Тема 11. Нижние уровни стека TCP/IP

- Взаимодействие сетей IP с сетями других технологий
- Инкапсуляция IP-пакетов в кадры Ethernet, Token Ring и FDDI
- Протокол последовательного канала SLIP
- Протокол PPP
- Базовый формат кадра PPP
- Протокол LCP
- Аутентификация по протоколам PAP и CHAP
- Протоколы NCP и LQM

Задачи протокола IP при взаимодействии с протоколами нижних уровней

1. Правила инкапсуляции IP-пакета в кадры
   - В какой тип кадра, если их несколько (Ethernet)
   - Как заполняются служебные поля кадра
2. Правила работы ARP
   - Broadcast - просто
   - Non-Broadcast with Multiple Access – как?
   - Point-to-point – нужен?
3. Работа с логическими подсетями (VLAN, FR, ATM)

Три типа сетей нижнего уровня

1. Широковещательные (broadcast)

   Локальные сети: Ethernet, Token Ring, FDDI

2. Точка-точка (Point-to-Point)

   PPP, HDLC

3. Нешироковещательные сети с множественным доступом

   X.25, frame relay, ATM – сети с предварительным образованием виртуальных каналов

1. Широковещательные (broadcast)

   Локальные сети: Ethernet, Token Ring, FDDI

Спецификации инкапсуляции IP-пакетов в кадры Ethernet, Token Ring и FDDI

1. *Инкапсуляция пакетов IP в кадры Ethernet*

⇒**Ethernet DIX (Ethernet II)**

```
ETHERNET: Destination address : 484C00054699
ETHERNET: Source address : 008048EB814C
ETHERNET: Ethernet Type : 0x0800 (IP:  DOD Internet Protocol)
  IP: ID = 0x654E; Proto = TCP; Len: 40
```

⇒**Ethernet LLC/SNAP**

```
   ETHERNET: Destination address : 01000CCCCCCC
   ETHERNET: Source address : 00E0F77F1920
   ETHERNET: Frame Length : 303 (0x012F)

LLC: DSAP = 0xAA : INDIVIDUAL : Sub-Network Access Protocol (SNAP)
LLC: SSAP = 0xAA: COMMAND : Sub-Network Access Protocol (SNAP)
LLC: Frame Category: Unnumbered Frame
LLC: Command = UI
LLC: LLC Data: Number of data bytes remaining = 286 (0x011E)
 SNAP: ETYPE = 0x08000 (IP:  DOD Internet Protocol)
 IP: ID = 0x654E; Proto = TCP; Len: 40
```

2. *Инкапсуляция пакетов IP в кадры Token Ring и FDDI -* *формат* ***LLC/SNAP***

2. Протоколы точка-точка (Point-to-Point)

   PPP, HDLC

Протоколы "точка-точка"

Протокол SLIP (Serial Line IP) - очень простая схема кадрирования, невысокая надежность

Протокол HDLC (High-level Data Link Control)

- Сложный протокол, работающий на основе алгоритмов установления соединения и скользящего окна
- Он использует 12 различных типов кадров и обеспечивает снижение вероятности искажения бита с 10^{-3} до 10^{-9}.
- Обеспечивается управление потоком данных за счет механизма окна и специальных кадров, приостанавливающих на время передачу данных от источника
- Протокол рассчитан на полнодуплексные соединения
- В семейство HDLC входят протоколы LAP-B, LAP-D и LAP-M, используемые в сетях X.25, ISDN и в модемах соответственно

Протокол PPP (Point-to-Point Protocol) - разработан группой Internet Engineering Task Forse взамен устаревшего протокола SLIP

- Протокол PPP стал фактическим стандартом для глобальных линий связи на выделенных каналах корпоративной сети
- Протокол PPP первоначально использовал формат кадров HDLC и дополнил их собственными полями - поля протокола PPP вложены в поле данных кадра HDLC

Протокол LAP-B:

Ст рукт ура кадра LAP-B

N(S), N(R) -

Типы кадров LAP-B

(1) *Информационный* - данные X.25

Супервизорные

(2) Готовность к приему

(3) Неготовность к приему

(4) Отказ

(5) Селективный отказ

Ненумерованные

(6) Установить режим нормальных ответов

(7) Установить режим асинхронных ответов

(8) Разъединить

(9) Запрос передачи

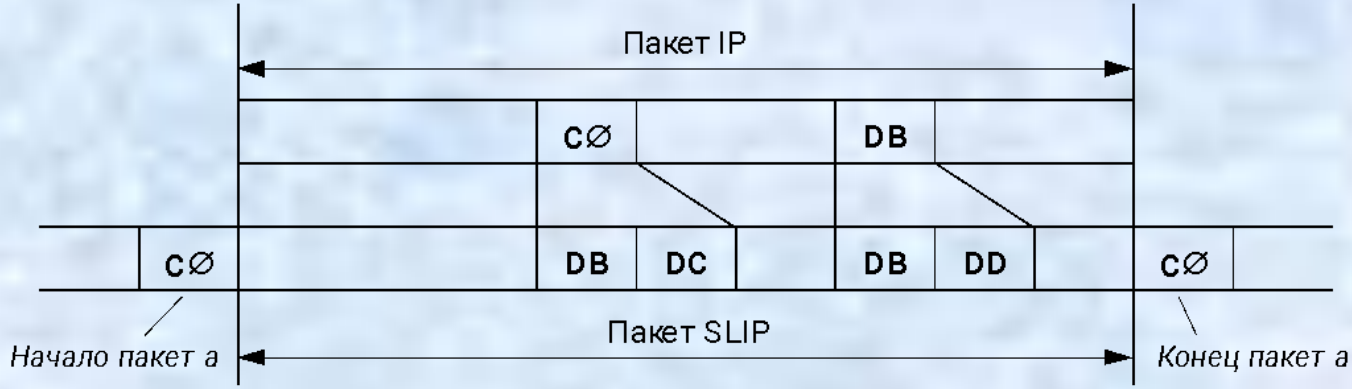
(10)Сброс

(11)Отказ от кадра

(12)Подтверждение

Протокол последовательного канала SLIP

- SLIP (Serial Line IP) - позволяет устройствам, соединенным последовательной линией IP связи, работать по протоколам TCP/IP
- В 1984 г. был встроен в операционную систему 4.2 Berkley Unix
- Дает возможность подключаться к сети Internet посредством стандартного порта RS232
- Выполняет работу по выделению из последовательности передаваемых по последовательному каналу бит границ пакета IP
- Большинство реализаций протокола SLIP поддерживают спецификацию Compressed SLIP (CSLIP)



Ограничения протокола SLIP

- размер инкапсулируемого пакета IP не должен превышать 1006 байтов
- нет механизмов, дающих возможность обмениваться адресной информацией. Это ограничение не позволяет использовать SLIP для некоторых видов сетевого сервиса
- можно передавать трафик лишь одного сетевого протокола
- не предусмотрены процедуры обнаружения и коррекции ошибок

Протокол PPP (Point-to-Point Protocol)

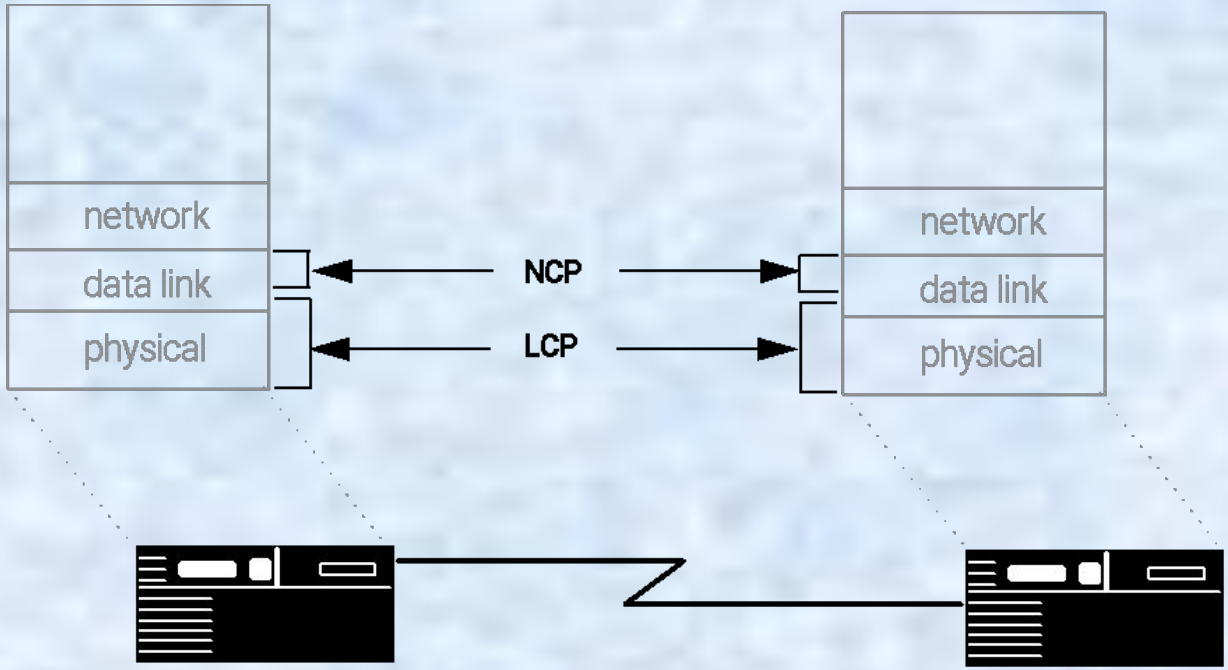
Основное назначение - организация одновременной передачи по одному логическому каналу "точка-точка" нескольких протоколов сетевого уровня

