Министерство науки и высшего образования РФ

Федеральное государственное бюджетное образовательное учреждение высшего образования

ФГБОУ ВО «ДАГЕСТАНСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

КАФЕДРА УиИТСиВТ

УЧЕБНО-МЕТОДИЧЕСКИЕ УКАЗАНИЯ
К ВЫПОЛНЕНИЮ ЛАБОРАТОРНЫХ РАБОТ
ПО ДИСЦИПЛИНЕ "ИНФОРМАЦИОННЫЕ СЕТИ И ТЕЛЕКОММУНИКАЦИИ"

МАХАЧКАЛА 2021

Учебно-методические указания к выполнению лабораторных работ по дисциплине "Информационные сети и телекоммуникации" для бакалавров направления подготовки 27.03.04 - "Управление в технических системах". Махачкала, ИПЦ ДГТУ, 2021.- 40 с.

В данных методических указаниях к выполнению лабораторной работы "Информационные сети и телекоммуникации" для бакалавров направления подготовки 27.03.04 - " Управление в технических системах "в полном объёме выполнена предварительная подготовка к лабораторной работе, в неё входит подготовка отчёта, подготовка ответов на контрольные вопросы и задания.

Составители: Искендерова Э.Т., ассистент
Тетакаев У.Р.,к.т.н., ст.преподаватель

Рецензенты: 1.Хазамова М.А.,к.т.н.,доцент
кафедры ТиОЭ ДГТУ

2. Кобзаренко Д.Н., д.т.н., ведущий научный сотрудник лаборатории комплексного освоенияэнергоресурсов Института проблем геотермии и возобновляемой энергетики филиала ФГБУН ОИВТ РАН

Печатается по постановлению Ученого совета Дагестанского государственного технического университета от_____2021г.

Лабораторная работа № 1.
ПРОТОКОЛ MICROSOFT TCP/IP

Цель работы: ознакомление и изучение протокола TCP/IP.

1. Теоретические сведения

1.1. Определение IP-протокола

Название данного протокола - Internet Protocol - отражает его суть: он должен передавать пакеты между сетями. В каждой очередной сети, лежащей на пути перемещения пакета, протокол IP вызывает средства транспортировки, принятые в этой сети, чтобы с их помощью передать этот пакет на маршрутизатор, ведущий к следующей сети, или непосредственно на узел-получатель.

Протокол IP относится к протоколам без установления соединений. Перед IP не ставится задача надёжной доставки сообщений от отправителя к получателю, он обрабатывает каждый IP-пакет как независимую единицу, не имеющую связи ни с какими другими пакетами. В протоколе IP нет механизмов, обычно применяемых для увеличения достоверности конечных данных: отсутствует квитирование, нет процедуры упорядочивания, повторных передач или других подобных функций. Если во время продвижения пакета произошла какая-либо ошибка, то протокол IP по своей инициативе ничего не предпринимает для её исправления. Все вопросы обеспечения надёжности доставки данных по составной сети в стеке TCP/IP решает протокол TCP, работающий непосредственно над протоколом IP.

Важной особенностью протокола IP, отличающей его от других сетевых протоколов, является его способность выполнять динамическую фрагментацию пакетов при передаче их между сетями с различными значениями полей данных.

Имеется прямая связь между функциональной сложностью протокола и сложностью заголовка пакетов, которые этот пакет используют. Это объясняется тем, что основные служебные данные, на основании которых протокол выполняет то или иное действие, переносятся между двумя модулями, реализующими этот протокол на разных машинах, именно в полях заголовков пакетов.

1.2. Структура IP-пакета

IP-пакет состоит из заголовка и поля данных. Заголовок, как правило, имеющий длину 20 байт, имеет следующую структуру рис.

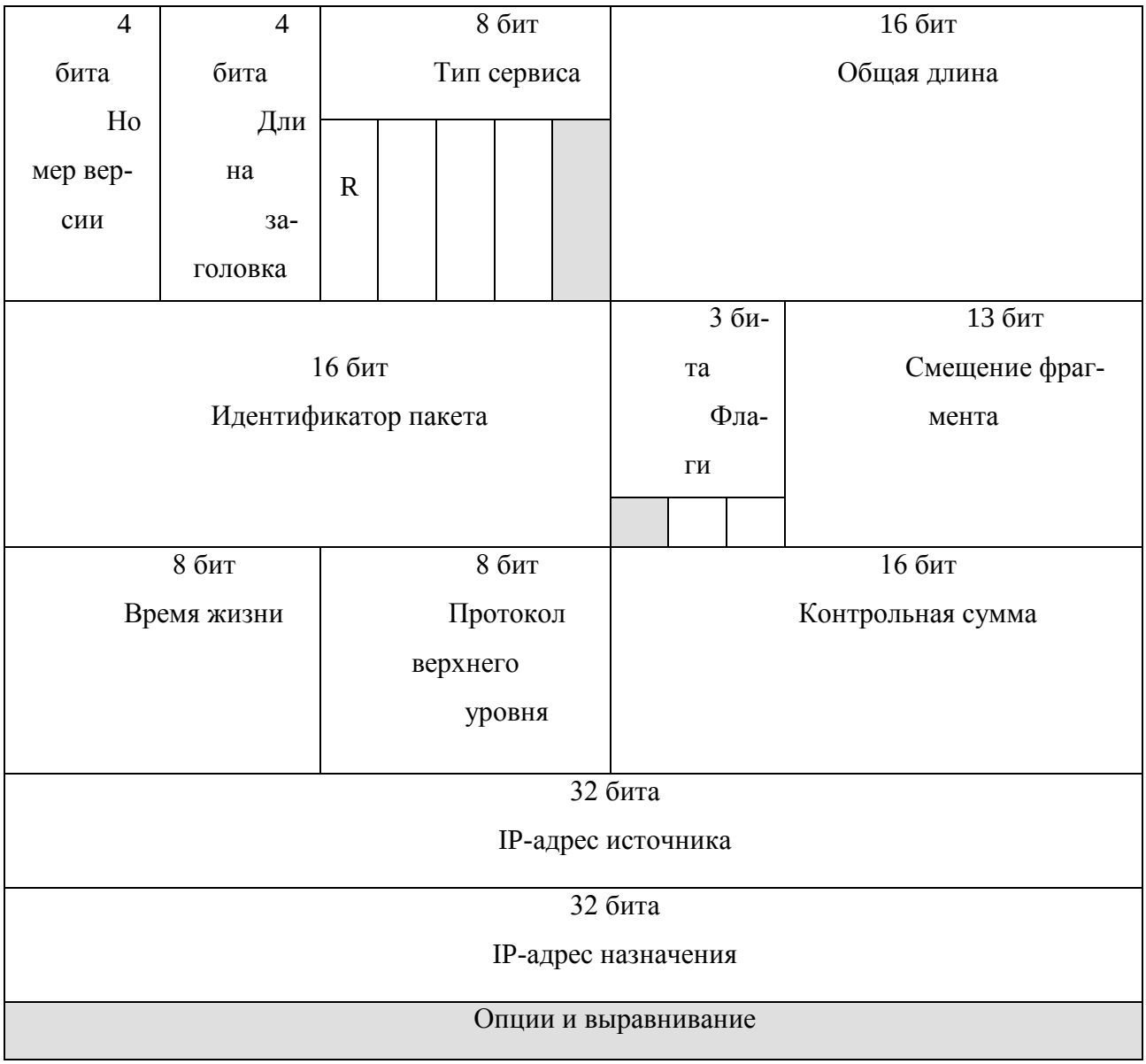


Рис. 1. Структура заголовка IP-пакета.

Поле Номер версии (Version)указывает версию протокола IP, сейчас используется версия IPv4 и готовится переход на версию IPv6.

Поле Длина заголовка (IHL) указывает значение длины заголовка, измеренное в 32-битовых словах. Обычно заголовок имеет длину в 20 байт (пять 32-битовых слов), но при увеличении объёма служебной информации эта длина может быть увеличена за счёт использования дополнительных байт в поле Опции. Наибольший заголовок занимает 60 октетов.

Поле Тип сервиса (Type of Service) занимает один байт и задает приоритетность пакета и вид критерия выбора маршрута. Первые три бита этого поля образуют подполе приоритета пакета (Precedence). Приоритет может иметь значения от самого низкого - 0 (нормальный пакет) до самого высокого - 7 (пакет управляющей информации). Маршрутизаторы и

компьютеры принимают во внимание приоритет пакета и обрабатывают более важные пакеты в первую очередь. Поле Тип сервиса содержит также три бита, определяющие критерий выбора маршрута. Реально выбор осуществляется между тремя альтернативами: малой задержкой, высокой достоверностью и высокой пропускной способностью. Установленный бит D (delay) говорит о том, что маршрут должен выбираться для минимизации задержки доставки данного пакета, бит T - для максимизации пропускной способности, а бит R - для максимизации надёжности доставки. Во многих сетях улучшение одного из этих параметров связано с ухудшением другого, кроме того, обработка каждого из них требует дополнительных вычислительных затрат. Поэтому редко, когда имеет смысл устанавливать одновременно хотя бы два из этих трёх критериев выбора маршрута. Зарезервированные биты имеют нулевое значение.

Поле Общая длина (Total Lenth) означает общую длину пакета с учетом заголовка и поля данных. Максимальная длина пакета ограничена разрядностью поля, определяющего эту величину, и составляет 65 535 байт, однако в большинстве хост-компьютеров и сетей столь большие пакеты не используются. При передаче по сетям различного типа длина пакета выбирается с учетом максимальной длины пакета протокола нижнего уровня, несущего IP-пакеты. Если это кадры Ethernet, то выбираются пакеты с максимальной длиной 1500 байт, умещающиеся в поле данных кадра Ethernet. В стандарте предусматривается, что все хосты должны быть готовы принимать пакеты вплоть до 576 байт длиной (приходят ли они целиком или по фрагментам). Хостам рекомендуется пакеты размером более чем 576 байт, только если они уверены, что принимающий хост или промежуточная сеть готовы обслужить пакет такой длины.

Поле Идентификатор пакета (Identification) используется для распознавания пакетов, образовавшихся путём фрагментации исходного пакета. Все фрагменты должны иметь одинаковое значение этого поля.

Поле Флаги (Flags) содержит признаки, связанные с фрагментацией. Установленный бит D (Do not Fragment) запрещает маршрутизатору фрагментировать данный пакет, а установленный бит M (More Fragments) говорит о том, что данный пакет является промежуточным (не конечным) фрагментом. Оставший ся бит зарезервирован.

Поле Смещение фрагмента (Fragment Offset) задаёт смещение в байтах поля данных этого пакета от начала общего поля данных исходного пакета, подвергнутого фрагментации. Используется при сборке и разборке фрагментов пакетов при передачах их между сетями с различными свойствами. Смещение должно быть кратно 8 байт.

Поле Время жизни (Time to Live) означает предельный срок, в течение которого пакет может перемещаться по сети. Время жизни каждого пакета задаётся источником передачи и

измеряется в секундах. На маршрутизаторах и в других узлах сети по истечении каждой секунды из текущего времени жизни вычитается единица; единица вычитается и в том случае, когда время задержки меньше секунды. Поскольку современные маршрутизаторы редко обрабатывают пакет дольше, чем за одну секунду, то время жизни можно считать равным максимальному числу узлов, которые разрешено пройти данному пакету до того, как он достигнет места назначения. Если параметр времени жизни станет нулевым до того, как пакет достигнет получателя, этот пакет будет уничтожен. Время жизни можно рассматривать как часовой механизм самоуничтожения. Значение этого поля изменяется при обработке заголовка IP-пакета.

