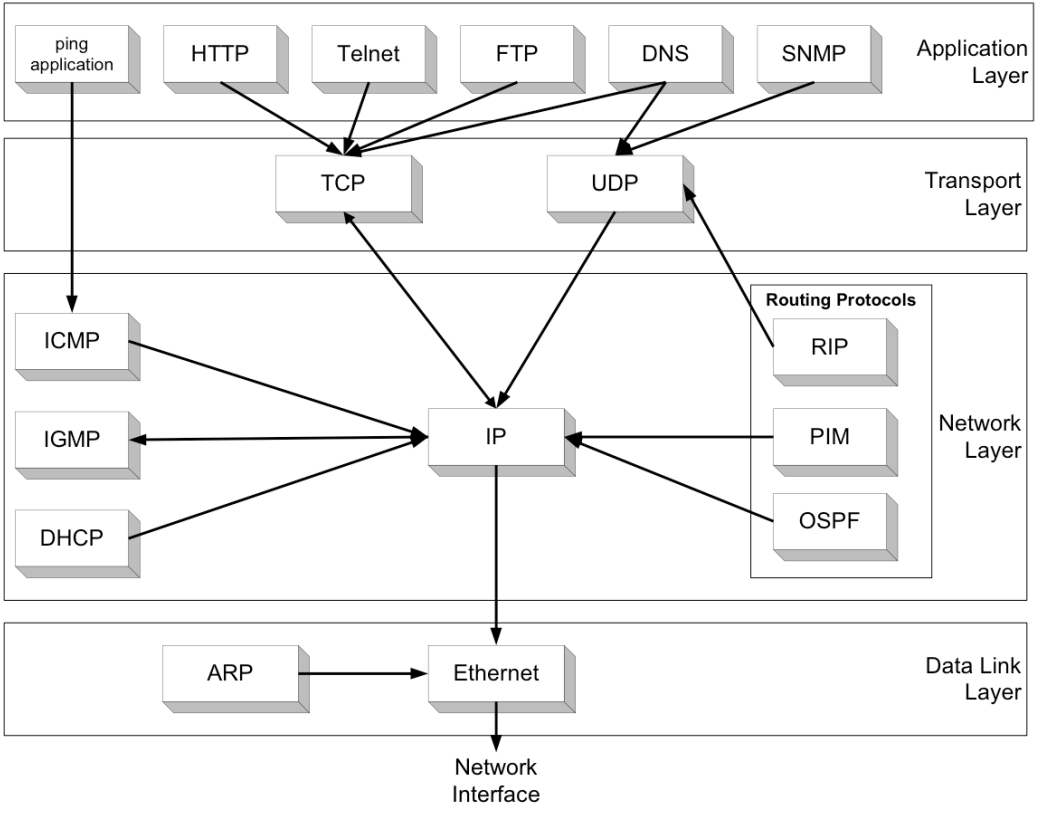
IP протокол.

Содержание темы.

1. Назначение протоколов сетевого уровня
2. Классификация и типы протоколов сетевого уровня
3. Назначение и алгоритм работы сетевого протокола IP
4. Функции IPv4
   - Инкапсуляция
   - Адресация
   - Маршрутизация
   - Фрагментация
   - Идентификация
   - Параметризация
5. Блок-схема управления IP на узле и алгоритмы работы модулей IP
6. Упражнения.
7. Протокол IPv6
8. Лабораторный практикум:
   - Исследование IP дейтаграмм (sniffing) – сбор, фильтрация, анализ.
   - Адресация сетей и узлов, разбиение сетей на подсети.
   - Статическая маршрутизация.
   - Динамическая маршрутизация.

1. Назначение протоколов сетевого уровня.



Протоколы сетевого уровня (Network layer) отвечают за:

- назначение логических адресов и их трансляцию в физические;
- за коммутацию и перенаправление;
- определение кратчайшего маршрута от системы-отправителя к системе-получателю;
- за отслеживание неполадок и затворов в сети;
- за передачу от системы-отправителя к системе-получателю;
- за управление групповой передачей данных по интерсети.

Коммутация в локальной сети происходит на основе MAC-адресов, поэтому IP-модуль пользуется таблицей соответствия вида IP-адрес – MAC-адрес, которую заполняет протокол нахождения адреса ARP (Address Resolution Protocol), сам IP-адрес может назначаться вручную или автоматически через DHCP (Dynamic Host Configuration Protocol).

Чтобы найти оптимальный маршрут, IP-модуль использует таблицу маршрутизации (ТМ), которую составляют протоколы маршрутизации (RIP, OSPF) и другие компоненты системы.

О возникших проблемах маршрутизаторы и узлы извещают друг друга при помощи протокола управляющих сообщений ICMP (Internet Control Message Protocol).

Групповая рассылка производится протоколами управления группами: в локальной сети IGMP (Internet Group Management Protocol) и глобально PIM (Protocol Independent Multicast).

В этой и следующих лекциях приводится подробное описание упомянутых протоколов.

2. Классификация и типы протоколов сетевого уровня.

Протоколы сетевого уровня перенаправляют данные от источника к получателю и могут быть разделены на два класса:

- **Протоколы с установкой соединения** (например, X.25) - начинают передачу данных с вызова или установки маршрута следования пакетов от источника к получателю. После чего начинают последовательную передачу данных и затем по окончании передачи разрывают связь.
- **Протоколы без установки соединения** (например, IP) - посылают данные, содержащие полную адресную информацию в каждом пакете. Каждый пакет содержит адрес отправителя и получателя. Далее каждое промежуточное сетевое устройство считывает адресную информацию и принимает решение о маршрутизации данных. Письмо или пакет данных передается от одного промежуточного устройства к другому до тех пор, пока не будет доставлено получателю. Протоколы без установки соединения не гарантируют поступление информации к получателю в том порядке, в котором она была отправлена, т.к. разные пакеты могут пройти разными маршрутами. За восстановления порядка данных при использовании сетевых протоколов без установки соединения отвечают транспортные протоколы.

Типы протоколов сетевого уровня:

- Протоколы взаимодействия
  - IP/IPv4/IPv6 - Internet Protocol
  - IPX – Internetwork Packet Exchange (протокол межсетевого обмена)
  - X.25 (частично этот протокол реализован на уровне 2)
  - CLNP - Connection Less Network Protocol (протокол без организации соединений)
- IPsec - Internet Protocol Security
- Протоколы маршрутизации
  - RIP - Routing Information Protocol
  - OSPF - Open Shortest Path First
  - IS-IS - Intermediate System to Intermediate System
  - BGP - Border Gateway Protocol
  - PIM - Protocol Independent Multicast
- Протоколы управления
  - ICMP - Internet Control Message Protocol
  - IGMP - Internet Group Management Protocol
- Протоколы разрешения и назначения адресов (работают на границах уровней)
  - ARP - Address Resolution Protocol (в IPv6 функции берёт ICMPv6)
  - DHCP - Dynamic Host Configuration Protocol
  - SLAAC - State-Less Address AutoConfiguration

3. Назначение и алгоритм работы сетевого протокола IP.