Поддерживаются протоколы:

- ⇒ AppleTalk
- ⇒ DECnet phase IV
- ⇒ IPX
- ⇒ IP
- ⇒ OSI
- ⇒ XNS
- ⇒ Banyan VINES
- ⇒ NetBEUI (проект стандарта)

PPP может использоваться для инкапсуляции других протоколов канального уровня, например, Ethernet, для их обработки по алгоритму моста

Компоненты протокола PPP

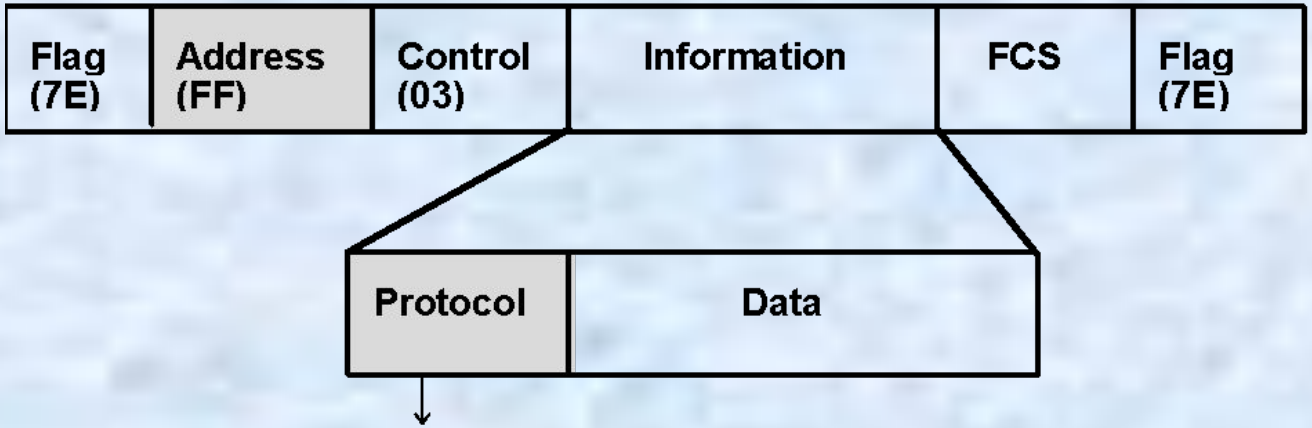


- При установлении сессии сначала работает протокол LCP,
  а затем протокол NCP
- Протокол LCP организует переговорный процесс о параметрах канала, например, о максимальном размере кадра. Протокол LCP также и завершает PPP-соединение между узлами

- Протокол NCP позволяет договориться о том:
  - ⇒ какие сетевые протоколы будут передаваться в данной сессии PPP
  - ⇒ каковы из параметры, например, IP-адрес клиента
- LCP после открытия сессии работает все время в фоновом режиме до завершения связи
- При организации сессии с помощью протокола LCP стороны могут договориться об использовании некоторых необязательных протоколов:
  - ⇒ протокола Link Quality Monitoring (LQM), который используется для слежения за качеством канала связи
  - ⇒ протокола аутентификации Password Authentication Protocol (PAP)
  - ⇒ протокола аутентификации Challenge Handshake Authentication Protocol (CHAP)