Идентификатор Протокол верхнего уровня (Protocol) занимает один байт и указывает, какому протоколу верхнего принадлежит информация, размещения в поле данных пакета, например, это могут быть сегменты протокола TCP, дейтаграммы или иные пакеты.

Контрольная сумма (Header Checksum) рассчитывается только по заголовку. Поскольку некоторые поля заголовка изменяют своё значение в процессе передачи пакета по сети, контрольная сумма проверяется и повторно рассчитывается при каждой обработке IP-заголовка. Контрольная сумма - 16 бит - подсчитывается как дополнение к сумме всех 16-битовых слов заголовка. При её вычислении значение самого поля устанавливается в ноль. Если контрольная сумма не верна, то пакет будет отброшен, как только ошибка будет обнаружена.

Поле Опции (IP Options) является необязательным и используется обычно только при отладке сети. Механизм опций предоставляет функции управления, которые необходимы или просто полезны при определённых ситуациях, однако он не нужен при обычных коммуникациях. Это поле состоит из нескольких подполей, каждое из которых может быть одного из восьми типов. В этих подполях можно учитывать точный маршрут прохождения маршрутизаторов, регистрировать проходимые пакетом маршрутизаторы, помещать данные системы безопасности, а также временные отметки. Так как число подполей может быть произвольным, то в конце поля Опции должно быть добавлено несколько байт для выравнивания заголовка пакета по 32-битной границе.

Поле Выравнивание (Padding) используется для того, чтобы убедиться в том, что IP-заголовок заканчивается на 32-битной границе. Выравнивание осуществляется нулями.

Протокол TCP/IP (Transmission Control Protocol/Internet Protocol) в Windows NT 4.0 обеспечивает сетевое взаимодействие компьютеров под управлением Windows NT, и возможность подключения к ним сетевых устройств под управлением других ОС.

Протокол TCP/IP считается наиболее совершенным и распространенным протоколом из всех доступных на сегодняшний день. Все современные ОС поддерживают протокол

TCP/IP и все сети используют его для обеспечения передачи большей части своих данных. Этот протокол представляет надежную, ориентированную на соединение службу доставки.

Протокол TCP

Данные протокола TCP передаются сегментами, и соединение должно быть установлено до того, как узлы начнут обмениваться данными. TCP обеспечивает надежность, присваивая номер последовательности каждому передаваемому сегменту. Если сегмент разбивается на мелкие пакеты, то узел-получатель сможет узнать, все ли части получены. Для этого используются подтверждения. Для каждого отправленного сегмента узел-получатель должен вернуть отправителю подтверждение в течение определенного времени.

Если отправитель не получил подтверждения, то данные передаются повторно. Если сегмент поврежден, то узел-получатель отвергает его. Поскольку подтверждение в этом случае не посылается, отправитель передает сегмент еще раз.

Приложения идентифицируют себя на компьютере посредством номера порта протокола. Например, FTP-сервер использует определенный TCP-порт, поэтому другие приложения могут связаться с ним.

Порты могут иметь любой номер от 0 до 65536. Номера портов для приложений клиентов динамически назначаются операционной системой при обработке запроса на обслуживание.

Порты протокола TCP

Порт протокола TCP указывает место доставки сообщения. Номера портов, меньшие 256, определены как широко используемые. В таблице перечислены некоторые из таких портов.

Номер порта	Описание
21	FTP
23	Telnet
52	Доменная система имен (DNS)
139	Сервис NetBIOS

Установка связи по протоколу TCP.

Инициализация TCP-соединения происходит в три этапа. Ниже перечислены операции, из которых состоит этот процесс.

1. Узел-отправитель запрашивает соединение, посылая с установленным флагом синхронизации.

2. Узел-адресат подтверждает получение запроса, отправляя обратно сегмент с:

- установленным флагом синхронизации;
- порядковым номером начального байта сегмента, который он может послать, или номером последовательности;
- подтверждением, включающий порядковый номер следующего сегмента, который он ожидает получить.

3. Запрашивающий узел посылает обратно сегмент с подтверждением номера последовательности и номером своего подтверждения (рис.2).

Рис.2.

Для завершения соединения TCP действует аналогично. Это гарантирует, что оба узла закончат передачу и примут все данные.

Структура TCP-пакета

Все пакеты протокола TCP имеют две части – заголовок и даннные. В таблице представленны поля заголовка TCP-пакета.

Поле	Описание
Source Port (Порт отправителя)	TCP – порт узла-отправителя
Destination Port (Порт получателя)	TCP – порт узла-получателя, определяет конечную точку соединения.
Sequence Number (Порядковый номер)	Номер последовательности пакета. Используетсядля проверки получения всех байт соединения.
Data Length (Длина данных)	Длина TCP-пакета
Flags (флаги)	Это поле описывает содержимое сегмента.
Checksum (Контрольная сумма)	Проверяет поврежден ли заголовок.

Протокол IP

Протокол IP не ориентирован на соединение, поскольку он не устанавливает сеанс связи, перед тем как начать обмен данными. Протокол ненадежный – он не гарантирует доставку, хотя делает все возможное для доставки пакета. По пути пакет может быть потерян, доставлен в неправильной последовательности, продублирован или задержан.

Протокол IP не требует подтверждения при приеме данных. Отправитель или получатель не информируется при потере пакета или доставке его в неправильной последовательности. Ответственность за подтверждение получения пакетов несут высокоуровневые транспортные протоколы, например TCP.

Маршрутизация (routing) – процесс выбора пути для передачи пакетов. Маршрутизация осуществляется на узле TCP/IP в момент отправки IP-пакетов, а затем – на IP-маршрутизаторе.

Маршрутизатор (router) – это устройство, которое перенаправляет пакеты из одной физической сети в другую. Маршрутизаторы также называют шлюзами (gateways).

Поля IP-пакета приведены в таблице.

Поле	Описание
Source IP-address (IP-адрес отправителя)	Идентифицирует отправителя пакета при помощи IP-адреса
Destination IP-address (IP-адрес получателя)	Идентифицирует получателя пакета при помощи IP-адреса
Protocol (Протокол)	Информирует протокол IP узла-получателя о том, какому протоколу – TCP или UDP его передать.
Checksum (Контрольная сумма)	Используется для проверки целостности пришедшего пакета.
Time to live, или TTL (Время существования)	Определяет, сколько времени пакет находится в сети, перед тем как он будет отвергнут. Предотвращает бесконечное блуждание пакетов по сети. Маршрутизаторы должны уменьшать TTL на количество секунд, проведенных пакетом в маршрутизаторе. TTL уменьшается по меньшей мере на одну секунду каждый раз, когда пакет проходит через маршрутизаторе. По умолчанию в Windows NT 4.0 TTL равно 128 секундам.

Реализация IP на маршрутизаторе.

Маршрутизатор обрабатывает полученные им IP-пакеты следующим образом:

1. Уменьшает значение TTL на 1 секунду или больше, если пакет надолго задерживается на маршрутизаторе.

Если значение TTL достигает нуля, пакет отвергается.

2. Пакет может быть фрагментирован, если его размер слишком велик для сети дальнейшего следования

3. Если может быть фрагментирован, то IP создает для каждого нового пакета (фрагмента) отдельный заголовок, устанавливая:

- Flag(флаг), указывающий, что существуют и другие фрагменты, которые будут отправлены в след;
- Fragment ID(Идентификатор фрагмента), идентифицирующий все фрагменты, составляющие один пакет;
- Fragment Offset(Смещение фрагмента), обеспечивающий правильную сборку пакета на узле-получателе.

4. Вычисляет новую контрольную сумму.

5. Определяет адрес сетевого адаптера следующего маршрутизатора.

6. Направляет пакет дальше в сеть.

Этот процесс повторяется на каждом маршрутизаторе до тех пор, пока пакет не дойдет до адресата; там протокол IP собирает из фрагментов пакет в первоначальном виде.

1.3. Стек протоколов TCP/IP

Так как стек TCP/IP был разработан до появления модели взаимодействия открытых систем ISO/OSI, то, хотя он также имеет многоуровневую структуру, соответствие уровней стека TCP/IP уровням модели OSI достаточно условно.

Структура протоколов TCP/IP приведена на рис. 3. Протоколы TCP/IP делятся на 4 уровня.

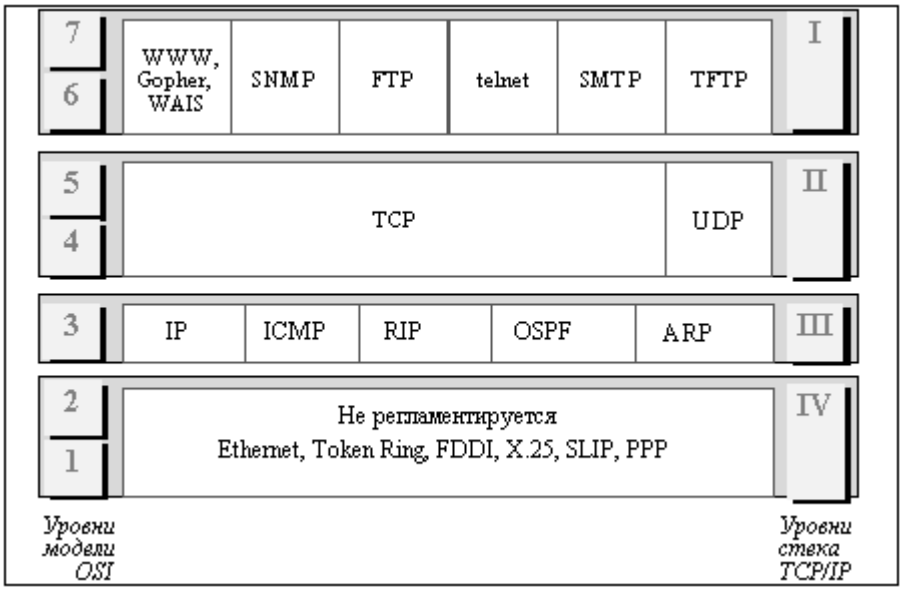


Рис. 3. Стек TCP/IP

Самый нижний (уровень IV) соответствует физическому и канальному уровням модели OSI. Этот уровень в протоколах TCP/IP не регламентируется, но поддерживает все популярные стандарты физического и канального уровня: для локальных сетей это Ethernet, Token Ring, FDDI, Fast Ethernet, 100VG-AnyLAN, для глобальных сетей - протоколы соединений "точка-точка" SLIP и PPP, протоколы территориальных сетей с коммутацией пакетов X.25, frame relay. Разработана также специальная спецификация, определяющая использование технологии ATM в качестве транспорта канального уровня. Обычно при появлении новой технологии локальных или глобальных сетей она быстро включается в стек TCP/IP за счет разработки соответствующего RFC, определяющего метод инкапсуляции пакетов IP в ее кадры.

Следующий уровень (уровень III) - это уровень межсетевого взаимодействия, который занимается передачей пакетов с использованием различных транспортных технологий локальных сетей, территориальных сетей, линий специальной связи и т. п.