IP (Internet Protocol) - краеугольный камень стека (набора) TCP/IP, названного так по двум составляющим его протоколам (IP и TCP), которые в паре обеспечивают сетевой транспортный сервис.

Наиболее распространенными являются протоколы IPv4 и IPv6. Будущее за IPv6.

В общем случае протокол IPv6 несовместим с протоколом IPv4, но, зато совместим со всеми остальными протоколами Интернета, включая TCP, UDP, ICMP, OSPF, DNS (иногда требуются небольшие изменения).

В интерсети TCP/IP протокол IP отвечает за передачу данных от исходной до целевой системы. Он не ориентирован на соединение. В TCP/IP службы с ориентацией на соединения работают на транспортном уровне, благодаря чему удается избежать ориентации на соединения на сетевом уровне и сократить издержки на передачу излишних управляющих данных.

Протокол транспортного уровня, например, TCP или UDP, передает данные на сетевой уровень, а IP инкапсулирует их в кадр, добавляя свой заголовок и получая в результате дейтаграмму (datagram) так, как показано на рис.

Рис. IP помещает данные транспортного уровня в дейтаграмму.

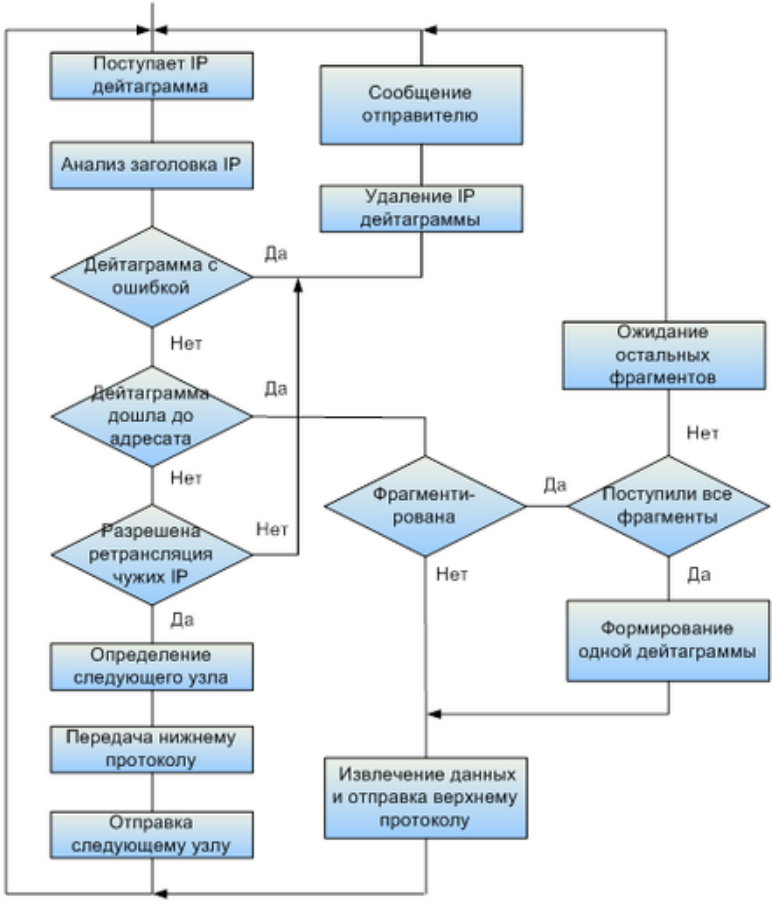
Дейтаграмма адресована именно тому компьютеру, которому предназначены данные, независимо от того, находится он в локальной или удаленной сети. Не считая нескольких небольших модификаций, на всем пути к целевой системе дейтаграмма сохраняет первоначальный вид. Закончив создание дейтаграммы, IP передает ее протоколу канального уровня для передачи в сеть.

В процессе передачи данных разные системы могут добавлять к дейтаграмме различные заголовки протоколов канального уровня, но сами данные остаются неизменными. Открыть и использовать содержимое разрешается только получателю данных.

Протоколы различных уровней модели OSI по-разному называют создаваемые ими структуры. Например, то, что на канальном уровне кадр, для сетевого уровня будет дейтаграммой. Более общее название для структурной единицы данных на любом уровне - это пакет (packet).

3.1. Алгоритм работы IP.

Алгоритм работы протокола ip на узле сети выглядит следующим образом:



4. Функции IPv4.

Стандарт IPv4.

Протоколы TCP/IP описаны в документах RFC (Requests For Comments), публикуемых рабочей группой IETF (Internet Engineering Task Force). В отличии от большинства сетевых стандартов, спецификации TCP/IP предоставляются в общее пользование и бесплатно доступны в Интернете на многих сайтах, в том числе на домашней странице IETF по адресу http://www.ietf.org. Спецификация протокола IP опубликована в RFC 791 (сентябрь 1981 г.) и ратифицирована как Internet Standard 5.

Протокол IP выполняет несколько важных сетевых функций, в том числе:

1. **Инкапсуляцию** - упаковка пакета данных транспортного уровня в дейтаграмму.
2. **Адресацию** - идентификация систем в сети по их IP-адресам.
3. **Маршрутизацию** - определение наиболее эффективного пути к целевой системе.
4. **Фрагментацию** - разбиение данных на фрагменты, по размеру подходящие для передачи по сети (MTU).
5. **Идентификацию** протокола верхнего уровня, сгенерировавшего данные для IP.
6. **Параметризацию** – установка опций IP для выполнения специфичных задач.

4.1. Инкапсуляция.

Заголовок, добавляемый протоколом IP к данным, полученным от протокола транспортного уровня, обычно имеет длину 20 байт. Формат дейтаграммы (пакета) IPv4 показан на рис.

Octet	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
00	Version				IHL				DSCP						ECN		Total Length															
04	Identification																Flags			Fragment Offset											
08	Time To Live								Protocol								Header Checksum															
12	Source IP Address																															
16	Destination IP Address																															
20-n	Options (variable dimension) min=0, max=10x32 bit																													PAD		
n+1-m	Data (variable dimension)																															

Рис. Формат дейтаграммы IPv4.

Назначение полей IPv4 пакета (дейтаграммы) следующие:

- **Version** (4 бита) - версия протокола IP, использованная для создания дейтаграммы. Для IPv4 значение поля должно быть равно 4.
- **Internet Header Length (IHL)** (4 бита) - длина заголовка дейтаграммы, выраженная в 32-битовых (dword) словах. Именно это поле указывает на начало блока данных в пакете.

Обычно длина заголовка дейтаграммы равна 5 словам (20 байтам), но, если в дейтаграмму включены дополнительные параметры, она может быть и больше.

