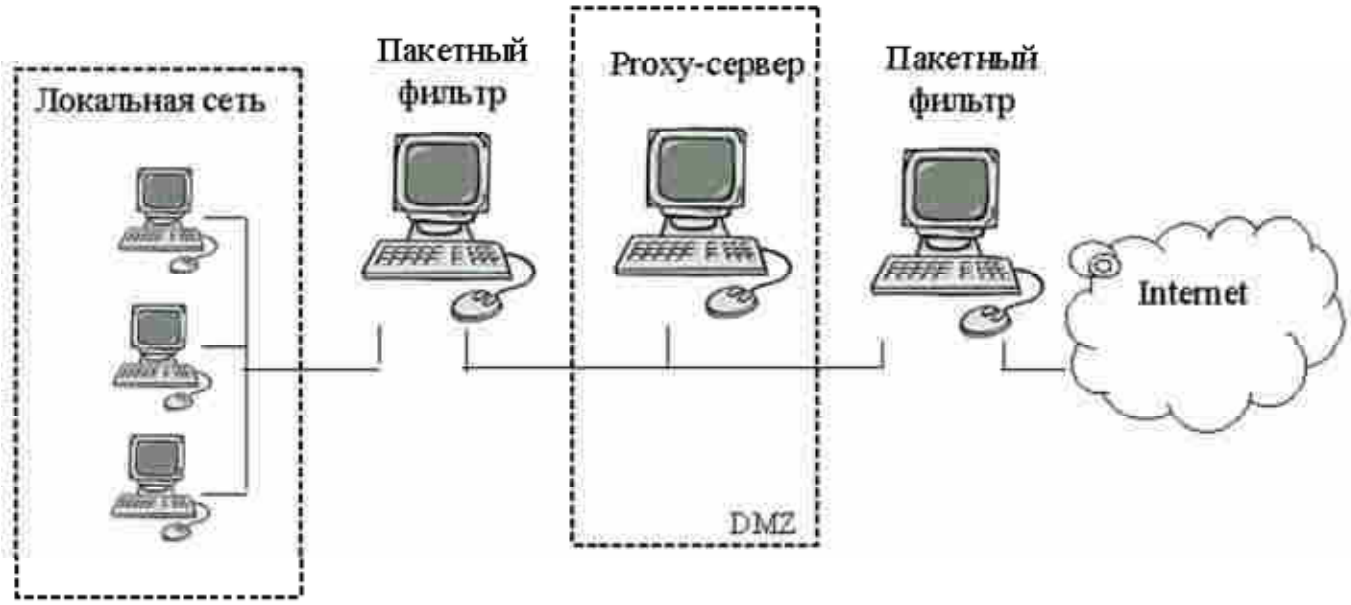


Рисунок 58 – МСЭ с использованием экранированной подсети

Внутри изолированной подсети должен находиться прикладной шлюз, а также могут находиться различные информационные серверы (mail, WWW, FTP), модемные пулы и т.п. Пакетный фильтр, установленный на границе с внешней сетью, должен фильтровать пакеты по следующим правилам:

– пропускать прикладной трафик от прикладного шлюза во внешнюю сеть;
– пропускать прикладной трафик из внешней сети к прикладному шлюзу;
– все остальные виды трафика блокировать.

Внутренний пакетный фильтр управляет трафиком «локальная сеть – демилитаризованная зона» согласно следующим правилам:

– трафик от прикладного шлюза к внутренним системам пропускается;
– трафик к прикладному шлюзу от внутренних систем пропускается;
– трафик к информационным серверам внутри DMZ пропускается;
– все остальные виды трафика блокировать.

Такая схема дает возможность еще более гибко формировать политику безопасности, задавая различные правила фильтрации для двух пакетных фильтров (например, можно разрешить прохождения FTP-пакетов в DMZ из локальной сети для обновления информации на WWW-сервере и запретить доступ по FTP-протоколу к DMZ из внешней сети). Компьютеры, расположенные в демилитаризованной зоне, подвержены большему количеству атак, чем компьютеры локальной сети. Поэтому все компьютеры, находящиеся в DMZ, должны быть максимально укреплены (bastion host, укрепленный компьютер). На них должны быть удалены все неиспользуемые службы, максимально активизированы средства безопасности операционных систем (ужесточаются права доступа к объектам, минимизируется количество зарегистрированных субъектов,ведется строгий аудит).

Очевидно, что можно создать несколько экранированных подсетей, отделенных друг от друга собственным пакетным фильтром с определением правил доступа для каждой из подсетей. Выбор конкретной архитектуры МСЭ зависит от стоящих перед администратором задач, условий функционирования, стоимости того или иного решения

Современные информационные системы функционируют в условиях постоянных угроз, исходящих из сети Интернет. Для защиты от этих угроз существует множество средств на различных уровнях. Для обеспечения конфиденциальности ицелостностипередаваемой по глобальной сети информации можно использовать защищенные сетевые протоколы, которые используют криптографические методы. Для предотвращениявторжения извне, защиты от атак типа DoS необходимо использовать межсетевые экраны, основная задача которых – фильтрация входящего и исходящего сетевого трафика в зависимости от принятой политики безопасности.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

Основная литература

1. Олифер В.Г., Олифер Н.А. Компьютерные сети. Принципы, технологии, протоколы: Ученик для вузов. 4-е изд. – СПб.: Питер, 2015.
2. Таненбаум Э. Компьютерные сети. – М. «Вильямс», 2011

Дополнительная литература

3. Администрирование сети на основе Windows 2000. Учебный курс MCSE. Сертификационный экзамен 70-216. - СПб.: БХВ-Петербург, 2004
4. Вишневский В.М. Теоретические основы проектирования компьютерных сетей. – Москва: Техносфера, 2003.
5. Крелл М., Манн С. Linux. Администрирование сетей TCP/IP. – М. «Вильямс», 2003
6. Кузьменко Н. Компьютерные сети и сетевые технологии. – СПб.: «Наука и Техника», 2013
7. Попов И., Максимов Н. Компьютерные сети. – Москва: «Инфра-М», 2013
8. Стахнов А. Сетевое администрирование Linux. - СПб.: Питер-пресс, 2004
9. Microsoft Windows 2000: Server и Professional. Русские версии / А.Г. Андреев [и др.] Под общ. ред. А.Н. Чекмарева и Д.Б.Вишнякова. – СПб.: БХВ-Петербург, 2003