В качестве основного протокола сетевого уровня (в терминах модели OSI) в стеке используется протокол IP, который изначально проектировался как протокол передачи пакетов в составных сетях, состоящих из большого количества локальных сетей, объединенных как локальными, так и глобальными связями. Поэтому протокол IP хорошо работает в сетях со сложной топологией, рационально используя наличие в них подсистем и экономно расходуя пропускную способность низкоскоростных линий связи. Протокол IP является дейтаграммным протоколом, то есть он не гарантирует доставку пакетов до узла назначения, но старается это сделать.

К уровню межсетевого взаимодействия относятся и все протоколы, связанные с составлением и модификацией таблиц маршрутизации, такие как протоколы сбора маршрутной информации RIP (Routing Internet Protocol) и OSPF (Open Shortest Path First), а также протокол межсетевых управляющих сообщений ICMP (Internet Control Message Protocol). Последний протокол предназначен для обмена информацией об ошибках между маршрутизаторами сети и узлом - источником пакета. С помощью специальных пакетов ICMP сообщается о невозможности доставки пакета, о превышении времени жизни или продолжительности сборки пакета из фрагментов, об аномальных величинах параметров, об изменении маршрута пересылки и типа обслуживания, о состоянии системы и т.п.

Следующий уровень (уровень II) называется основным. На этом уровне функционируют протокол управления передачей TCP (Transmission Control Protocol) и протокол дейтаграмм пользователя UDP (User Datagram Protocol). Протокол TCP обеспечивает надежную передачу сообщений между удаленными прикладными процессами за счет образования виртуальных соединений. Протокол UDP обеспечивает передачу прикладных пакетов дейтаграммным способом, как и IP, и выполняет только функции связующего звена между сетевым протоколом и многочисленными прикладными процессами.

Верхний уровень (уровень I) называется прикладным. За долгие годы использования в сетях различных стран и организаций стек TCP/IP накопил большое количество протоколов и сервисов прикладного уровня. К ним относятся такие широко используемые протоколы, как протокол копирования файлов FTP, протокол эмуляции терминала telnet, почтовый протокол SMTP, используемый в электронной почте сети Internet, гипертекстовые сервисы доступа к удаленной информации, такие как WWW и многие другие. Остановимся несколько подробнее на некоторых из них.

Протокол пересылки файлов FTP (File Transfer Protocol) реализует удаленный доступ к файлу. Для того, чтобы обеспечить надежную передачу, FTP использует в качестве транспорта протокол с установлением соединений - TCP. Кроме пересылки файлов протокол FTP предлагает и другие услуги. Так, пользователю предоставляется возможность интерактивной работы с удаленной машиной, например, он может распечатать содержимое ее каталогов. Наконец, FTP выполняет аутентификацию пользователей. Прежде, чем получить доступ к файлу, в соответствии с протоколом пользователи должны сообщить свое имя и пароль. Для доступа к публичным каталогам FTP-архивов Internet парольная аутентификация не требуется, и ее обходят за счет использования для такого доступа предопределенного имени пользователя Anonymous.

В стеке TCP/IP протокол FTP предлагает наиболее широкий набор услуг для работы с файлами, однако он является и самым сложным для программирования. Приложения, кото-

рым не требуются все возможности FTP, могут использовать другой, более экономичный протокол - простейший протокол пересылки файлов TFTP (Trivial File Transfer Protocol). Этот протокол реализует только передачу файлов, причем в качестве транспорта используется более простой, чем TCP, протокол без установления соединения - UDP.

Протокол telnet обеспечивает передачу потока байтов между процессами, а также между процессом и терминалом. Наиболее часто этот протокол используется для эмуляции терминала удаленного компьютера. При использовании сервиса telnet пользователь фактически управляет удаленным компьютером так же, как и локальный пользователь, поэтому такой вид доступа требует хорошей защиты. Поэтому серверы telnet всегда используют как минимум аутентификацию по паролю, а иногда и более мощные средства защиты, например, систему Kerberos.

Протокол SNMP (Simple Network Management Protocol) используется для организации сетевого управления. Изначально протокол SNMP был разработан для удаленного контроля и управления маршрутизаторами Internet, которые традиционно часто называют также шлюзами. С ростом популярности протокол SNMP стали применять и для управления любым коммуникационным оборудованием - концентраторами, мостами, сетевыми адаптерами и т.д. и т.п. Проблема управления в протоколе SNMP разделяется на две задачи.

Первая задача связана с передачей информации. Протоколы передачи управляющей информации определяют процедуру взаимодействия SNMP-агента, работающего в управляемом оборудовании, и SNMP-монитора, работающего на компьютере администратора, который часто называют также консолью управления. Протоколы передачи определяют форматы сообщений, которыми обмениваются агенты и монитор.

Вторая задача связана с контролируемыми переменными, характеризующими состояние управляемого устройства. Стандарты регламентируют, какие данные должны сохраняться и накапливаться в устройствах, имена этих данных и синтаксис этих имен. В стандарте SNMP определена спецификация информационной базы данных управления сетью. Эта спецификация, известная как база данных MIB (Management Information Base), определяет те элементы данных, которые управляемое устройство должно сохранять, и допустимые операции над ними.

1.4. Протокол надежной доставки сообщений TCP

В стеке протоколов TCP/IP протокол TCP (Transmission Control Protocol) работает так же, как и протокол UDP, на транспортном уровне. Он обеспечивает надежную транспортировку данных между прикладными процессами путем установления логического соединения.

Сегменты TCP

Единицей данных протокола TCP является сегмент. Информация, поступающая к протоколу TCP в рамках логического соединения от протоколов более высокого уровня, рассматривается протоколом TCP как неструктурированный поток байт. Поступающие данные буферизуются средствами TCP. Для передачи на сетевой уровень из буфера "вырезается" некоторая непрерывная часть данных, называемая сегментом.

В протоколе TCP предусмотрен случай, когда приложение обращается с запросом о срочной передаче данных (бит PSH в запросе установлен в 1). В этом случае протокол TCP, не ожидая заполнения буфера до уровня размера сегмента, немедленно передает указанные данные в сеть. О таких данных говорят, что они передаются вне потока - out of band.

Не все сегменты, посланные через соединение, будут одного и того же размера, однако оба участника соединения должны договориться о максимальном размере сегмента, который они будут использовать. Этот размер выбирается таким образом, чтобы при упаковке сегмента в IP-пакет он помещался туда целиком, то есть максимальный размер сегмента не должен превосходить максимального размера поля данных IP-пакета. В противном случае пришлось бы выполнять фрагментацию, то есть делить сегмент на несколько частей, для того, чтобы он вместился в IP-пакет.

Аналогичные проблемы решаются и на сетевом уровне. Для того, чтобы избежать фрагментации, должен быть выбран соответствующий максимальный размер IP-пакета. Однако при этом должны быть приняты во внимание максимальные размеры поля данных кадров (MTU) всех протоколов канального уровня, используемых в сети. Максимальный размер сегмента не должен превышать минимальное значение на множестве всех MTU составной сети.

Порты и установление TCP-соединений

В протоколе TCP также, как и в UDP, для связи с прикладными процессами используются порты. Номера портам присваиваются аналогичным образом: имеются стандартные, зарезервированные номера (например, номер 21 закреплен за сервисом FTP, 23 - за telnet), а менее известные приложения пользуются произвольно выбранными локальными номерами.

Однако в протоколе TCP порты используются несколько иным способом. Для организации надежной передачи данных предусматривается установление логического соединения между двумя прикладными процессами. В рамках соединения осуществляется обязательное подтверждение правильности приема для всех переданных сообщений, и при необходимости выполняется повторная передача. Соединение в TCP позволяет вести передачу данных одновременно в обе стороны, то есть полнодуплексную передачу.

Соединение в протоколе TCP идентифицируется парой полных адресов обоих взаимодействующих процессов (оконечных точек). Адрес каждой из оконечных точек включает IP-адрес (номер сети и номер компьютера) и номер порта. Одна оконечная точка может участвовать в нескольких соединениях.

Установление соединения выполняется в следующей последовательности:

При установлении соединения одна из сторон является инициатором. Она посылает запрос к протоколу TCP на открытие порта для передачи (active open).

После открытия порта протокол TCP на стороне процесса-инициатора посылает запрос процессу, с которым требуется установить соединение.

Протокол TCP на приемной стороне открывает порт для приема данных (passive open) и возвращает квитанцию, подтверждающую прием запроса.

Для того чтобы передача могла вестись в обе стороны, протокол на приемной стороне также открывает порт для передачи (active port) и также передает запрос к противоположной стороне.

Сторона-инициатор открывает порт для приема и возвращает квитанцию. Соединение считается установленным. Далее происходит обмен данными в рамках данного соединения.

Концепция квитирования

В рамках соединения правильность передачи каждого сегмента должна подтверждаться квитанцией получателя. Квитирование - это один из традиционных методов обеспечения надежной связи. Идея квитирования состоит в следующем.

Для того, чтобы можно было организовать повторную передачу искаженных данных отправитель нумерует отправляемые единицы передаваемых данных (далее для простоты называемые кадрами). Для каждого кадра отправитель ожидает от приемника так называемую положительную квитанцию - служебное сообщение, извещающее о том, что исходный кадр был получен и данные в нем оказались корректными. Время этого ожидания ограничено - при отправке каждого кадра передатчик запускает таймер, и если по его истечению положительная квитанция на получена, то кадр считается утерянным. В некоторых протоколах приемник, в случае получения кадра с искаженными данными должен отправить отрицательную квитанцию - явное указание того, что данный кадр нужно передать повторно.

Существуют два подхода к организации процесса обмена положительными и отрицательными квитанциями: с простоями и с организацией "окна".

Метод с простоями требует, чтобы источник, пославший кадр, ожидал получения квитанции (положительной или отрицательной) от приемника и только после этого посылал следующий кадр (или повторял искаженный). Из рис. 4 видно, что в этом случае производительность обмена данными существенно снижается - хотя передатчик и мог бы послать сле-

дующий кадр сразу же после отправки предыдущего, он обязан ждать прихода квитанции. Снижение производительности для этого метода коррекции особенно заметно на низкоскоростных каналах связи, то есть в территориальных сетях.

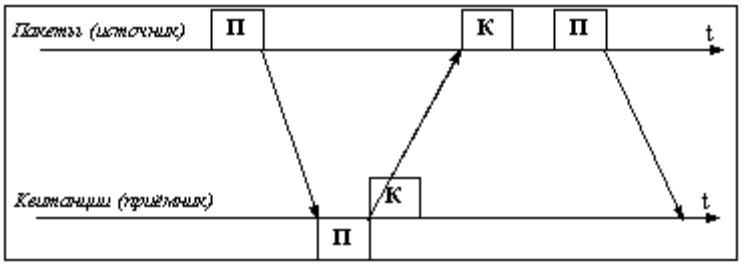


Рис. 4. Метод подтверждения корректности передачи кадров с простоем источника

Во втором методе для повышения коэффициента использования линии источнику разрешается передать некоторое количество кадров в непрерывном режиме, то есть в максимально возможном для источника темпе, без получения на эти кадры ответных квитанций. Количество кадров, которые разрешается передавать таким образом, называется размером окна. Рис. 5 иллюстрирует данный метод для размера окна в W кадров. Обычно кадры при обмене нумеруются циклически, от 1 до W. При отправке кадра с номером 1 источнику разрешается передать еще W-1 кадров до получения квитанции на кадр 1. Если же за это время квитанция на кадр 1 так и не пришла, то процесс передачи приостанавливается, и по истечению некоторого тайм-аута кадр 1 считается утерянным (или квитанция на него утеряна) и он передается снова.