- **Differentiated Services Code Point (DSCP) –** используется для разделения трафика на классы обслуживания, например для установки чувствительному к задержкам трафику, такому как VoIP, большего приоритета. Вторая интерпретация этого поля определяет **Type Of Service (TOS)** – тип обслуживания. DSCP и TOS рассмотрены ниже.
- **Explicit Congestion Notification (ECN)** - указатель перегрузки - предупреждение о перегрузке сети без потери пакетов (управление IP-потоком). Является необязательной функцией и используется только если оба хоста её поддерживают.
- **Total Length** (2 байта) - длина дейтаграммы в байтах с учетом данных и всех полей заголовка. Минимальное корректное значение для этого поля равно 20, максимальное 65535, большие пакеты не всегда можно передать, тогда их делят на части (MTU).
- **Identification** (2 байта) - уникальный идентификатор дейтаграммы назначаемый отправителем пакета. Для фрагментированного пакета все фрагменты должны иметь одинаковый идентификатор и не повторяться для разных пакетов, пока у обоих пакетов не истекло время жизни. Целевая система использует эту величину при сборке из пакетов дейтаграммы IP, которая была фрагментирована в процессе передачи.
- **Flags** (3 бита) - флаги, управляющие процессом фрагментации дейтаграммы.

Octet	16	17	18
06	0	DF	MF

  - 16 бит должен быть всегда равен нулю.

- 17 бит DF (don't fragment) определяет возможность фрагментации пакета (DF=1 запрет на фрагментацию для промежуточных узлов). Флаг DF=1 используется в случаях, когда отправителю известно, что у получателя нет достаточно ресурсов по восстановлению пакетов из фрагментов. Если пакет с флагом DF=1 должен быть передан через сеть с недостаточным MTU, то маршрутизатор **вынужден будет его отбросить** (и сообщить об этом отправителю посредством протокола ICMP).
- 18 бит MF (more fragments) показывает, является ли этот пакет последним (MF=0) или не последним (MF=1) в цепочке пакетов.

- **Fragment Offset** (13 битов) - значение, определяющие положение фрагмента во фрагментированной дейтаграмме. Смещение задается количеством восьми байтовых блоков, поэтому это значение требует умножения на 8 для перевода в байты. При делении данных пакета, размер всех фрагментов, кроме последнего, должен быть кратен 8 байтам.
- **Time To Live** (1 байт) - количество сетей (секунд), которое дейтаграмме разрешается пройти на пути к целевой системе. Каждый маршрутизатор, пересылающий дейтаграмму, уменьшает значение в этом поле на 1. Когда значение становится равным 0, дейтаграмма прекращает существование, пакет должен быть отброшен и отправителю пакета может быть послано сообщение Time Exceeded (ICMP код 11 тип 0).
- **Protocol** (1 байт) - код протокола, сгенерировавшего информацию в поле данных (например, TCP -6, UDP-17, ICMP-1). Поле Protocol рассмотрено дальше.
- **Header Checksum** (2 байта) - контрольная сумма заголовка IP, используемая для обнаружения ошибок, вычисляется на передающей, промежуточной и приемной стороне в соответствии с RFC 1071.

- **Source IP Address** (4 байта) - IP-адрес системы, создавший дейтаграмму.
- **Destination IP Address** (4 байта) - IP-адрес системы, в которую направляется дейтаграмма. Адресация IP рассмотрена ниже.
- **Options** (переменной длины) - необязательное поле для одного или нескольких IP-параметров. Размер и содержимое этого поля определяются количеством и типом параметров, размер не более 40 байт. Опции используются администраторами сетей для проверки работоспособности определенных маршрутов и для обычных пользователей не нужны. Опции IP рассмотрены ниже.
- **PAD** - padding - необязательное поле заполнитель до кратности заголовка 32 бит, для выравнивания Options.
- **Data** (переменной длины) - информация, сгенерированная протоколом, код которого указан в поле Protocol. Размер этого поля зависит от протокола канального уровня, используемого сетью, в которую система передает дейтаграмму (MTU).

Более подробно о формате IP дейтаграмм см. в лабораторной работе «Исследование IP дейтаграмм с помощью сетевого анализатора Wireshark».

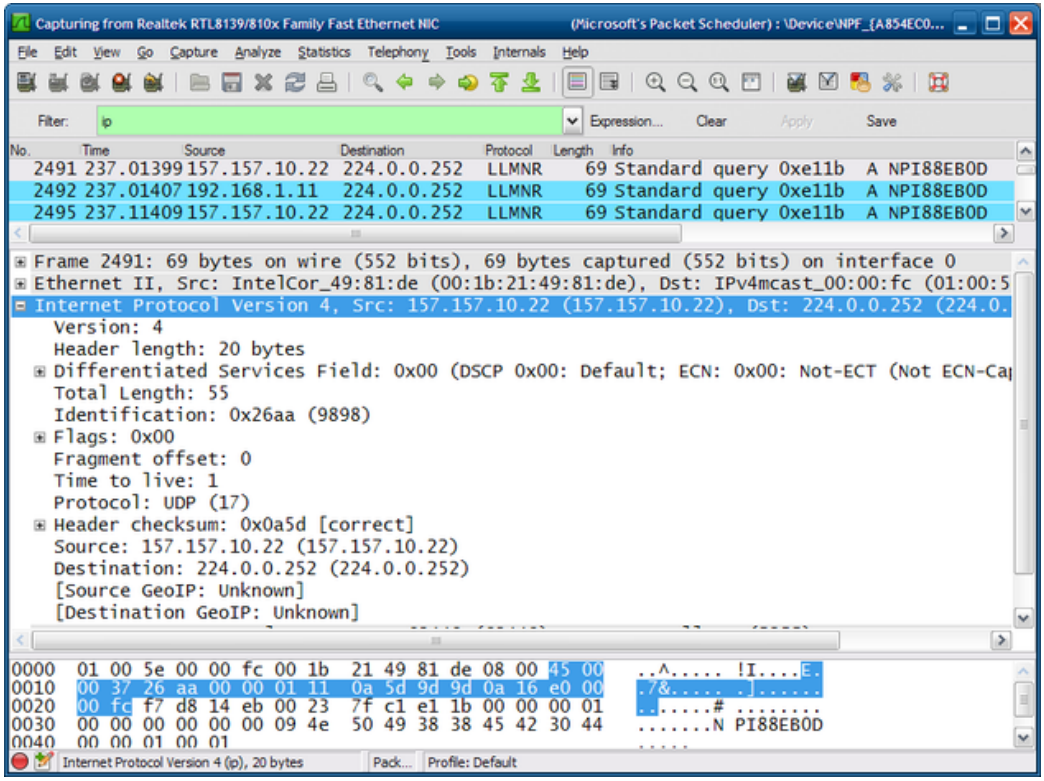


Рис. Пакет IPv4 перехваченный с помощью сниффера Wireshark.

4.1.1. Контрольная сумма заголовка IP (IP Header Checksum).

Метод обнаружения ошибок, используемый в большинстве протоколов TCP/IP (IP, ICMP, UDP, IGMP, TCP), назван контрольной суммой (Checksum, см. RFC 1071 (1988). Контрольная сумма защищает от искажений, которые могут возникнуть при передаче пакета.