Базовый формат кадра PPP

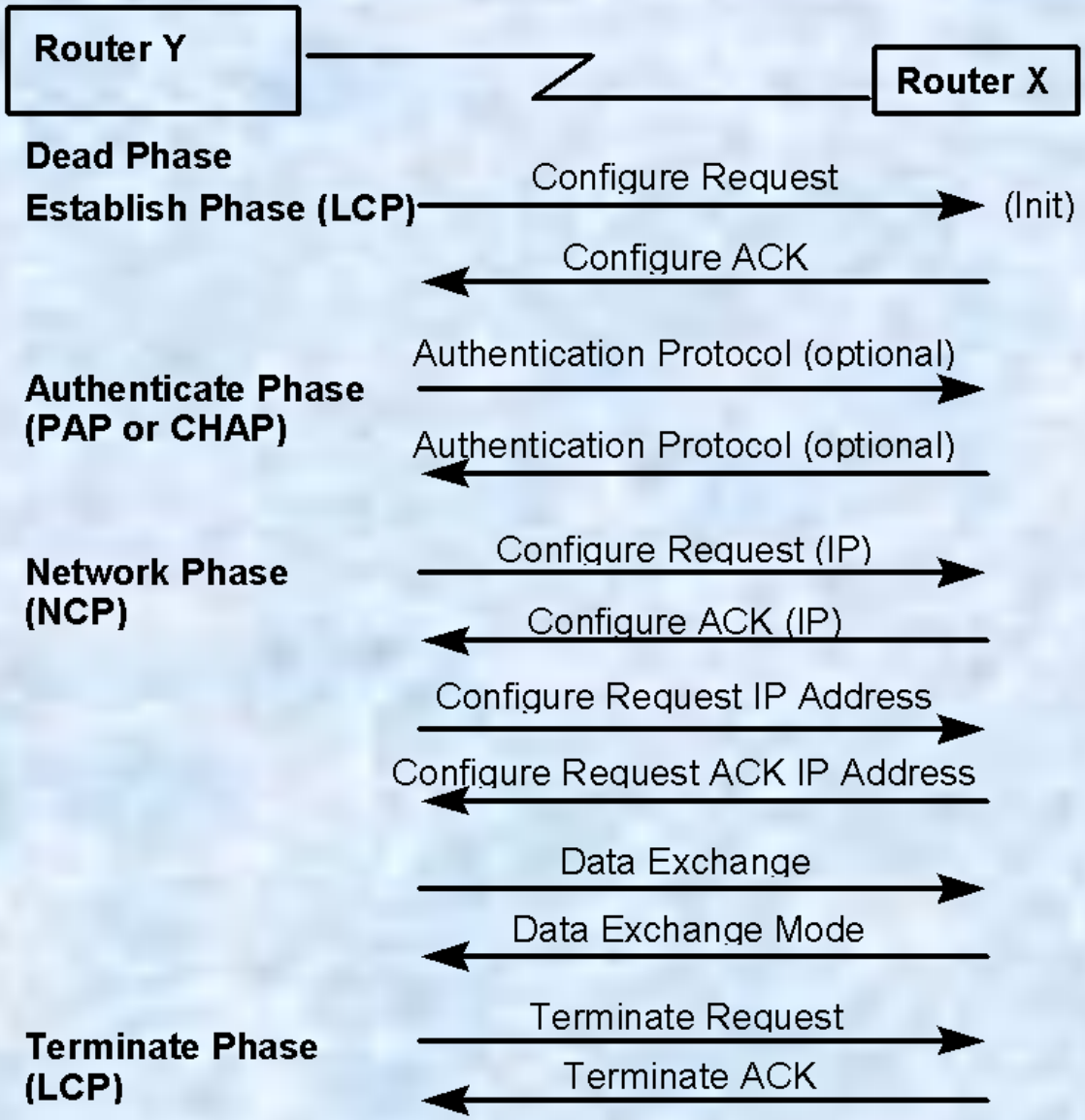
RFC1549 "PPP in HDLC Framing"



Указывает тип передаваемых данных : LCP, NCP, или протокол сетевого уровня:

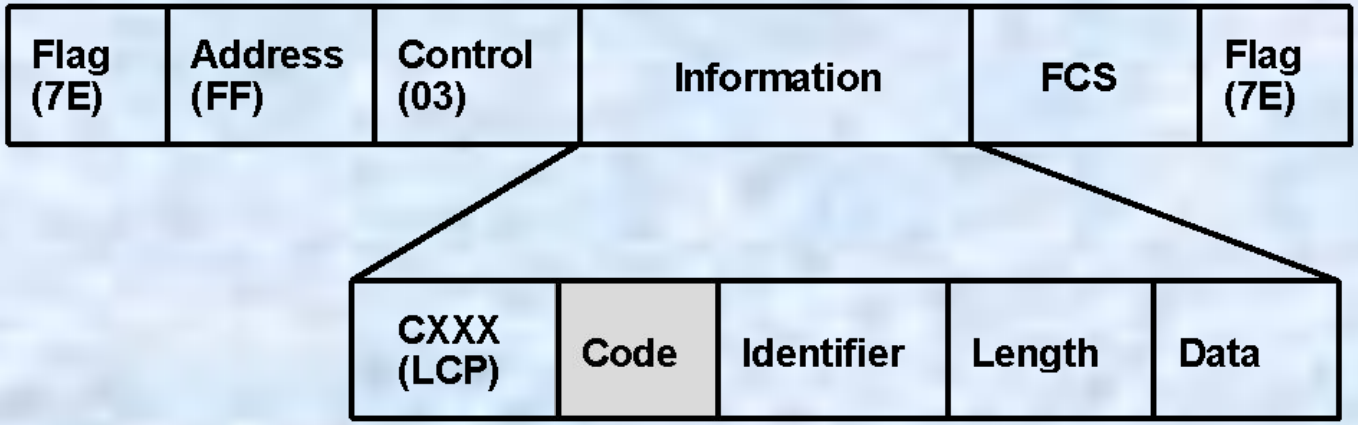
Cxxx = LCP
8xxx = NCP
0xxx = Протокол сетевого уровня

Фазы работы протокола PPP



- **Dead Phase** - определяет физическую готовность канала. В случае успешной инициализации физического уровня канал переходит в Establish Phase
- **Establish Phase** - инициализирует LCP и определяет параметры канала связи. Когда оба взаимодействующих узла получают сообщение Configure ACK, канал считается открытым и переходит в необязательную фазу аутентификации Authenticate Phase
- **Authenticate Phase** (необязательная фаза) - в этой фазе аутентифицируются обе точки, используя протоколы Password Authentication Protocol (PAP) или Challenge Handshake Authentication Protocol (CHAP). Network Phase Канал не переходит в фазу икации до завершения успешной аутентиф
- **Network Phase** - открывает сессии передачи по каналу PPP любых из поддерживаемых протоколов сетевого уровня, используя соответствующий протокол NCP. После открытия сессии NCP, PPP канал начинает передавать пользовательские данные
- **Terminate Phase**—

Структура пакета протокола LCP



Указывает тип LCP сообщения:

Config Req (code=1)
Config Ack (code=2)
Config Nak (code=3)
Config Rej (code=4)

Длина LCP пакета (code, ID, и data)