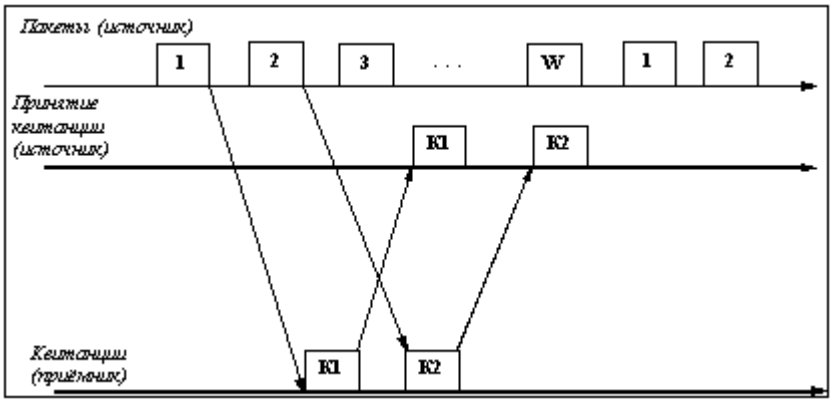


Рис. 5. Метод "окна" - непрерывная отправка пакетов

Если же поток квитанций поступает более-менее регулярно, в пределах допуска в W кадров, то скорость обмена достигает максимально возможной величины для данного канала и принятого протокола.

Этот алгоритм называют алгоритмом скользящего окна. Действительно, при каждом получении квитанции окно перемещается (скользит), захватывая новые данные, которые разрешается передавать без подтверждения.

Реализация скользящего окна в протоколе TCP

В протоколе TCP реализована разновидность алгоритма квитирования с использованием окна. Особенность этого алгоритма состоит в том, что, хотя единицей передаваемых данных является сегмент, окно определено на множестве нумерованных байт неструктурированного потока данных, поступающих с верхнего уровня и буферизуемых протоколом TCP.

Квитанция посылается только в случае правильного приема данных, отрицательные квитанции не посылаются. Таким образом, отсутствие квитанции означает либо прием искаженного сегмента, либо потерю сегмента, либо потерю квитанции.

В качестве квитанции получатель сегмента отсылает ответное сообщение (сегмент), в которое помещает число, на единицу превышающее максимальный номер байта в полученном сегменте. Если размер окна равен W, а последняя квитанция содержала значение N, то отправитель может посылать новые сегменты до тех пор, пока в очередной сегмент не попадет байт с номером N+W. Этот сегмент выходит за рамки окна, и передачу в таком случае необходимо приостановить до прихода следующей квитанции.

Выбор тайм-аута

Выбор времени ожидания (тайм-аута) очередной квитанции является важной задачей, результат решения которой влияет на производительность протокола TCP.

Тайм-аут не должен быть слишком коротким, чтобы по возможности исключить избыточные повторные передачи, которые снижают полезную пропускную способность системы. Но он не должен быть и слишком большим, чтобы избежать длительных простоев, связанных с ожиданием несуществующей или "заблудившейся" квитанции.

При выборе величины тайм-аута должны учитываться скорость и надежность физических линий связи, их протяженность и многие другие подобные факторы. В протоколе TCP тайм-аут определяется с помощью достаточно сложного адаптивного алгоритма, идея которого состоит в следующем. При каждой передаче засекается время от момента отправки сегмента до прихода квитанции о его приеме (время оборота). Получаемые значения времен

оборота усредняются с весовыми коэффициентами, возрастающими от предыдущего замера к последующему. Это делается с тем, чтобы усилить влияние последних замеров. В качестве тайм-аута выбирается среднее время оборота, умноженное на некоторый коэффициент. Практика показывает, что значение этого коэффициента должно превышать 2. В сетях с большим разбросом времени оборота при выборе тайм-аута учитывается и дисперсия этой величины.

Реакция на перегрузку сети

Варьируя величину окна, можно повлиять на загрузку сети. Чем больше окно, тем большую порцию неподтвержденных данных можно послать в сеть. Если сеть не справляется с нагрузкой, то возникают очереди в промежуточных узлах-маршрутизаторах и в конечных узлах-компьютерах.

При переполнении приемного буфера конечного узла "перегруженный" протокол TCP, отправляя квитанцию, помещает в нее новый, уменьшенный размер окна. Если он совсем отказывается от приема, то в квитанции указывается окно нулевого размера. Однако даже после этого приложение может послать сообщение на отказавшийся от приема порт. Для этого, сообщение должно сопровождаться пометкой "срочно" (бит URG в запросе установлен в 1). В такой ситуации порт обязан принять сегмент, даже если для этого придется вытеснить из буфера уже находящиеся там данные.

После приема квитанции с нулевым значением окна протокол-отправитель время от времени делает контрольные попытки продолжить обмен данными. Если протокол-приемник уже готов принимать информацию, то в ответ на контрольный запрос он посылает квитанцию с указанием ненулевого размера окна.

Другим проявлением перегрузки сети является переполнение буферов в маршрутизаторах. В таких случаях они могут централизовано изменить размер окна, посылая управляющие сообщения некоторым конечным узлам, что позволяет им дифференцировано управлять интенсивностью потока данных в разных частях сети.

Формат сообщений TCP

Сообщения протокола TCP называются сегментами и состоят из заголовка и блока данных. Заголовок сегмента имеет следующие поля:

Порт источника (SOURS PORT) занимает 2 байта, идентифицирует процесс-отправитель;

Порт назначения (DESTINATION PORT) занимает 2 байта, идентифицирует процесс-получатель;

Последовательный номер (SEQUENCE NUMBER) занимает 4 байта, указывает номер байта, который определяет смещение сегмента относительно потока отправляемых данных;

Подтвержденный номер (ACKNOWLEDGEMENT NUMBER) занимает 4 байта, содержит максимальный номер байта в полученном сегменте, увеличенный на единицу; именно это значение используется в качестве квитанции;

Длина заголовка (HLEN) занимает 4 бита, указывает длину заголовка сегмента TCP, измеренную в 32-битовых словах. Длина заголовка не фиксирована и может изменяться в зависимости от значений, устанавливаемых в поле Опции;

Резерв (RESERVED) занимает 6 битов, поле зарезервировано для последующего использования;

Кодовые биты (CODE BITS) занимают 6 битов, содержат служебную информацию о типе данного сегмента, задаваемую установкой в единицу соответствующих бит этого поля:

URG - срочное сообщение;

ACK - квитанция на принятый сегмент;

PSH - запрос на отправку сообщения без ожидания заполнения буфера;

RST - запрос на восстановление соединения;

SYN - сообщение используемое для синхронизации счетчиков переданных данных при установлении соединения;

FIN - признак достижения передающей стороной последнего байта в потоке передаваемых данных.

Окно (WINDOW) занимает 2 байта, содержит объявляемое значение размера окна в байтах;

Контрольная сумма (CHECKSUM) занимает 2 байта, рассчитывается по сегменту;

Указатель срочности (URGENT POINTER) занимает 2 байта, используется совместно с кодовым битом URG, указывает на конец данных, которые необходимо срочно принять, несмотря на переполнение буфера;

Опции (OPTIONS) - это поле имеет переменную длину и может вообще отсутствовать, максимальная величина поля 3 байта; используется для решения вспомогательных задач, например, при выборе максимального размера сегмента;

Заполнитель (PADDING) может иметь переменную длину, представляет собой фиктивное поле, используемое для доведения размера заголовка до целого числа 32-битовых слов.

2. Лабораторное задание

Лабораторное задание состоит из индивидуальных заданий. Получите у преподавателя вариант индивидуального задания.

Выполнение индивидуального задания заключается в освоении протокола TCP/IP и установлении связи с его помощью.

Задание 1. Исследовать протокол TCP и определить его составные части и места их расположения в сетевой операционной системе.

Задание 2. Установить связь с другим компьютером по протоколу TCP и переслать сообщение.

Задание 3. Исследовать протокол IP и определить его составные части и места их расположения в сетевой операционной системе.

Задание 4. Исследовать способы маршрутизации при помощи протокола TCP/IP.

Задание 5. Исследовать методы работы маршрутизатора при помощи протокола TCP/IP.

Задание 6. Исследовать реализацию протокола IP на маршрутизаторе.

3. Содержание отчета

Отчет должен содержать:

- Цель работы.
- Теоретические сведения.
- Описание выполнения работы.
- Выводы по работе.

4. Контрольные вопросы

1. Назначение протокола TCP/IP.
2. Как осуществляется установка связи по протоколу TCP.
3. Назовите особенность протокола IP.
4. Дайте определение маршрутизации.
5. Дайте определение маршрутизатора.
6. Назовите функции маршрутизатора.

5. Литература

1. Ефимова О.В. и др. Практикум по компьютерной технологии.- М.: АБФ, 2002.- 260с.

2. Кутугина, Е. С., Тутубалин, Д. К. Информационные технологии: Учеб. пособие. — Томск, 2005.

Лабораторная работа №2.
IP-АДРЕСАЦИЯ

Цель работы: ознакомление с IP-компонентами.

1. Теоретические сведения

1.1. Введение в IP- адресацию

В стеке TCP/IP использованы три типа адресов: локальные (аппаратные), IP-адреса и символьные доменные адреса.

Под *локальным адресом* понимается такой тип адреса, который используется средствами базовой технологии для доставки данных в пределах подсети, являющейся элементом составной интерсети. В разных подсетях допустимы разные сетевые технологии, разные стеки протоколов, поэтому при создании стека TCP/IP предполагалось наличие разных типов локальных адресов. Если подсетью интерсети является локальная сеть, то локальный адрес - это MAC-адрес. MAC-адрес назначается сетевым адаптерам и сетевым интерфейсам маршрутизаторов. MAC-адрес назначается производителями оборудования и является уникальным, т.к. управляется централизовано. Для всех существующих технологий локальных сетей MAC-адрес имеет формат 6 байт, например 11-A0-17-3D-BC-01. Однако протокол IP может работать и над протоколами более высокого уровня, например, IPX или X.25.

IP-адреса представляют собой основной тип адресов, на основании которых сетевой уровень передаёт пакеты между сетями. Эти адреса состоят из 4 байт, например 109.26.17.100. IP-адрес назначается администратором во время конфигурирования сети. Он состоит из двух частей: номера сети и номера узла. Номер сети может быть выбран администратором произвольно, либо назначен по рекомендации специального информационного центра Intenet, если сеть должна работать как составная часть Internet. Номер узла в протоколе IP назначается независимо от локального адреса узла. Маршрутизатор по определению входит сразу в несколько сетей, поэтому каждый его порт имеет собственный IP-адрес. Конечный узел также может входить в несколько IP-сетей, тогда компьютер должен иметь несколько IP-адресов, по числу сетевых связей. Таким образом, IP-адрес характеризует не отдельный компьютер или маршрутизатор, а одно сетевое соединение.