Контрольная сумма вычисляется в передатчике, полученное значение инвертируется и посылается с пакетом. Приемник повторяет те же самые вычисления для пакета, но, уже включая контрольную сумму. Если результат вычисления равен 0, то пакет принимается; в противном случае он отклоняется. Используется свойство сложения числа с собственной инверсией, которая всегда даёт все 1, а последующая инверсия результата должна давать все 0, например, 1010+0101=1111 =>0000.

Если контрольная сумма на стороне передачи не вычисляется, то для сигнализации об этом Checksum устанавливается в ноль, если Checksum при вычислении действительно оказалась равной нулю, то она заменяется на 65535 (все биты =1). Отказ от вычисления Checksum часто используют в ICMP и TCP.

Вычисление IP Header Checksum передатчиком.

Сначала значение поля контрольной суммы устанавливается в 0. Затем заголовок разбивается на 16-битные числа. Эти числа последовательно складываются. Если в результате получаются числа более, чем 16 бит, то оно тоже разбивается на два 16-битных числа которые складываются. Полученное итоговое число переписываем в инверсном бинарном виде (0->1, 1->0). Результат записывает в поле "Контрольная сумма".

Вычисление IP Header Checksum приемником.

Сначала заголовок полученного пакета разбивается на 16-битные числа, числа последовательно складываются (как на передатчике). Затем результат инвертируется. Если конечный результат равен 0, пакет принимается; в противном случае пакет отклоняется. Сообщение об ошибке не генерируется. Теперь задача верхних уровней (ICMP, IGMP, UDP и TCP) каким-либо образом определить, что дейтаграмма отсутствует, и обеспечить её повторную передачу.

Продемонстрируем процесс для примера 6 байт данных и 2 байт Checksum.

Передатчик	**Приёмник**
1000 0110 0101 1110 - 1-е 16-bit число 0000 0000 0000 0000 - 2-е 16-bit число (Checksum) 1010 1100 0110 0000 - 3-е 16-bit число 0111 0001 0010 1010 - 4-е 16-bit число	1000 0110 0101 1110 - 1-е 16-bit число 0101 1100 0001 0110 - 2-е 16-bit число (Checksum) 1010 1100 0110 0000 - 3-е 16-bit число 0111 0001 0010 1010 - 4-е 16-bit число
1000 0110 0101 1110 - 1-е 16-bit число + 0000 0000 0000 0000 - 2-е 16-bit число (Checksum) = 0 1000 0110 0101 1110 - сумма 1 без переполнения + 1010 1100 0110 0000 - 3-е 16-bit число = 1 0011 0010 1011 1110 - сумма с переполнением + \- - - - - - - - - - - - - -> 1 - дополнительная единица = 0 0011 0010 1011 1111 - сумма 2 без переполнения + 0111 0001 0010 1010 - 4-е 16-bit число = 0 1010 0011 1110 1001 - сумма без переполнения = 0101 1100 0001 0110 - инверсия =Checksum	1000 0110 0101 1110 - 1-е 16-bit число + 0101 1100 0001 0110 - 2-е 16-bit число (Checksum) = 0 1110 0010 0111 0100 - сумма 1 без переполнения + 1010 1100 0110 0000 - 3-е 16-bit число = 1 1000 1110 1101 0100 - сумма с переполнением + \- - - - - - - - - - - - - -> 1 - дополнительная единица = 0 1000 1110 1101 0101 - сумма 2 без переполнения + 0111 0001 0010 1010 - 4-е 16-bit число = 0 1111 1111 1111 1111 - сумма без переполнения = 0000 0000 0000 0000 - инверсия=0 (ошибок нет)

Контрольная сумма в IP-пакете защищает лишь заголовок, но не данные так как:

- заголовок IP изменяется в каждом маршрутизаторе, который он посещает, а данные не меняются. Если разрешить контрольной сумме проверять также и данные, то это увеличит времени обработки для каждого маршрутизатора;
- все протоколы высокого уровня (ICMP, IGMP, UDP и TCP), инкапсулируемые в IP-дейтаграмму, имеют собственную Checksum, охватывающую их заголовки и данные.

Используемый алгоритм Checksum сегодня считается слабой защитой. В общем случае распределенным сетевым приложениям рекомендуется использовать дополнительные программные средства для гарантирования целостности передаваемой информации.

4.2. Адресация IPv4.

Уникальность IP по сравнению с другими протоколами сетевого уровня состоит в том, что он обладает собственной системой адресов для идентификации компьютеров в интерсети почти любого размера (в других протоколах сетевого уровня, например, в NetBEUI или IPX, для идентификации компьютеров в ЛВС используются имена или аппаратные адреса).

Адрес IP имеет длину 32 бита и состоит из идентификатора сети и идентификатора хоста. Хостом (host) в TCP/IP называется сетевой адаптер компьютера или другого устройства. Обычно говорят об IP-адресе компьютера, но в действительности адрес принадлежит сетевому адаптеру (сетевой плате). Если на компьютере (например, маршрутизаторе) установлено два адаптера или адаптер и модем для удаленного соединения с сетью, у него будет два IP-адреса – по одному для каждого интерфейса.

IP-адреса, записанные системой в поля Source IP Address и Destination IP Address заголовка IP, идентифицируют систему, создавшую пакет, и систему, которой он предназначен. Если пакет не покинет пределы ЛВС, целевой IP-адрес указывает на ту же систему, что и целевой адрес в заголовке протокола канального уровня. Если пакет адресован системе в другой сети, целевые адреса протоколов сетевого и канального уровней различаются.

IP - сквозной протокол, т. е. он полностью отвечает за доставку данных целевой системе, не ограничиваясь их перемещением по локальной сети, как протокол канального уровня.

IP и ARP.

Протоколы канального уровня с IP-адресами не работают, поэтому для передачи дейтаграммы IP должен сообщить протоколу канального уровня аппаратный адрес системы в локальной сети. Для этого IP прибегает к помощи другого протокола из набора TCP/IP - протокола разрешения адреса **ARP (Address Resolution Protocol)**. ARP рассылает широковещательное сообщение с IP-адресом системы в локальной сети. Система, которой принадлежит этот IP-адрес, отвечает на него, подставляя в ответное сообщение свой аппаратный адрес.

Если целевая система дейтаграммы находится в локальной сети, в сообщении ARP содержится ее IP-адрес. Если целевая система находится в другой сети, IP-адрес в сообщении ARP принадлежит маршрутизатору. Получив ответ на сообщение ARP, протокол IP в системе-источнике передает дейтаграмму протоколу канального уровня, сопроводив ее аппаратным адресом, необходимым для построения кадра.

Более подробно об адресации см. в лабораторной работе «Адресация, сети, подсети, суперсети».

4.3. Маршрутизация.

Маршрутизацией (routing) называется процесс выбора в интерсети самого эффективного маршрута для передачи дейтаграмм от системы-отправителя к системе-получателю.

