# Структура взаимодействия в Интернете

Министерство образования РФ

Владимирский государственный университет

Муромский институт

Факультет \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Кафедра \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Контрольная работа

***По стандартизации и сертификации***

***тема:*** Структура взаимодействия в Интернете.

**Руководитель:**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(фамилия, инициалы)

                                 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись)               (дата)                                            **Студент:**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(группа)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(фамилия, инициалы)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись)              (дата)

**Муром 2003 г.**

**План:**

**1. Иерархическая система сетевого взаимодействия**

### 2. Структура стека TCP/IP. Краткая характеристика протоколов

**3. Адресация в IP-сетях**

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

**б) Три основных класса IP-адресов**

**в) Соглашения о специальных адресах: broadcast, multicast, loopback**

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

**д) Отображение символьных адресов на IP-адреса: служба DNS**

**е) Автоматизация процесса назначения IP-адресов узлам сети - протокол DHCP**

**4. Заключение**

**5. Список литературы**

**1 Иерархическая система сетевого взаимодействия**

Информационное взаимодействие в сети Интернет строится в соответствии с правилами и требованиями общего международного стандарта ISO 7498 (ISO – International Organization of Standartization).

Этот стандарт имеет тройной заголовок «Информационные вычислительные системы – Взаимодействие открытых систем – Эталонная модель». Обычно его называют короче – «Эталонная модель взаимодействия открытых систем». Публикация этого стандарта в 1983 году подвела итог многолетней работы многих известных телекоммуникационных компаний и стандартизирующих организаций.

Основной идеей, которая положена в основу этого документа, является разбиение процесса информационного взаимодействия между системами на уровни с четко разграниченными функциями.

Идея такого разбиения не была революционной. Можно вспомнить, что слоистую архитектуру имели информационные взаимодействия в сетях SNA (System Network Archithecture).

В качестве прообраза модели взаимодействия OSI (Open System Interconnection) была использована структура, предложенная ANSI (American National Standarts Institute). Основные работы по созданию текста документа были выполнены CCITT (Consultative Committee for International Telegraphy), а итоговый документ появился в виде стандарта ISO. Статус стандарта ISO важен для данного документа, поскольку ISO 7498 является стандартом стандартов в области телекоммуникаций.

Преимущества слоистой организации взаимодействия заключается в том, что она обеспечивает независимую разработку уровневых стандартов, модульность аппаратуры и программного обеспечения информационно-вычислительных систем и способствует тем самым техническому прогрессу в данной области.

При использовании многоуровневой модели проблема перемещения информации между узлами  сети разбивается на более мелкие и, следовательно, более легко разрешимые проблемы.

Многоуровневая модель четко описывает, каким образом информация проделывает путь через среду сети от одной прикладной программы, к примеру, обработки таблиц, до иной прикладной программы обработки тех же таблиц, находящейся на другом компьютере сети.

Предположим, например, что система А, имеет информацию для отправки в систему В. Прикладная программа системы А начинает взаимодействовать с уровнем 4 системы А (верхний уровень), который, в свою очередь, начинает взаимодействовать с уровнем 3 системы А, и т.д. – до уровня 1 системы А. Задача уровня 1 отдавать, а потом забирать информацию из физической среды сети.

*Поскольку информация, которая должна быть отослана, проходит вниз через уровни системы, по мере этого продвижения она становится все меньше похожей на человеческий язык и все больше похожей на ту информацию, которую понимают компьютеры, а именно «единицы» и «нули».*

После того как информация проходит через физическую среду сети и поступает в систему В, она последовательно обрабатывается на каждом уровне системы В обратном порядке – сначала на уровне 1, затем на уровне 2 и т.д., пока, наконец, не достигнет прикладной программы системы В.

Многоуровневая модель не предполагает наличия непосредственной связи между одноименными уровнями взаимодействующих систем. Следовательно, каждый уровень А должен полагаться на услуги, предоставляемые ему смежными уровнями системы А, чтобы помочь осуществить связь с соответствующим уровнем системы В. Для того чтобы выполнить эту задачу, уровень 4 системы А должен воспользоваться услугами уровня 3 системы А, тогда уровень 4 будет называться «пользователем услуг», а уровень 3 – «источником услуг».