Символьные доменные адреса строятся по иерархическому признаку. Составляющие полного символьного имени разделяются точкой и перечисляются в следующем порядке: простое имя конечного узла, имя группы узлов (например, для большой организации), имя более крупной группы (поддомена) и так до имени домена самого высокого уровня (обычно географического: ru - Россия, md - Молдова, ua - Украина). Пример имени - etf.pgu.tirastel.md. Между доменными именем и IP-адресом узла нет никакого алгоритмического соответствия, поэтому необходимо использовать дополнительные таблицы или службы, чтобы узел сети однозначно определялся как по доменному имени, так и по IP-адресу. В сетях TCP/IP используется специальная распределённая служба Domain Name System (DNS), которая устанавливает это соответствие на основании создаваемых администраторами сети таблиц соответствия.

IP-адрес имеет длину 4 байта и обычно записывается в виде четырёх чисел, разделённых точками, например, 128.10.2.30 - это традиционная десятичная форма представления адреса, в 10000000 00001010 00000010 00011110 - двоичная форма этого же адреса.

Адрес состоит из двух логических частей - номера сети и номера узла в сети. Какая часть адреса относится к номеру, а какая - к номеру узла, определяется значениями первых бит адреса. Значения этих бит являются также признаками класса IP-адреса.

Если адрес начинается с 0, то сеть относится к классу A и номер сети занимает один байт, остальные 3 байта интерпретируются как номер узла в сети. Эти сети имеют номера в диапазоне от 1 до 126. Количество узлов в такой сети может быть 2^{24} - 16 777 216 узлов.

Если адрес начинается с 10, то он относится к классу B. В этих сетях под номер сети и под номер узла отводится по 16 бит. Сети класса B являются сетями средних размеров с количеством узлов 2^{16}- 65 536.

Если адрес начинается с последовательности 110, то это сеть класса C, в которой под номер сети отводится 24 бита, а под номер узла - 8 бит. Сети этого класса наиболее распространены, число узлов в них - 2^8 - 256.

Если адрес начинается с последовательности 1110, то он является адресом класса D и обозначает особый групповой адрес - multicast. Если в пакете в качестве адреса назначения указан адрес класса D, то такой пакет должны получить все узлы, которым присвоен данный адрес.

Если адрес начинается с последовательности 11110, то этот адрес принадлежит к классу E. Адреса этого класса зарезервированы для будущих применений.

В протоколе IP существует несколько соглашений об особой интерпретации IP-адресов.

- Если весь IP-адрес состоит только из двоичных нулей, то он обозначает адрес того узла, который сгенерировал этот пакет, этот режим используется только в некоторых сообщениях ICMP.

- Если в поле номера сети стоят только нули, то по умолчанию читается, что узел назначения принадлежит той же самой сети, что и узел, который отправил пакет.

- Если все двоичные разряды IP-адреса равны 1, то пакет с таким адресом назначения должен рассылаться всем узлам, находящимся в той же сети, что и источник этого пакета. Такая посылка называется ограниченным широковещательным сообщением.

- Если в поле номера узла назначения стоят только единицы, то пакет рассылается всем узлам сети с заданным номером. Например, пакет с адресом 192.190.21.255 доставляется всем узлам сети 192.190.21.0. Такая рассылка называется широковещательным сообщением.

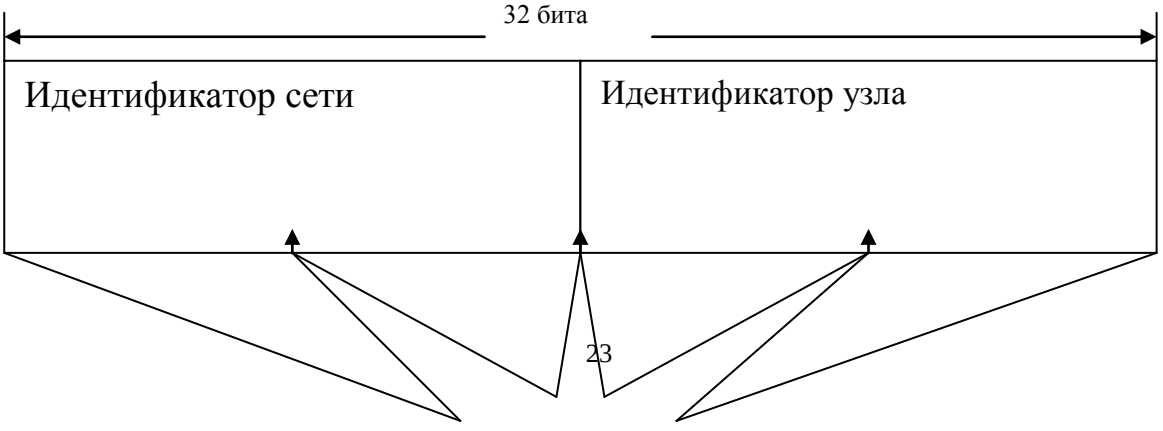
Особый смысл имеет IP-адрес, первый октет которого равен 127. Он используется для тестирования программ и взаимодействия процессов в пределах одной машины. Когда программа посылает данные по IP-адресу 127.0.0.1, то образуется как бы "петля". Данные не передаются по сети, а возвращаются модулям верхнего уровня как только что прочитанные.

Идентификаторы сетей и узлов.

IP-адрес может быть записан в двух форматах – двоичном (binary) и десятичном с точками (dotted decimal). Каждый IP-адрес имеет длину 32 бета и состоит из четырех 8-битных полей, называемых октетами (octets), которые отделяются друг от друга точками. Каждый октет представляет десятичное число в диапазоне от 0 до 255. Эти 32 разряда IP-адреса содержат идентификатор сети и узла.

Формат записи адреса в виде четырех десятичных чисел, разделенных течками, наиболее удобен для восприятия. Далее показаны различные формы записи IP- адреса.

Двоичный формат	**Десятичный формат с точками**
10000011 01101011 00000011 00011000	131.107.3.24



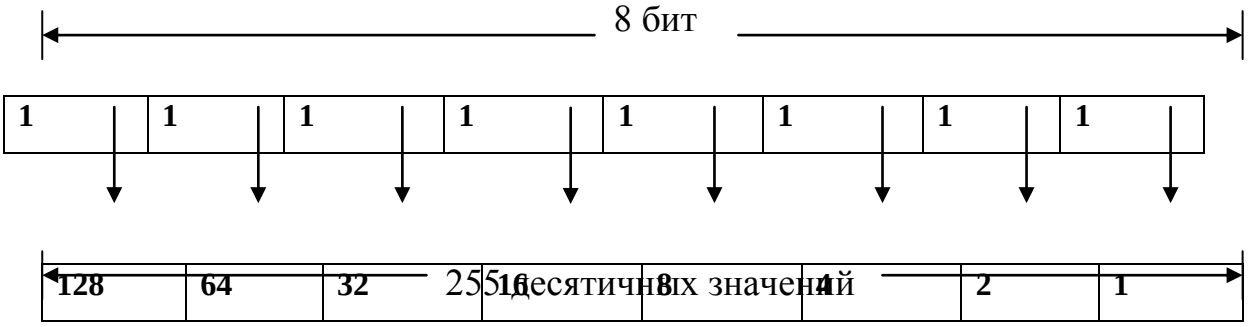
W. X. Y. Z

Пример: **131.107.3.24**

Преобразование IP-адреса из двоичного формата в десятичный.

В двоичном формате каждому биту в октете сопоставлено определенное десятичное число. Максимальное десятичное значение октета равно 255(участвует каждый бит). Каждый октет преобразуется в число отдельно от других.

Бит, установленный в 0, всегда соответствует нулевому значению. Бит, установленный в 1, может быть преобразован в десятичное число. Младший бит октета представляет десятичное число 1, а старший – 128. Максимальное значение октета (255) достигается, когда каждый его бит равен 1.



В следующей таблице показано, как биты одного октета преобразуются в десятичное число.

Двоичная запись	**Значение бит**	**Десятичное число**
00000000	0	0
00000001	1	1
00000011	1+2	3
00000111	1+2+4	7
00001111	1+2+4+8	15
00011111	1+2+4+8+16	31
00111111	1+2+4+8+16+32	63
01111111	1+2+4+8+16+32+64	127
11111111	1+2+4+8+16+32+64 +128	255

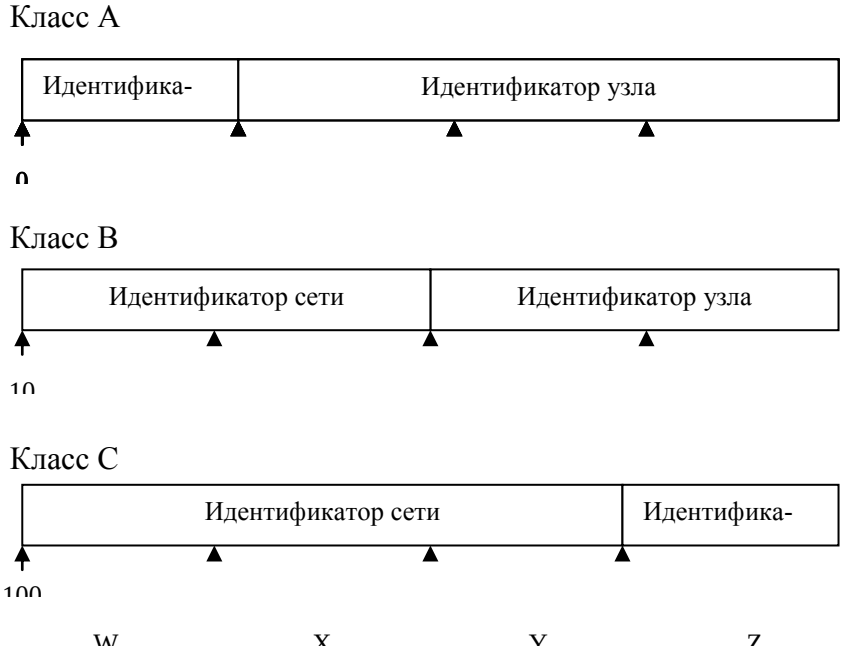
Классы IP-адресов.

Каждый класс IP-адресов определяет, какая часть адреса отводится под идентификатор сети, а какая – под идентификатор узла.

Протокол TCP/IP поддерживает адреса классов A, B и C. Класс адреса определяет, какие биты относятся к идентификатору сети, а какие – к идентификатору узла. Также он определяет максимально возможное количество узлов в сети.

Класс IP-адреса идентифицируют по значению его первого октета, 32-разрядные IP-адреса могут быть присвоены в общей совокупности 3720314628 узлам. Ниже показано, как определяются поля в IP-адресах разных классов.

Класс	**IP-адрес**	**Идентификатор сети**	**Идентификатор узла**
A	w.x.y.z	w	x.y.z
B	w.x.y.z	w.x	y.z
C	w.x.y.z	w.x.y	z



Класс A.

Адреса класса A назначаются узлам очень большой сети. Старший бит в адресах этого класса всегда равен нулю. Следующие семь бит первого октета представляют идентификатор сети. Оставшиеся 24 бита (три октета) содержат идентификатор узла. Это позволяет иметь 126 сетей с числом узлов до 17 миллионов в каждой.

Класс B.

Адреса класса B назначаются узлам в больших и средних по размеру в сетях. В двух старших битах IP-адреса класса B записывается двоичное значение 10. Следующие 14 бит содержат идентификатор сети (два первых октета). Оставшиеся 16 бит (два октета) представ-

ляют идентификатор узла. Таким образом возможно существование 16384 сетей класса B, в акждой из которых около 65000 узлов.