Маршрутизаторы при обработке дейтаграмм IP могут учитывать байт TOS/DSCP.

Байт TOS содержащий набор критериев, определяющих тип обслуживания IP-пакетов, представлен на рисунке.

Octet	8	9	10	11	12	13	14	15
01	Precedence			D	T	R	C	-

Биты категории срочности (Precedence)

- 8-10 - приоритет данного IP-сегмента при перегрузках (в пределах от 000 до 111 в двоичном представлении). Если маршрутизатор перегружен и должен удалить некоторые дейтаграммы, то первыми будут удалены дейтаграммы с самой низкой категорией срочности. Некоторые дейтаграммы в Интернете более важны, чем другие. Например, дейтаграмма, используемая для управления сетью, намного более срочная и важная, чем дейтаграмма, содержащая дополнительную информацию для группы. В настоящее время подполе категории срочности **не используется**. Это, как ожидается, будет функционировать в будущем.

Биты типа обслуживания (DTRC). Хотя бит может иметь значение либо 0, либо 1, но в каждой дейтаграмме только один бит может иметь значение 1 (кроме security). Интерпретация битов дана в таблице

TOS Value	Description	Пояснения
0000	Default	Нормально (по умолчанию)
0001	Minimize Monetary Cost	Минимизация стоимости передачи IP-сегмента
0010	Maximize Reliability	Максимизация надежности передачи IP-сегмента (0 - нормальная, 1 - высокая надежность)
0100	Maximize Throughput	Максимизация пропускной способности маршрута, по которому должен отправляться IP-сегмент (0 - низкая, 1 - высокая пропускная способность)
1000	Minimize Delay	Минимизация времени задержки передачи IP-сегмента (0 - нормальная, 1 - низкая задержка)
1111	Maximize Security	

Таблица. Типы обслуживания по умолчанию.

Протокол	Биты обслуживания	Описание
ICMP	0000	Нормально
BOOTP	0000	Нормально
NNTP	0001	Минимизация стоимости
IGP	0010	Максимизация надежности
SNMP	0010	Максимизация надежности
TELNET	1000	Минимизация задержки
FTP (данные)	0100	Минимизация задержки
FTP (управление)	1000	Минимизация задержки
TFTP	1000	Максимизация пропускной способности
SMTP (команда)	1000	Минимизация задержки
DNS (UDP-запрос)	0100	Нормально
DNS (TCP-запрос)	0000	Нормально
DNS (Зона)	0100	Максимизация пропускной способности

Байт DSCP (Differentiated Services Code Point) - различные услуги

В этой трактовке первые 6 битов компонуют кодовую комбинацию подполя, а последние два бита не используются. Кодовая комбинация подполя может применяться двумя различными способами:

- Когда 3 самых правых бита — нулевые, 3 крайних левых бита интерпретируются так же, как биты категории срочности при интерпретации типа сервиса. Другими словами, это совместимо со старой интерпретацией TOS.
- Когда 2 самых правых бита — не все нули, 6 битов определяют 64 услуги, основанные на назначении приоритета с помощью Интернета или местных полномочий согласно табл. 4.3. Первая категория содержит 32 типа сервиса; вторая и третья содержит 16 типов каждая. Первая категория (числа 0, 2, 4, ..., 62) назначает полномочия Интернета (IETF — Internet Task Force). Вторая категория (3, 7, 11, 15, ..., 63) может использоваться местными организациями. Третья категория (1, 5, 9, ..., 61) является временной и может применяться для экспериментальных целей. Обратите внимание, что числа не непрерывны. Если бы они были непрерывны, то первая категория расположилась бы от 0 до 31, вторая от 32 до 47, и третья от 48 до 63. Это было бы несовместимо с интерпретацией бит типа услуг, потому что XXX000(включает 0, 8, 16, 24, 32, 40, 48 и 56) попадал бы во все три категории. Вместо этого, при этом методе назначения все эти услуги принадлежат категории 1. Эти назначения еще нельзя считать окончательными.

Таблица 4.3. Значение битов кода

Категория	Кодовая комбинация	Назначенные полномочия
1	XXXX0	Интернет
2	XXX11	Местные
3	XXX01	Временные и экспериментальные

Более подробно о маршрутизации непосредственно см. тему “Сетевая маршрутизация” и в лабораторном практикуме по маршрутизации.

4.3.1. Замечания по TTL.

Поле TTL (время жизни) это 8-битное поле, которое отправитель устанавливает в какое-либо значение. Рекомендуемое исходное значение указано в Assigned Numbers RFC и в настоящее время равно 64. Более старые системы устанавливают это значение в 15 или 32. ICMP эхо отклики часто отправляются с TTL, установленным в максимальное значение - 255.

Каждый маршрутизатор, который обрабатывает дейтаграмму, уменьшает значение TTL на единицу или на количество секунд, в течение которых маршрутизатор обрабатывал дейтаграмму. Так как большинство маршрутизаторов задерживает дейтаграмму меньше чем секунду, поле TTL, как правило, уменьшается на единицу и довольно точно соответствует количеству пересылок (хопов).

С помощью поля TTL предотвращается зацикливание дейтаграммы в петлях маршрутизации. Например, если маршрутизатор вышел из строя или имеет неверную конфигурацию, то TTL гарантирует что дейтаграмма будет уничтожена в петле маршрутизации.

Как правило, системы не должны получать дейтаграммы с TTL равным 0.

Когда маршрутизатор получает IP дейтаграмму с TTL равным либо 0, либо 1, он не должен отправлять эту дейтаграмму дальше, он уничтожает ее и посылает хосту, который ее отправил ICMP сообщение "время истекло" (time exceeded).

А конечный хост-приёмник должен доставить подобную дейтаграмму в приложение.

4.4. Фрагментация.

На пути пакета от отправителя к получателю могут встречаться локальные и глобальные сети разных типов с разными допустимыми размерами полей данных кадров канального уровня (Maximum Transfer Unit – MTU). Например, сети Ethernet могут передавать кадры, несущие до 1500 байт данных, в других сетях действуют свои ограничения.

Протокол	MTU
Hyperchannel	65 535
Token Ring (16 Мбит/с)	17 914
Token Ring (4 Мбит/с)	4 484
FDDI	4 352
Ethernet	1500
X.25	576
PPP	296

Протокол IP умеет передавать дейтаграммы, длина которых больше MTU промежуточной сети, за счет фрагментирования – разбиения “большого пакета” на некоторое количество частей (фрагментов), размер каждой из которых удовлетворяет промежуточную сеть. После того, как все фрагменты будут переданы через промежуточную сеть, они будут собраны на узле-получателе модулем протокола IP обратно в “большой пакет” и только потом переданы вышестоящему протоколу (TCP, UDP, ICMP и др.).

Отметим, что сборку пакета из фрагментов осуществляет только получатель, а не какой-либо

из промежуточных маршрутизаторов. Маршрутизаторы могут только фрагментировать пакеты, но не собирать их. Это связано с тем, что разные фрагменты одного пакета не обязательно будут проходить через одни и те же маршрутизаторы.