Информация по оказываемым услугам передается между уровнями в специальном информационном блоке, который называется заголовком. Заголовок обычно предшествует передаваемой информации. Предположим, что система А хочет отправить в систему В какой-либо текст, называемый «данные» или «информация». Этот текст передается из прикладной программы системы А в верхний уровень этой системы. Прикладной уровень системы А должен передать определенную информацию в прикладной уровень системы В, поэтому он помещается управляющую информацию своего уровня в виде заголовка перед фактическим текстом, который должен быть передан. Построенный таким образом информационный блок передается в уровень 3 системы А, который может предварить его своей собственной управляющей информацией, и т.д.

Размеры сообщения увеличиваются по мере того, как оно проходит вниз через уровни до тех пор, пока не достигнет сети, где оригинальный текст и вся связанная с ним управляющая информация перемещаются в систему В и поглощаются уровнем 1 системы В. Уровень 1 системы В отделяет от поступившей информации и обрабатывает заголовок 1 уровня, после чего он определяет, как обрабатывать поступивший информационный блок. Слегка уменьшенный в размерах информационный блок передается на уровень 2, который отделяет заголовок этого же уровня, анализирует его, чтобы узнать о действиях, которые он должен выполнить и т.д. Когда информационный блок наконец доходит до прикладной программы системы В, он должен содержать только оригинальный текст.

Структура заголовка и собственно  данных относительна и зависит от уровня, который в данный момент анализирует информационный блок. Например, на уровне 2 информационный блок состоит из заголовка этого же уровня и следующих за ним данных. Однако данные уровня 2 могут содержать заголовки уровней 3 и 4. Кроме того, заголовок уровня 2 является просто данными для уровня 1. Помимо заголовка на каждом уровне системы информационный блок завершается соответствующей контрольной суммой КонтСум. Данная модель напоминает собой вложенные друг в друга матрешки. Самая маленькая из них – это и есть пользовательские данные, а все остальные служат для доставки данных в точку назначения.

Иными словами, в результате работы этого механизма каждый пакет более высокого уровня вкладывается в «конверт» протокола нижнего уровня. Здесь уместно провести аналогию с обычными постовыми отправлениями. Так, например, если вы пишите обычное письмо и вкладываете его в конверт с адресом, то текст письма будет информационным сообщением, которое вы хотите отправить, а конверт - заголовком «почтового» протокола. На почте ваше письмо перекладывают в мешок (протокол низкого уровня) с письмами того же или близкого назначения и т.п. Электронные протоколы работают по той же схеме, только доставку и целостность обычных писем обеспечивают добросовестность служащих отделений связи, а электронным протоколам приходится следить за этим самостоятельно.

В соответствии с ISO 7498 выделяются семь уровней (слоев) информационного взаимодействия:

7. Уровень приложения

6. Уровень представления

5. Уровень сессии

4. Транспортный уровень

3. Сетевой уровень

2. Канальный уровень

1. Физический уровень

Информационное взаимодействие двух или более систем, таким образом, представляет собой совокупность информационных взаимодействий уровневых подсистем, причем каждый слой локальной информационной системы взаимодействует только с соответствующим слоем удаленной системы.

*Протоколом называется набор алгоритмов (правил) взаимодействия объектов одноименных уровней.*

Слои (уровни) одной информационной системы также взаимодействуют друг с другом, причем в непосредственном взаимодействии участвуют только соседние уровни. Как правило, средний уровень пользуется услугами, которые ему предоставляет нижний уровень, а сам, в свою очередь, предоставляет услуги для верхнего уровня.

*Интерфейсом мы будем называть совокупность правил, в соответствии с которыми осуществляется взаимодействие с объектом данного уровня.*

Иерархическая организация сетевого взаимодействия позволяет обеспечивать преемственность разработанных структур и их быструю адаптацию к изменениям, происходящим в технологиях передачи данных. Например, при переходе на новый способ передачи данных по физическому носителю, изменения коснуться только нижних уровней и совсем не затронут верхние в том случае, если система протоколов организована в соответствии с требованиями ISO 7498. На практике требования данного стандарта реализуются в виде стека протоколов.