Класс C.

Ареса класса C применяются в небольших сетях. Три старших бита IP-адреса этого класса содержат двоичное значение 110. Следующие 21 бит составляет идентификатор сети (первые три октета). Оставшиеся 8 бит (последний октет) отводится под идентификатор узла. Всего возможно около 2000000 сетей класса C, содержащих до 254 узлов.

	Количество сетей	Количество узлов в сети	Диапазон значений идентификаторов сети
Класс A	126	16777214	1-126
Класс B	16384	65534	128-191
Класс C	2097152	254	192-223

IP-адреса и маска подсетей.

Маска подсети – это 32-разрядное значение, используемое для выделения из IP-адреса его частей: идентификаторов сети и узла. Такая процедура необходима при выяснении того, относится тот или иной IP-адрес к локальной или удаленной сети.

Каждый узел TCP/IP должен иметь маску подсети – либо задаваемую по умолчанию (в том случае, когда сеть не делится на подсети), либо специальную (если сеть разбита на подсети).

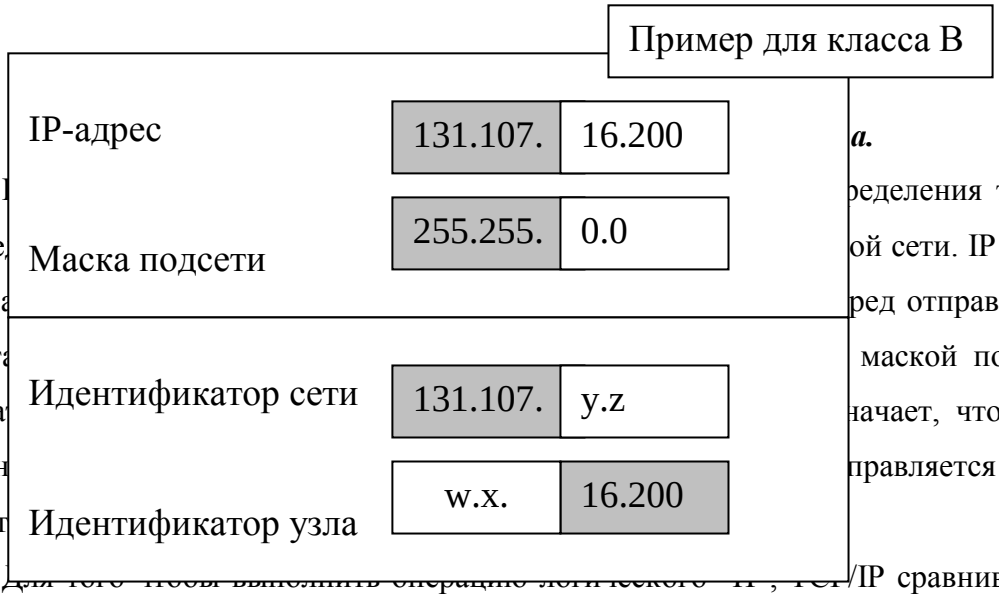
Маска подсети, задаваемая по умолчанию.

Задаваемая по умолчанию маска подсети используется в том случае, если сеть TCP/IP не разделяется на подсети. Даже в сети, состоящей из одного сегмента, всем узлам TCP/IP необходима маска подсети. Значение маски подсети по умолчанию зависит от используемого в данной сети класса IP-адресов.

В маске подсети биты, соответствующие идентификатору сети, устанавливаются в 1. Таким образом, значение каждого октета будет равно 255. Все биты, соответствующие идентификатору узла, устанавливаются в 0.

Класс адресов	**Биты, используемые для маски подсети**	**Десятичная запись с точками**

Класс A	11111111 00000000 00000000 00000000	255.0.0.0
Класс B	11111111 11111111 00000000 00000000	255.255.0.0
Класс C	11111111 11111111 11111111 00000000	255.255.255.0



а.

определения того, какому узлу пре… …ой сети. IP – адрес узла склады… …ред отправкой каждого IP-пакета … маской подсети. Если результат … …начает, что получатель пакета н… …правляется на IP-адрес маршрут…

Для того чтобы выполнить операцию логического "И", TCP/IP сравнивает попарно соответствующие биты адреса и маски. Если оба бита равны 1, результат равен 1. В остальных случаях результирующий бит равен 0.

Сопоставление бит	**Результат**
1 “И” 1	1
1 “И” 0	0
0 “И” 0	0
0 “И” 1	0

IP-адрес Маска подсети	10011111 11100000 00000111 10000001 11111111 11111111 00000000 00000000
Результат	10011111 11100000 00000000 00000000

1.2. Организация доменов и доменных имен

Для идентификации компьютеров аппаратное и программное обеспечение в сетях TCP/IP полагается на IP-адреса, поэтому для доступа к сетевому ресурсу в параметрах программы вполне достаточно указать IP-адрес. Например, команда ftp://192.45.66.17 будет устанавливать сеанс связи с нужным ftp-сервером, а команда http://203.23.106.33 откроет начальную страницу на корпоративном Web-сервере. Однако пользователю гораздо удобнее работать с символьными именами и для этого в сетях TCP/IP символьные имена хостов и механизм для установления соответствия между символьными именами и IP-адресами.

В операционных системах, которые первоначально разрабатывались для работы в локальных сетях, таких как Novell, Windows и OS/2 пользователи всегда работали с символьными именами компьютеров. Для стека TCP/IP, расчитанного в общем случае на работу в больших территориально распределённых сетях, такой подход оказался неэффективным. Плоские имена не дают возможности разработать единый алгоритм обеспечения уникальности имен в пределах большой сети. Широковещательный способ установления соответствия между символьными именами и локальными адресами хорошо работает только в небольшой локальной сети, не разделённой на подсети. Для эффективной организации именования компьютеров в больших сетях естественным является применение иерархических составных имен.

В стеке TCP/IP применяется доменная система имен, которая имеет иерархическую древовидную структуру, допускающую использование в имени произвольного количества составных частей. Иерархия доменных имен аналогична файловой иерархии. Дерево имен начинается с корня, обозначаемого (.). Затем следует старшая символьная часть имени, вторая по старшинству символьная часть и т.д. Младшая часть имени соответствует конечному узлу сети. Запись доменного имени начинается с самой младшей составляющей, а заканчивается самой старшей. Составные части доменного имени отделяют друг от друга точкой.

Разделение имени на части позволяет разделить административную ответственность за назначение уникальных имен в пределах своего уровня иерархии. Разделение административной ответственности позволяет решить проблему образования уникальных имен без взаимных консультаций. Совокупность имен, у которых несколько старших составных частей совпадают, образуют домен имен. Например, имена www1.zil.mmt.ru, ftp.zil.mmt.ru, yandex.ru и s1.mgu.ru входят в домен ru, так как все эти имена имеют одну общую старшую часть - имя ru.

Если один домен входит в другой домен как его составная часть, то такой домен могут называть поддоменом, хотя название домен за ним также остается. Обычно подддомен

называют по имени той его старшей составляющей, которая отличает его от других поддоменов. Если в каждом домене и поддомене обеспечивается уникальность имен следующего уровня иерархии, то и вся система имен будет состоять из уникальных имен.

По аналогии с файловой системой, в доменной системе имен различают краткие имена, относительные имена и полные доменные имена. Краткое имя - это имя конечного узла сети: хоста или порта маршрутизатора. Относительное имя - это составное имя, начинающееся с некоторого уровня иерархии, но не самого верхнего. Полное доменное имя включает составляющие всех уровней иерархии, начиная от кратного имени и кончая корневой точкой.

В Internet корневой домен управляется центром InterNIC. Домены верхнего уровня назначаются для каждой страны, а также на организационной основе. Имена этих доменов должны следовать международному стандарту ISO 3166. Для обозначения стран используются двух- и трёхбуквенные аббревиатуры, а для различных типов организаций - следующие обозначения:

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

Каждый домен администрируется отдельной организацией, которая обычно разбивает свой домен на поддомены и делегирует функции администрирования ниже по дереву. В России такой организацией является РосНИИРОС, которая отвечает за делегирование имен поддоменов в домене ru.

Адресация в IP-сетях

Типы адресов: физический (MAC-адрес), сетевой (IP-адрес) и символьный (DNS-имя)

Каждый компьютер в сети TCP/IP имеет адреса трех уровней:

- Локальный адрес узла, определяемый технологией, с помощью которой построена отдельная сеть, в которую входит данный узел. Для узлов, входящих в локальные сети - это MAC-адрес сетевого адаптера или порта маршрутизатора, например, 11-A0-17-3D-BC-01. Эти адреса назначаются производителями оборудования и являются уникальными адресами, так как управляются централизовано. Для всех существующих технологий локальных сетей MAC-адрес имеет формат 6 байтов: старшие 3 байта - идентификатор фирмы про-

изводителя, а младшие 3 байта назначаются уникальным образом самим производителем. Для узлов, входящих в глобальные сети, такие как X.25 или frame relay, локальный адрес назначается администратором глобальной сети.

- IP-адрес, состоящий из 4 байт, например, 109.26.17.100. Этот адрес используется на сетевом уровне. Он назначается администратором во время конфигурирования компьютеров и маршрутизаторов. IP-адрес состоит из двух частей: номера сети и номера узла. Номер сети может быть выбран администратором произвольно, либо назначен по рекомендации специального подразделения Internet (Network Information Center, NIC), если сеть должна работать как составная часть Internet. Обычно провайдеры услуг Internet получают диапазоны адресов у подразделений NIC, а затем распределяют их между своими абонентами.

Номер узла в протоколе IP назначается независимо от локального адреса узла. Деление IP-адреса на поле номера сети и номера узла - гибкое, и граница между этими полями может устанавливаться весьма произвольно. Узел может входить в несколько IP-сетей. В этом случае узел должен иметь несколько IP-адресов, по числу сетевых связей. Таким образом IP-адрес характеризует не отдельный компьютер или маршрутизатор, а одно сетевое соединение.

- Символьный идентификатор-имя, например, SERV1.IBM.COM. Этот адрес назначается администратором и состоит из нескольких частей, например, имени машины, имени организации, имени домена. Такой адрес, называемый также DNS-именем, используется на прикладном уровне, например, в протоколах FTP или telnet.

Три основных класса IP-адресов

IP-адрес имеет длину 4 байта и обычно записывается в виде четырех чисел, представляющих значения каждого байта в десятичной форме, и разделенных точками, например:

128.10.2.30 - традиционная десятичная форма представления адреса,

10000000 00001010 00000010 00011110 - двоичная форма представления этого же адреса.

На рисунке 1 показана структура IP-адреса.