Для организации процессов фрагментации и дефрагментации в заголовках IP дейтаграмм используются поля Identification, Flags, Fragment Offset.

Если пакет с флагом DF=1 должен быть передан через сеть с недостаточным MTU, то маршрутизатор вынужден будет его отбросить (и сообщить об этом отправителю посредством протокола ICMP тип 3 код 4).

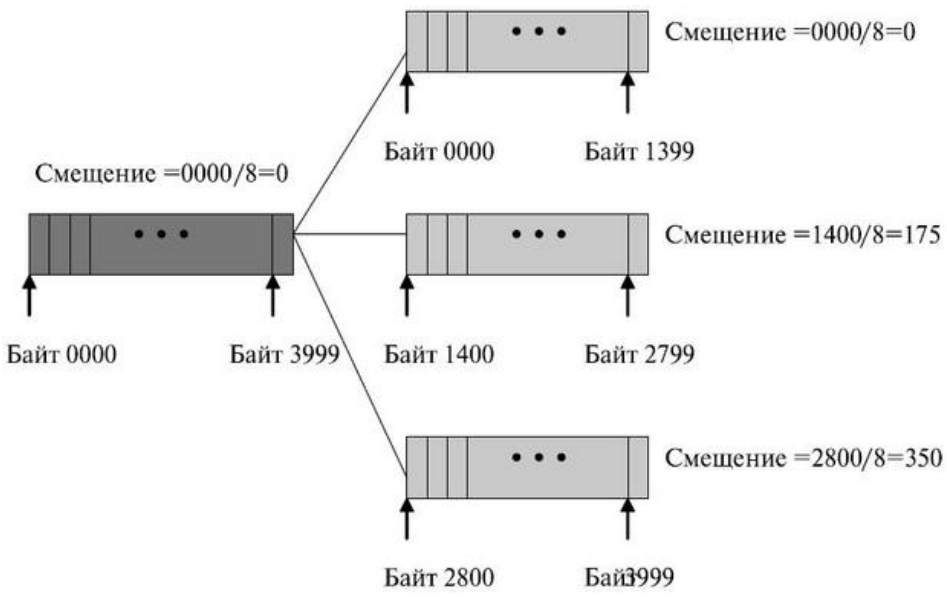


Рис. Пример фрагментации

4.4.1. Нежелательность фрагментации.

Когда IP дейтаграмма фрагментируется, каждый фрагмент становится пакетом, с собственным IP заголовком, и маршрутизируется независимо от других пакетов. Поэтому возможна ситуация когда дейтаграммы прибудут в конечный пункт назначения в другом порядке, нежели они были исходно отправлены и фрагментированы. Но, в IP заголовке хранится достаточно информации для того, чтобы дейтаграмма была корректно собрана (дефрагментирована) до передачи её следующему протоколу (например, TCP или UDP).

Существует одна особенность, которая делает фрагментацию нежелательной: если один фрагмент потерялся, дейтаграмма должна быть передана повторно целиком.

Это объясняется тем, что IP не имеет тайм-аутов и не осуществляет повторной передачи - за это несут ответственность верхние уровни. TCP осуществляет тайм-аут и повторную передачу, а UDP нет. Если это необходимо, то UDP приложения осуществляют тайм-аут и повторную передачу самостоятельно на уровне приложений.

Когда потерялся фрагмент из TCP сегмента, TCP отработает тайм-аут и повторно передаст TCP сегмент целиком (IP дейтаграмма). Не существует способа повторно передать только один фрагмент дейтаграммы. И действительно, если фрагментация была осуществлена промежуточным маршрутизатором, а не отправляющей системой, отправляющая система не сможет знать как дейтаграмма была фрагментирована в процессе передачи. Именно по причине повторной передачи всей дейтаграммы фрагментации стараются избежать.

Несмотря на то что существует возможность отправить дейтаграмму размером 65535 байт, большинство канальных уровней поделят подобную дейтаграмму на фрагменты. Более того, от хоста не требуется принимать дейтаграмму размером больше чем 576 байт. TCP делит пользовательские данные на части, поэтому это ограничение обычно не оказывает влияния на TCP. Что касается UDP, услугами которого пользуются многие приложения (RIP, TFTP, BOOTP, DNS, SNMP), то он ограничивает себя 512 байтами пользовательских данных, что даже меньше ограничения в 576 байт. Большинство приложений в настоящее время (особенно те, которые поддерживают NFS - Network File System) позволяют использовать IP дейтаграмму размером 8192 байта.

4.5. Идентификация протокола верхнего уровня.

Чтобы корректно обработать принятую дейтаграмму, целевая система должна знать, каким протоколом сгенерирована информация в поле данных. Для этого в поле Protocol заголовка IP записывается информация о протоколе верхнего уровня, от которого эти данные были получены (куда назначаются). В соответствии с этой информацией система-получатель передает входящие дейтаграммы соответствующему протоколу верхнего уровня.

Присвоенные номера протоколов можно найти на сайте IANA http://www.iana.org/assignments/protocol-numbers/protocol-numbers.xml.

Например, TCP -6, UDP -17, ICMP -1, IGMP -2, IPv4encapsulation -4, IPv6encapsulation -41, HIP -139, experimentation -253,254.

4.6. Параметризация.

Параметры (опции) IP задаются в необязательном поле Options, которое позволяет передавать в дейтаграммах дополнительные сведения.

Опции могут быть максимально 40 байт. Опции не требуются для каждой дейтаграммы. Они используются для испытания сети и отладки. Хотя опции — не обязательная часть заголовка IP, но, все применяемые средства должны быть способны обработать опции, если они окажутся в заголовке.

4.6.1. Формат опций.

Поле	Размер в битах	Описание
Копировать	1	Устанавливается в 1 если требуется копировать опции в заголовки всех фрагментов.
Класс опции	2	0 для «управляющих» опций и 2 для опций «измерений и отладки». 1 и 3 зарезервированы.
Номер (тип) опции	5	Указывает номер опции. Присвоенные номера опций размещаются на сайте IANA http://www.iana.org/assignments/ip-parameters/ip-parameters.xhtml.
Размер опции	8	Указывает размер опции (с учетом всех полей). Может не указываться для опций без аргументов.
Аргументы опции	Переменный	Дополнительные данные, используемые опцией.

4.6.2. Типы опций.

IANA определила более 30 опций, но, в настоящее время используются шесть опций. Две из них — 1-байтные, и они не требуют полей длины или данных. Четыре из них — многобайтные опции; они требуют полей длины и данных, см. рис.

4.6.3. Опция конец опции (End of Options List - EOOL) — однобайтовая опция, имеющая номер 0, используемая для того, чтобы дополнить конец поля опции. Однако она может использоваться только как последняя опция. Только один конец опции может использоваться опцией. После этой опции приемник ищет данные полезной нагрузки.