*Стеком называется иерархически организованная группа взаимодействующих протоколов.*

Протоколы, которые входят в стек, имеют специализированный интерфейс и предназначены для взаимодействия только с протоколами соответствующих уровней данного стека. В качестве примеров таких стеков можно привести стек TCP/IP и протоколы X.25.

Уровни 7-5 считаются верхними и, как правило, не отражают специфики конкретной сети. Блок данных пользователя (сообщение) этими уровнями рассматривается как единое целое. Изменения могут испытывать только сами данные.

Уровни 1-3 и иногда 4 считаются нижними уровнями OSI. На каждом из этих уровней определяется свой формат представления данных. При прохождении по стеку с 4-го уровня до первого сообщение пользователя последовательно фрагментируется и преобразуется в последовательность блоков данных соответствующего уровня.

*Процесс помещения фрагментированных блоков данных одного уровня в блоки данных другого уровня называют инкапсуляцией.*

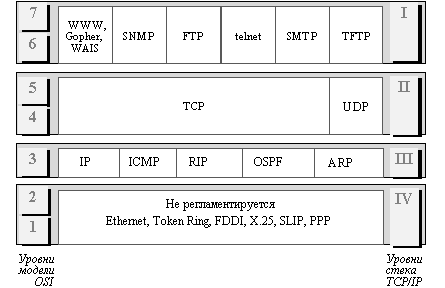
Обычно инкапсулируются данные протоколов верхних уровней в блоки данных протоколов нижних уровней (сетевой - канальный), но также может выполняться инкапсуляция для протоколов одноименных уровней (IP-X.25).

### 

### 2 Структура стека TCP/IP. Краткая характеристика протоколов

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 рисунке. Протоколы TCP/IP делятся на 4 уровня.



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, которым 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.

**3 Адресация в IP-сетях**

**а) 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 МАС-адрес сетевого адаптера или порта маршрутизатора, например, 11-А0-17-3D-BC-01. Эти адреса назначаются производителями оборудования и являются уникальными адресами, так как управляются централизовано. Для всех существующих технологий локальных сетей МАС-адрес имеет формат 6 байтов: старшие 3 байта - идентификатор фирмы производителя, а младшие 3 байта назначаются уникальным образом самим производителем. Для узлов, входящих в глобальные сети, такие как Х.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.

**б) 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.

Класс А

|  |  |  |
| --- | --- | --- |
| 0 | N сети | N узла |

Класс В

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | 0 | N сети | N узла |

Класс С

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 1 | 0 | N сети | N узла |

Класс D

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | 1 | 1 | 0 | адрес группы multicast |

Класс Е

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1 | 1 | 1 | 1 | 0 | зарезервирован |

Рис. 1. Структура IР-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 А, и номер сети занимает один байт, остальные 3 байта интерпретируются как номер узла в сети. Сети класса А 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 А количество узлов должно быть больше 216 , но не превышать 224.
* Если первые два бита адреса равны 10, то сеть относится к классу В и является сетью средних размеров с числом узлов 28 - 216. В сетях класса В 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 С 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 Е, он зарезервирован для будущих применений.

В таблице приведены диапазоны номеров сетей, соответствующих каждому классу сетей.

|  |  |  |
| --- | --- | --- |
| **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 |

**в) Соглашения о специальных адресах: broadcast, multicast, loopback**

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

* если IР-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.

**г) 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 МАС-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 известный 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 рисунке 2 показан формат пакета протокола ARP для передачи по сети Ethernet.

|  |  |  |
| --- | --- | --- |
| **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 обращается к выделенному маршрутизатору с запросом и автоматически получает ответ без участия администратора.

**д) 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.

**е) 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.

**4 Заключение**

В этой работе описана лишь малая часть тех стандартов которые используются на сегодняшний день.

**5 Список литературы**

1. Филимонов А. Протоколы Интернета. БХВ-Петербург., 2003 г.

2. Узел в Интернете: http://www.citforum.ru