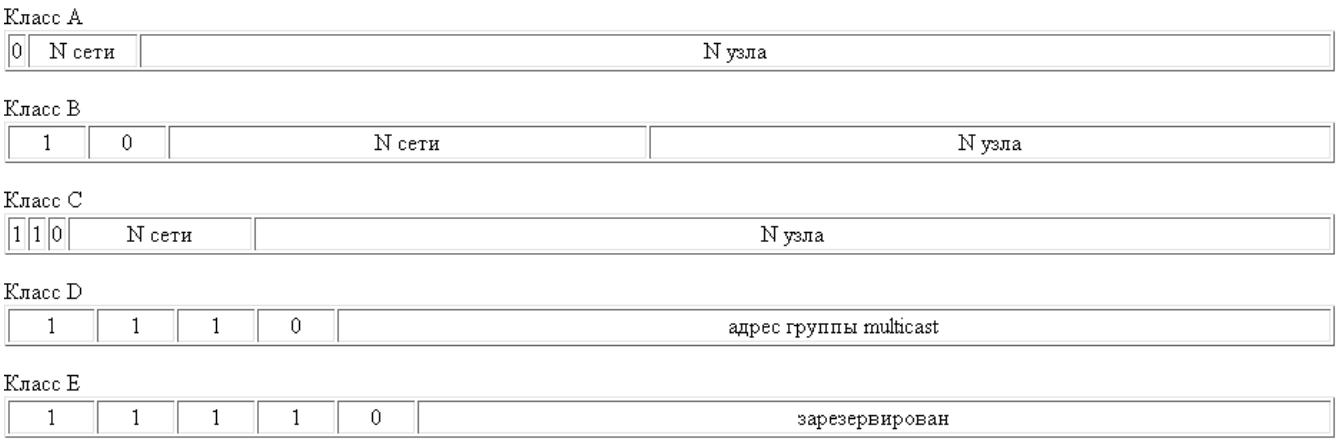


Рис. 1. Структура IP-адреса

Адрес состоит из двух логических частей - номера сети и номера узла в сети. Какая часть адреса относится к номеру сети, а какая к номеру узла, определяется значениями первых битов адреса:

Если адрес начинается с 0, то сеть относят к классу A, и номер сети занимает один байт, остальные 3 байта интерпретируются как номер узла в сети. Сети класса A имеют номера в диапазоне от 1 до 126. (Номер 0 не используется, а номер 127 зарезервирован для специальных целей, о чем будет сказано ниже.) В сетях класса A количество узлов должно быть больше 216 , но не превышать 224.

Если первые два бита адреса равны 10, то сеть относится к классу B и является сетью средних размеров с числом узлов 28 - 216. В сетях класса B под адрес сети и под адрес узла отводится по 16 битов, то есть по 2 байта.

Если адрес начинается с последовательности 110, то это сеть класса C с числом узлов не больше 28. Под адрес сети отводится 24 бита, а под адрес узла - 8 битов.

Если адрес начинается с последовательности 1110, то он является адресом класса D и обозначает особый, групповой адрес - multicast. Если в пакете в качестве адреса назначения указан адрес класса D, то такой пакет должны получить все узлы, которым присвоен данный адрес.

Если адрес начинается с последовательности 11110, то это адрес класса E, он зарезервирован для будущих применений.

В таблице приведены диапазоны номеров сетей, соответствующих каждому классу сетей. Класс Наименьший адрес Наибольший адрес

Класс	Наименьший адрес	Наибольший адрес
A	01.0.0	126.0.0.0
B	128.0.0.0	191.255.0.0
C	192.0.1.0.	223.255.255.0
D	224.0.0.0	239.255.255.255
E	240.0.0.0	247.255.255.255

Соглашения о специальных адресах: broadcast, multicast, loopback

В протоколе IP существует несколько соглашений об особой интерпретации IP-адресов:

- если IP-адрес состоит только из двоичных нулей,

0 0 0 0 ..... 0 0 0 0

то он обозначает адрес того узла, который сгенерировал этот пакет;

- если в поле номера сети стоят 0,

0 0 0 0 0	Номер узла

то по умолчанию считается, что этот узел принадлежит той же самой сети, что и узел, который отправил пакет;

- если все двоичные разряды IP-адреса равны 1,

1 1 1 1 .....1 1

то пакет с таким адресом назначения должен рассылаться всем узлам, находящимся в той же сети, что и источник этого пакета. Такая рассылка называется ограниченным широковещательным сообщением (limited broadcast);

- если в поле адреса назначения стоят сплошные 1,

Номер сети	1111.....11

то пакет, имеющий такой адрес рассылается всем узлам сети с заданным номером. Такая рассылка называется широковещательным сообщением (broadcast);

- адрес 127.0.0.1 зарезервирован для организации обратной связи при тестировании работы программного обеспечения узла без реальной отправки пакета по сети. Этот адрес имеет название loopback.

Уже упоминавшаяся форма группового IP-адреса - multicast - означает, что данный пакет должен быть доставлен сразу нескольким узлам, которые образуют группу с номером, указанным в поле адреса. Узлы сами идентифицируют себя, то есть определяют, к какой из групп они относятся. Один и тот же узел может входить в несколько групп. Такие сообщения в отличие от широковещательных называются мультивещательными. Групповой адрес не делится на поля номера сети и узла и обрабатывается маршрутизатором особым образом.

В протоколе IP нет понятия широковещательности в том смысле, в котором оно используется в протоколах канального уровня локальных сетей, когда данные должны быть доставлены абсолютно всем узлам. Как ограниченный широковещательный IP-адрес, так и широковещательный IP-адрес имеют пределы распространения в интерсети - они ограничены либо сетью, к которой принадлежит узел - источник пакета, либо сетью, номер которой указан в адресе назначения. Поэтому деление сети с помощью маршрутизаторов на части локализует широковещательный шторм пределами одной из составляющих общую сеть частей

просто потому, что нет способа адресовать пакет одновременно всем узлам всех сетей составной сети.

Отображение физических адресов на IP-адреса: протоколы ARP и RARP

В протоколе IP-адрес узла, то есть адрес компьютера или порта маршрутизатора, назначается произвольно администратором сети и прямо не связан с его локальным адресом, как это сделано, например, в протоколе IPX. Подход, используемый в IP, удобно использовать в крупных сетях и по причине его независимости от формата локального адреса, и по причине стабильности, так как в противном случае, при смене на компьютере сетевого адаптера это изменение должны бы были учитывать все адресаты всемирной сети Internet (в том случае, конечно, если сеть подключена к Internet'у).

Локальный адрес используется в протоколе IP только в пределах локальной сети при обмене данными между маршрутизатором и узлом этой сети. Маршрутизатор, получив пакет для узла одной из сетей, непосредственно подключенных к его портам, должен для передачи пакета сформировать кадр в соответствии с требованиями принятой в этой сети технологии и указать в нем локальный адрес узла, например его MAC-адрес. В пришедшем пакете этот адрес не указан, поэтому перед маршрутизатором встает задача поиска его по известному IP-адресу, который указан в пакете в качестве адреса назначения. С аналогичной задачей сталкивается и конечный узел, когда он хочет отправить пакет в удаленную сеть через маршрутизатор, подключенный к той же локальной сети, что и данный узел.

Для определения локального адреса по IP-адресу используется протокол разрешения адреса Address Resolution Protocol, ARP. Протокол ARP работает различным образом в зависимости от того, какой протокол канального уровня работает в данной сети - протокол локальной сети (Ethernet, Token Ring, FDDI) с возможностью широковещательного доступа одновременно ко всем узлам сети, или же протокол глобальной сети (X.25, frame relay), как правило не поддерживающий широковещательный доступ. Существует также протокол, решающий обратную задачу - нахождение IP-адреса по известному локальному адресу. Он называется реверсивный ARP - RARP (Reverse Address Resolution Protocol) и используется при старте бездисковых станций, не знающих в начальный момент своего IP-адреса, но знающих адрес своего сетевого адаптера.

В локальных сетях протокол ARP использует широковещательные кадры протокола канального уровня для поиска в сети узла с заданным IP-адресом.

Узел, которому нужно выполнить отображение IP-адреса на локальный адрес, формирует ARP запрос, вкладывает его в кадр протокола канального уровня, указывая в нем из-

вестный IP-адрес, и рассылает запрос широковещательно. Все узлы локальной сети получают ARP запрос и сравнивают указанный там IP-адрес с собственным. В случае их совпадения узел формирует ARP-ответ, в котором указывает свой IP-адрес и свой локальный адрес и отправляет его уже направленно, так как в ARP запросе отправитель указывает свой локальный адрес. ARP-запросы и ответы используют один и тот же формат пакета. Так как локальные адреса могут в различных типах сетей иметь различную длину, то формат пакета протокола ARP зависит от типа сети. На рис. 2 показан формат пакета протокола ARP для передачи по сети Ethernet.

0 8 16 31

Тип сети		**Тип протокола**
Длина локального адреса	Длина сетевого адреса	Операция
Локальный адрес отправителя (байты 0 - 3)		
Локальный адрес отправителя (байты 4 - 5)		IP-адрес отправителя (байты 0-1)
IP-адрес отправителя (байты 2-3)		Искомый локальный адрес (байты 0 - 1)
Искомый локальный адрес (байты 2-5)		
Искомый IP-адрес (байты 0 - 3)		

Рис. 2. Формат пакета протокола ARP

В поле типа сети для сетей Ethernet указывается значение 1. Поле типа протокола позволяет использовать пакеты ARP не только для протокола IP, но и для других сетевых протоколов. Для IP значение этого поля равно 080016.

Длина локального адреса для протокола Ethernet равна 6 байтам, а длина IP-адреса - 4 байтам. В поле операции для ARP запросов указывается значение 1 для протокола ARP и 2 для протокола RARP.

Узел, отправляющий ARP-запрос, заполняет в пакете все поля, кроме поля искомого локального адреса (для RARP-запроса не указывается искомый IP-адрес). Значение этого поля заполняется узлом, опознавшим свой IP-адрес.

В глобальных сетях администратору сети чаще всего приходится вручную формировать ARP-таблицы, в которых он задает, например, соответствие IP-адреса адресу узла сети X.25, который имеет смысл локального адреса. В последнее время наметилась тенденция автоматизации работы протокола ARP и в глобальных сетях. Для этой цели среди всех маршрутизаторов, подключенных к какой-либо глобальной сети, выделяется специальный маршрутизатор, который ведет ARP-таблицу для всех остальных узлов и маршрутизаторов этой сети. При таком централизованном подходе для всех узлов и маршрутизаторов вручную нужно задать только IP-адрес и локальный адрес выделенного маршрутизатора. Затем каждый узел и маршрутизатор регистрирует свои адреса в выделенном маршрутизаторе, а при необходимости установления соответствия между IP-адресом и локальным адресом узел об-

ращается к выделенному маршрутизатору с запросом и автоматически получает ответ без участия администратора.

Отображение символьных адресов на IP-адреса: служба DNS

DNS (Domain Name System) - это распределенная база данных, поддерживающая иерархическую систему имен для идентификации узлов в сети Internet. Служба DNS предназначена для автоматического поиска IP-адреса по известному символьному имени узла. Спецификация DNS определяется стандартами RFC 1034 и 1035. DNS требует статической конфигурации своих таблиц, отображающих имена компьютеров в IP-адрес.

Протокол DNS является служебным протоколом прикладного уровня. Этот протокол несимметричен - в нем определены DNS-серверы и DNS-клиенты. DNS-серверы хранят часть распределенной базы данных о соответствии символьных имен и IP-адресов. Эта база данных распределена по административным доменам сети Internet. Клиенты сервера DNS знают IP-адрес сервера DNS своего административного домена и по протоколу IP передают запрос, в котором сообщают известное символьное имя и просят вернуть соответствующий ему IP-адрес.