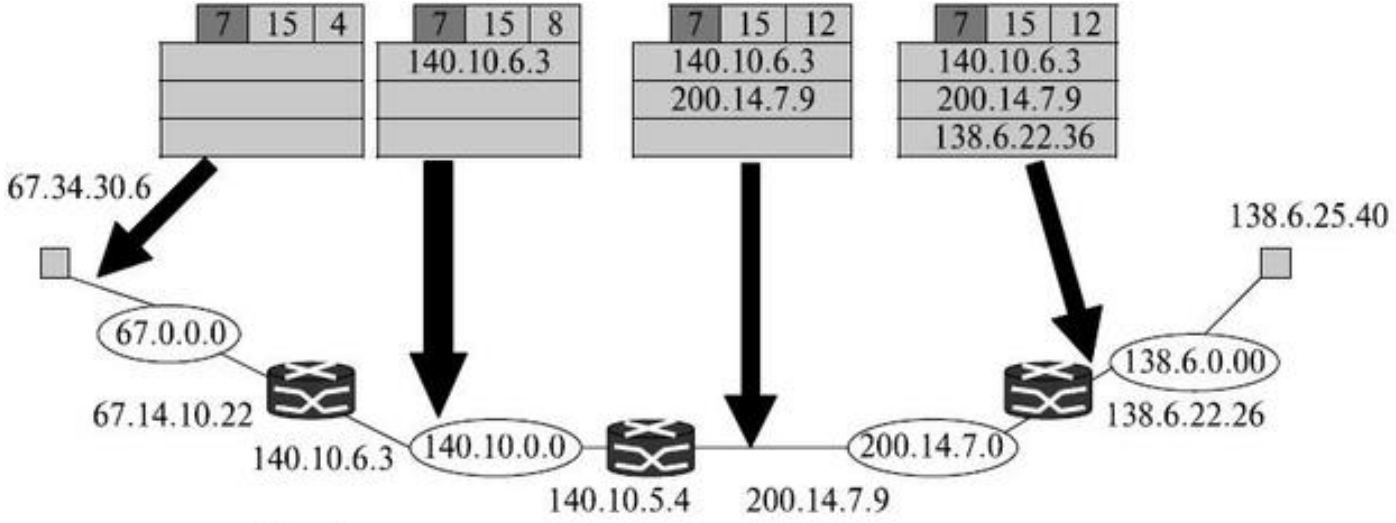
4.6.4. Опция нет оператора (No Operation - NOP) — однобайтовая опция имеющая номер 1, используемая как наполнитель между опциями. Например, она может использоваться для того, чтобы выровнять следующую к границе опцию на 32 бита или 16 бит.

4.6.5. Опция свободная маршрутизация (Loose Source and Record Route – LSRR) имеет номер опции 3 и разрешает маршрутизатору использовать любой из альтернативных маршрутов для достижения очередного адреса назначения из данных маршрута. Если адрес назначения (Destination Address, DA) не достигнут и значение указателя не больше значения длины, то в поле адреса назначения записывается очередной адрес из данных маршрута, в текущее поле данных маршрута – собственный адрес (SA) достигнутой сети, а значение указателя увеличивается на 4. Если значение указателя больше значения длины, то маршрутизация производится по адресу назначения без использования записи маршрута.

4.6.6. Опция метки времени (Time Stamp - TS) имеет номер 4 и используется, чтобы сделать запись времени обработки дейтаграммы маршрутизатором. Время выражено в миллисекундах, отсчет ведется с полуночи – так называемое Универсальное время.

Обработка значения времени дейтаграммы может помочь пользователям и менеджерам проследить поведение маршрутизаторов в Интернете. Можно оценить время, которое требуется для дейтаграммы, чтобы пройти от одного маршрутизатора к другому.

4.6.7. Опция запись полного маршрута (Record Route - RR) имеет номер опции 7 и предписывает маршрутизатору записывать все IP-адреса, через которые проходит маршрут следования пакета, см рис.



4.6.8. Опция строгая маршрутизация от источника (Strict Source and Record Route – SSRR) имеет номер 9 и предписывает маршрутизатору использовать только смежные маршрутизаторы для достижения очередного адреса из данных маршрута. Она применяется источником, чтобы точно предопределить маршрут для дейтаграммы, как он проходит через Интернет.

Если дейтаграмма задает строгую маршрутизацию от источника, все маршрутизаторы, определенные в опции и только они, должны быть пройдены дейтаграммой. Маршрутизатор не должен быть пройден, если его адрес IP не содержится в списке дейтаграммы. Если дейтаграмма проходит маршрутизатор, который не находится в списке, дейтаграмма отклоняется и создается сообщение об ошибках. Если дейтаграмма достигает пункта назначения и некоторые из адресов не были использованы, она будет также отклонена и создано сообщение об ошибках.

Задание маршрута источником может быть полезно с точки зрения нескольких целей. Отправитель может выбрать маршрут с определенным типом обслуживания, например, с минимальной задержкой или максимальной производительностью. Альтернативно, он может выбрать маршрут, который более безопасен или более достоверен для целей отправителя — например так, чтобы дейтаграмма не проходила через сеть конкурента.

Формат подобен опции "записи маршрута", за исключением того, что все адреса IP введены отправителем.

4.6.10. Пример разбора дейтаграммы.

Какие опции из шести вышеописанных должны быть скопированы в каждый фрагмент и что это за опции?

Решение. Проанализируем первый (старший слева) бит кодов каждой опции и пять последних битов (номер опции).

- Код 000**00000:** конец опции; не копируется.
- Код 000**00001:** нет операции; не копируется.
- Код 100**00011:** свободная маршрутизация от источника; копируется в каждый фрагмент.
- Код 010**00100:** метка времени; не копируется.
- Код 000**00111:** запись маршрута; не копируется.
- Код 100**01001:** строгая маршрутизация от источника; копируется в каждый фрагмент.

5. Блок-схема управления IP на узле и алгоритмы работы модулей IP.

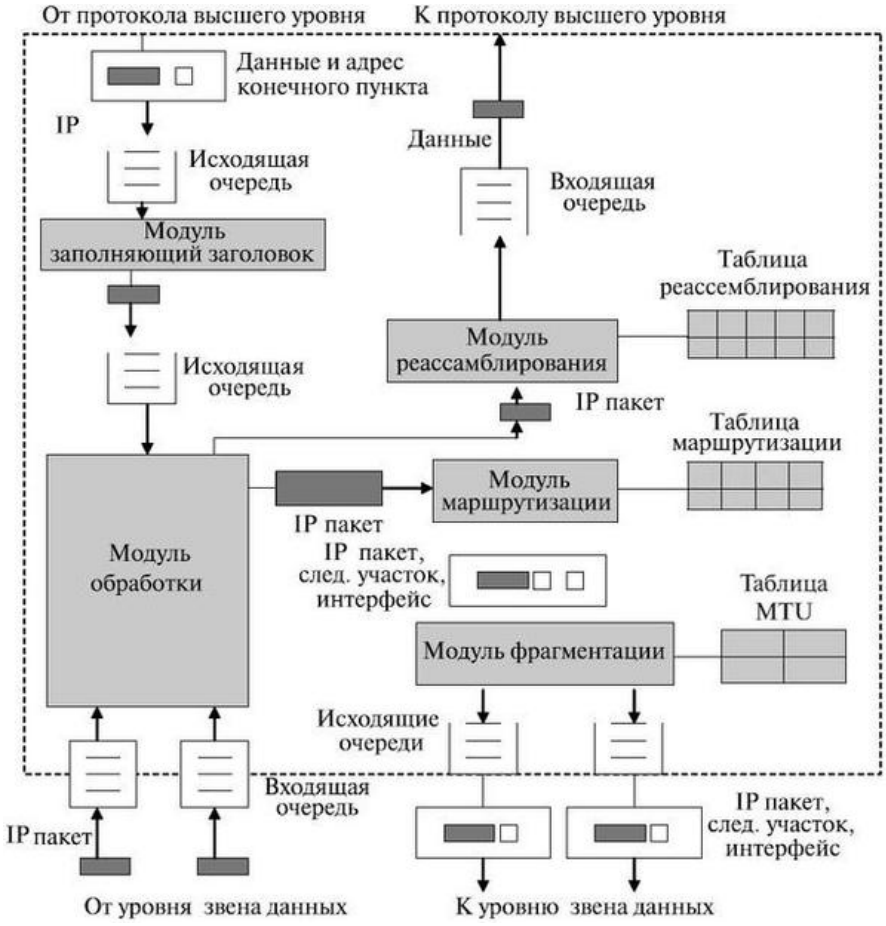


Рис. Алгоритм работы модуля обработки

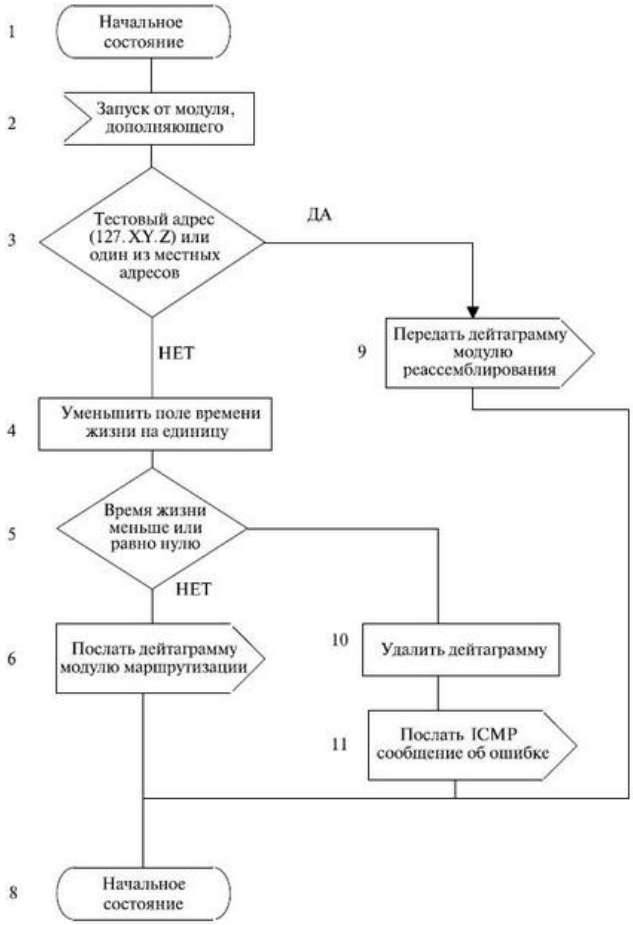
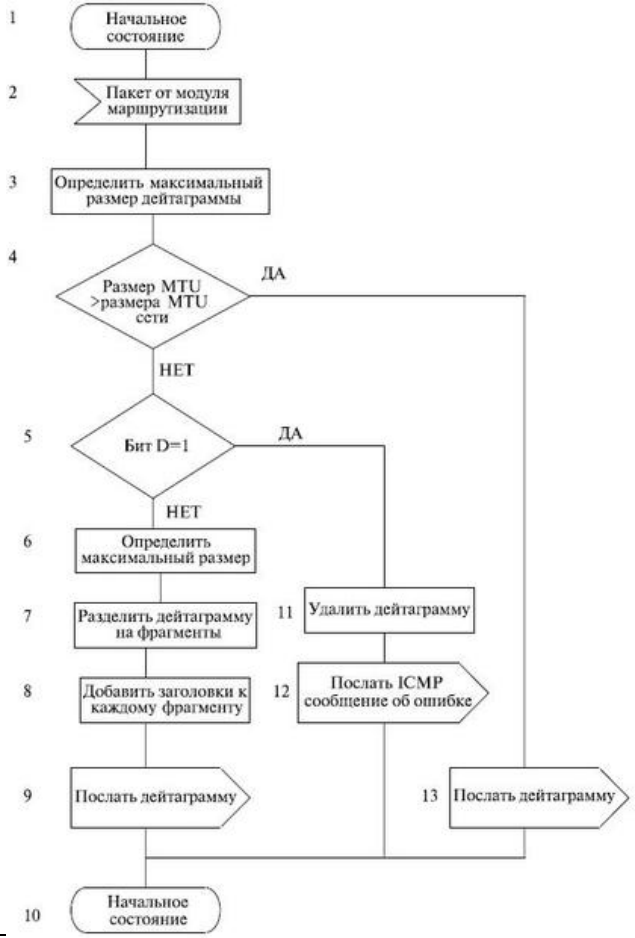


Рис. Алгоритм работы модуля фрагментации



6. Упражнения.

1. Какое поле IP-заголовка меняется от маршрутизатора к маршрутизатору?
2. Вычислите значение HLEN, если общая длина равна 1200 байт, 1176 из которых — данные высокого уровня HLEN.
3. В пункте 4.4. перечислены различные MTU в диапазоне от 296 до 65535. Какие преимущества имеет большой MTU? Какие преимущества имеет малый MTU?
4. Какое максимальное число маршрутизаторов, которое может быть записано, если опция метка времени имеет значение 1? Почему?
5. Значение общей длины поля IP-дейтаграммы равно 36, а значение длины заголовка равно 5. Сколько бит данных переносит такой пакет?
6. IP-дейтаграмма прибывает со значением смещения 0, и MF=0 (старший бит фрагментации). Существует ли этот фрагмент?
7. IP-дейтаграмма прибыла со следующей информацией в заголовке (hex-представление):

   45 00 00 54 00 03 00 00 20 06 00 00 7C 4E 03 02 B4 0E 0F 02

   a. Какие опции имеются в этой дейтаграмме?
   b. Фрагментирован ли пакет?
   c. Каков объем данных?
   d. Используется ли контрольная сумма?
   e. Сколько маршрутизаторов прошел пакет?
   f. Каков идентификационный номер пакета?
   g. Каков тип обслуживания?