Если данные о запрошенном соответствии хранятся в базе данного DNS-сервера, то он сразу посылает ответ клиенту, если же нет - то он посылает запрос DNS-серверу другого домена, который может сам обработать запрос, либо передать его другому DNS-серверу. Все DNS-серверы соединены иерархически, в соответствии с иерархией доменов сети Internet. Клиент опрашивает эти серверы имен, пока не найдет нужные отображения. Этот процесс ускоряется из-за того, что серверы имен постоянно кэшируют информацию, предоставляемую по запросам. Клиентские компьютеры могут использовать в своей работе IP-адреса нескольких DNS-серверов, для повышения надежности своей работы.

База данных DNS имеет структуру дерева, называемого доменным пространством имен, в котором каждый домен (узел дерева) имеет имя и может содержать поддомены. Имя домена идентифицирует его положение в этой базе данных по отношению к родительскому домену, причем точки в имени отделяют части, соответствующие узлам домена.

Корень базы данных DNS управляется центром Internet Network Information Center. Домены верхнего уровня назначаются для каждой страны, а также на организационной основе. Имена этих доменов должны следовать международному стандарту ISO 3166. Для обозначения стран используются трехбуквенные и двухбуквенные аббревиатуры, а для различных типов организаций используются следующие аббревиатуры:

com - коммерческие организации (например, microsoft.com);

edu - образовательные (например, mit.edu);

gov - правительственные организации (например, nsf.gov);

org - некоммерческие организации (например, fidonet.org);

net - организации, поддерживающие сети (например, nsf.net).

Каждый домен DNS администрируется отдельной организацией, которая обычно разбивает свой домен на поддомены и передает функции администрирования этих поддоменов другим организациям. Каждый домен имеет уникальное имя, а каждый из поддоменов имеет уникальное имя внутри своего домена. Имя домена может содержать до 63 символов. Каждый хост в сети Internet однозначно определяется своим полным доменным именем (fully qualified domain name, FQDN), которое включает имена всех доменов по направлению от хоста к корню. Пример полного DNS-имени: citint.dol.ru.

Автоматизация процесса назначения IP-адресов узлам сети - протокол DHCP

Как уже было сказано, IP-адреса могут назначаться администратором сети вручную. Это представляет для администратора утомительную процедуру. Ситуация усложняется еще тем, что многие пользователи не обладают достаточными знаниями для того, чтобы конфигурировать свои компьютеры для работы в интерсети и должны поэтому полагаться на администраторов.

Протокол Dynamic Host Configuration Protocol (DHCP) был разработан для того, чтобы освободить администратора от этих проблем. Основным назначением DHCP является динамическое назначение IP-адресов. Однако, кроме динамического, DHCP может поддерживать и более простые способы ручного и автоматического статического назначения адресов.

В ручной процедуре назначения адресов активное участие принимает администратор, который предоставляет DHCP-серверу информацию о соответствии IP-адресов физическим адресам или другим идентификаторам клиентов. Эти адреса сообщаются клиентам в ответ на их запросы к DHCP-серверу.

При автоматическом статическом способе DHCP-сервер присваивает IP-адрес (и, возможно, другие параметры конфигурации клиента) из пула наличных IP-адресов без вмешательства оператора. Границы пула назначаемых адресов задает администратор при конфигурировании DHCP-сервера. Между идентификатором клиента и его IP-адресом по-прежнему, как и при ручном назначении, существует постоянное соответствие. Оно устанавливается в момент первичного назначения сервером DHCP IP-адреса клиенту. При всех последующих запросах сервер возвращает тот же самый IP-адрес.

При динамическом распределении адресов DHCP-сервер выдает адрес клиенту на ограниченное время, что дает возможность впоследствии повторно использовать IP-адреса

другими компьютерами. Динамическое разделение адресов позволяет строить IP-сеть, количество узлов в которой намного превышает количество имеющихся в распоряжении администратора IP-адресов.

DHCP обеспечивает надежный и простой способ конфигурации сети TCP/IP, гарантируя отсутствие конфликтов адресов за счет централизованного управления их распределением. Администратор управляет процессом назначения адресов с помощью параметра "продолжительности аренды" (lease duration), которая определяет, как долго компьютер может использовать назначенный IP-адрес, перед тем как снова запросить его от сервера DHCP в аренду.

Примером работы протокола DHCP может служить ситуация, когда компьютер, являющийся клиентом DHCP, удаляется из подсети. При этом назначенный ему IP-адрес автоматически освобождается. Когда компьютер подключается к другой подсети, то ему автоматически назначается новый адрес. Ни пользователь, ни сетевой администратор не вмешиваются в этот процесс. Это свойство очень важно для мобильных пользователей.

Протокол DHCP использует модель клиент-сервер. Во время старта системы компьютер-клиент DHCP, находящийся в состоянии "инициализация", посылает сообщение discover (исследовать), которое широковещательно распространяется по локальной сети и передается всем DHCP-серверам частной интерсети. Каждый DHCP-сервер, получивший это сообщение, отвечает на него сообщением offer (предложение), которое содержит IP-адрес и конфигурационную информацию.

Компьютер-клиент DHCP переходит в состояние "выбор" и собирает конфигурационные предложения от DHCP-серверов. Затем он выбирает одно из этих предложений, переходит в состояние "запрос" и отправляет сообщение request (запрос) тому DHCP-серверу, чье предложение было выбрано.

Выбранный DHCP-сервер посылает сообщение DHCP-acknowledgment (подтверждение), содержащее тот же IP-адрес, который уже был послан ранее на стадии исследования, а также параметр аренды для этого адреса. Кроме того, DHCP-сервер посылает параметры сетевой конфигурации. После того, как клиент получит это подтверждение, он переходит в состояние "связь", находясь в котором он может принимать участие в работе сети TCP/IP. Компьютеры-клиенты, которые имеют локальные диски, сохраняют полученный адрес для использования при последующих стартах системы. При приближении момента истечения срока аренды адреса компьютер пытается обновить параметры аренды у DHCP-сервера, а если этот IP-адрес не может быть выделен снова, то ему возвращается другой IP-адрес.

В протоколе DHCP описывается несколько типов сообщений, которые используются для обнаружения и выбора DHCP-серверов, для запросов информации о конфигурации, для

продления и досрочного прекращения лицензии на IP-адрес. Все эти операции направлены на то, чтобы освободить администратора сети от утомительных рутинных операций по конфигурированию сети.

Однако использование DHCP несет в себе и некоторые проблемы. Во-первых, это проблема согласования информационной адресной базы в службах DHCP и DNS. Как известно, DNS служит для преобразования символьных имен в IP-адреса. Если IP-адреса будут динамически изменятся сервером DHCP, то эти изменения необходимо также динамически вносить в базу данных сервера DNS. Хотя протокол динамического взаимодействия между службами DNS и DHCP уже реализован некоторыми фирмами (так называемая служба Dynamic DNS), стандарт на него пока не принят.

Во-вторых, нестабильность IP-адресов усложняет процесс управления сетью. Системы управления, основанные на протоколе SNMP, разработаны с расчетом на статичность IP-адресов. Аналогичные проблемы возникают и при конфигурировании фильтров маршрутизаторов, которые оперируют с IP-адресами.

Наконец, централизация процедуры назначения адресов снижает надежность системы: при отказе DHCP-сервера все его клиенты оказываются не в состоянии получить IP-адрес и другую информацию о конфигурации. Последствия такого отказа могут быть уменьшены путем использовании в сети нескольких серверов DHCP, каждый из которых имеет свой пул IP-адресов.

1.3. Система доменных имен DNS

DNS - это централизованная служба, основанная на распределённой базе отображений "доменное имя - IP-адрес". Она использует в своей работе протокол типа "клиент-сервер". DNS-серверы поддерживают распределённую базу отображений, а DNS-клиенты обращаются к серверам с запросами о разрешении доменного имени в IP-адрес.

Служба DNS использует текстовые файлы, которые администратор подготавливает вручную. Однако служба DNS хранит только часть имён сети, а не все имена. При росте количества узлов в сети проблема масштабирования решается созданием новых доменов и подддоменов имен и добавлением в службу DNS новых серверов.

Каждый DNS-сервер кроме таблицы отображений содержит ссылки на DNS-серверы своих поддоменов, которые связывают отдельные DNS-серверы в единую службу DNS. Ссылки представляют собой IP-адреса соответствующих серверов. Процедура разрешения DNS-имени во многом аналогична процедуре поиска файловой системой адреса файла по его символьному имени. Для доменных имен, так же как и для символьных имен файлов, характерна независимость именования от физического местоположения.

Существует две основные схемы разрешения DNS-имен. В первом варианте работу по поиску IP-адреса координирует DNS-клиент. Такая схема взаимодействия называется нерекурсивной или итеративной, когда клиент сам итеративно выполняет последовательность запросов к разным серверам имен. Так как эта схема загружает клиента сложной работой, то она применяется редко.

Во втором варианте реализуется рекурсивная процедура, в которой клиент перепоручает работу своему серверу. Практически все DNS-клиенты используют рекурсивную процедуру. Для ускорения поиска IP-адресов DNS-серверы широко применяют процедуру кэширования проходящих через них ответов, которые хранятся от нескольких часов до нескольких дней.

2. Лабораторное задание

Лабораторное задание состоит из индивидуальных заданий. Вариант индивидуального задания получите у преподавателя.

2.1. Индивидуальные задания

Выполнение индивидуального задания заключается в освоении способов IP адресации.

Задание 1. Установить связь с другим компьютером в сети и определить свой и его IP-адрес.

Задание 2. Установить связь с другим компьютером и определить маску локальной сети, подсети и домена в целом.

Задание 3. Исследовать IP-адрес, определить его структуру и преобразовать этот адрес из двоичного формата в десятичный.

Задание 4. Исследовать IP-адрес, определить его структуру и преобразовать этот адрес из десятичного формата в шестнадцатиричный.

Задание 5. Установить связь с другим компьютером и определить типы классов IP-адресов, применяемых в данной локальной сети, подсети и в домене в целом.

Задание 6. Исследовать реализация IP-адреса и определить маску подсетей.

3. Содержание отчета

Отчет должен содержать:

- Цель работы.

- Теоретические сведения.
- Описание выполнения работы.
- Выводы по работе.

4. Контрольные вопросы:

1.Что представляет собой IP-адрес.

2. Что понимается под локальным адресом?

3. Как осуществляется преобразование IP-адреса из двоичного формата в десятичный.

4. Назовите классы IP-адресов.

5. Как определяется адрес назначения пакета?

5. Литература

1. Айден К., Фибельман Х., Крамер М. Аппаратные средства PC. –СПб.:BHV-Санкт-Петербург, 1996. - 680 с.

2. Дэвис, Барбер. Сети связи для вычислительных машин. – М.: Мир, 1992. – 452 с.

3. Мячев А.А., Иванов В.В. Интерфейсы вычислительных систем на базе мини- и микро-ЭВМ /Под ред. Б.Н. Наумова. – М.: Радио и связь, 1986. – 248 с.

4. Ларионов А.М. и др. Вычислительные комплексы, системы и сети. /Учебник для вузов. – Л.: Энергоатомиздат. Ленингр. отд-ние, 1987. – 288 с.

5. Ефимова О.В. и др. Практикум по компьютерной технологии.- М.: АБФ, 2002.- 260с.

6. Кутугина, Е. С., Тутубалин, Д. К. Информационные технологии: Учеб. пособие. — Томск, 2005.