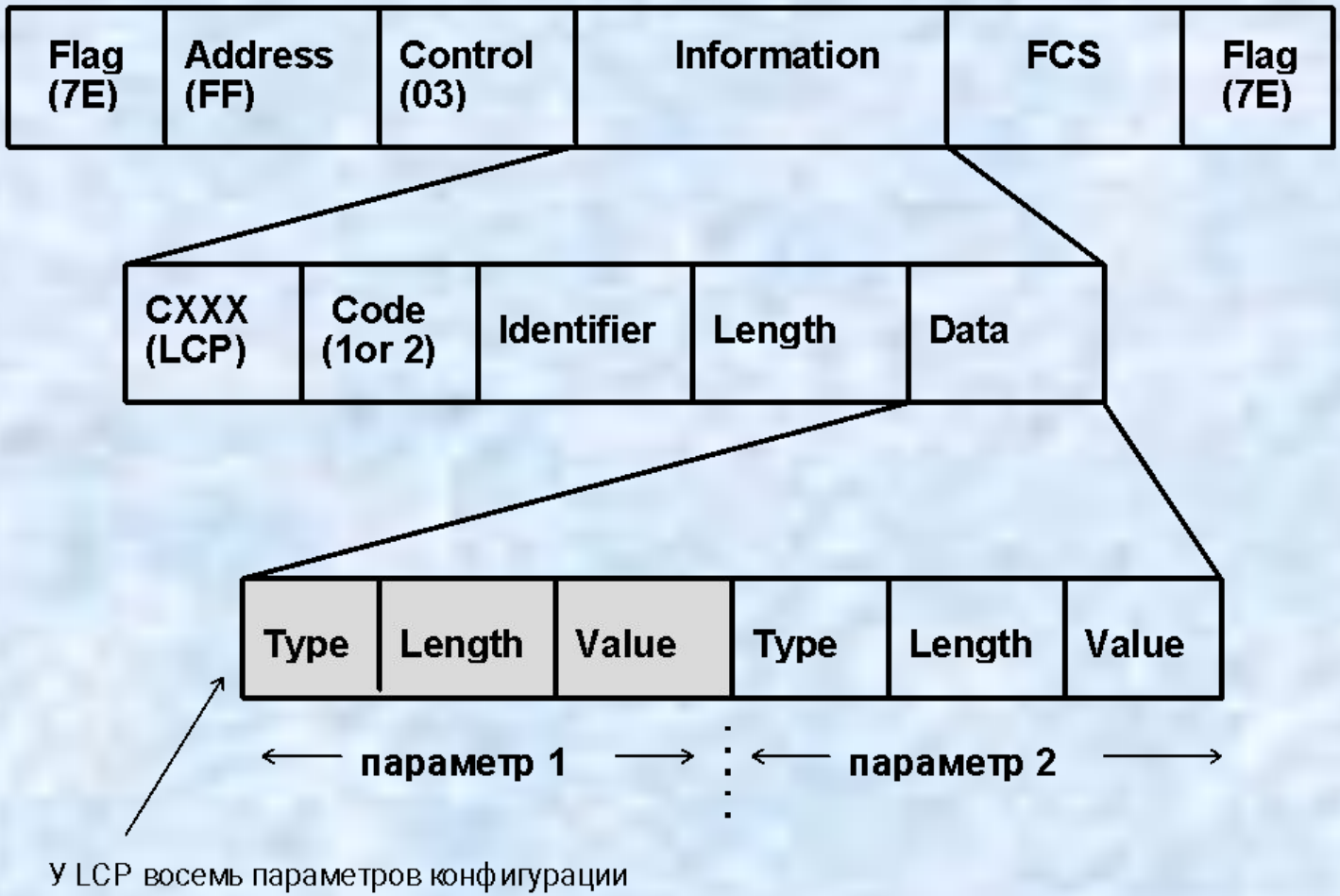
Ставит в соответствие LCP запросы и ответы

Примеры LCP пакетов, соответствующие различным кодам сообщений:

- Configure Request (code=1)
  - ⇒ Открытие соединения
  - ⇒ Обмен параметрами конфигурации
  - ⇒ Прием оговоренных параметров от другой стороны
- Configure Ack (code=2)
  - ⇒ Ответ на запрос Configure Request
  - ⇒ Указание на то, что значения параметров, полученных в Configure Request корректны
  - ⇒ Сигнализирует, что канал должен быть открыт по прибытию пакета
- Configure Nak (code=3)
  - ⇒ Показывает, что значения параметров неприемлемы
- Configure Reject (code=4)
  - ⇒ Указывает, что некоторые из параметров неприемлемы
  - ⇒ Шлет новый пакет Configure Request, не содержащий неприемлемые параметры, найденные в отвергнутом пакете

Поле Identifier идентифицирует LCP- запросы и LCP-ответы, помечая ответ на определенный запрос тем же идентификатором, что и запрос

Согласование параметров канала с помощью протокола LCP



Типы согласуемых по LCP параметров

Параметр	Описание	Тип	Длина	Значение
Maximum Receive Unit	Для согласования максимальных размеров пакетов (только для одного направления).	1	4	По умолчанию = 1500
Async-Control Character-Map	Согласует использование управляющих символов для асинхронных линий.	2	6	По умолчанию = FFFFFFFF
Authentication-Protocol	Используется для соглашения о используемом протоколе аутентификации. В некоторых реализациях допускается использование более одного протокола аутентификации.	3	>=4	C023 = Password Authentication Protocol C223 = Challenge/Response Authentication Protocol
Quality-Protocol	Определяет, когда и как часто в канале теряются данные	4	>=4	По умолчанию = нет C025 = Link Quality Report
Magic-Number	Для определения закольцованного канала и других аномалий на канальном уровне.	5	6	По умолчанию = нет
RESERVED				
Protocol-Field-Compressed	Согласует протокол сжатия на втором уровне.	7	2	По умолчанию = Запрещено
Address-and-Control Field-Compressed	Согласует сжатие полей address control (и имеющих фиксированные значения 03 и FF).	8	2	По умолчанию = Не сжимать
FCS-Alternatives	Согласует длину поля контрольной суммы (32 или 16 бит).	9	2	По умолчанию = 16-бит FCS

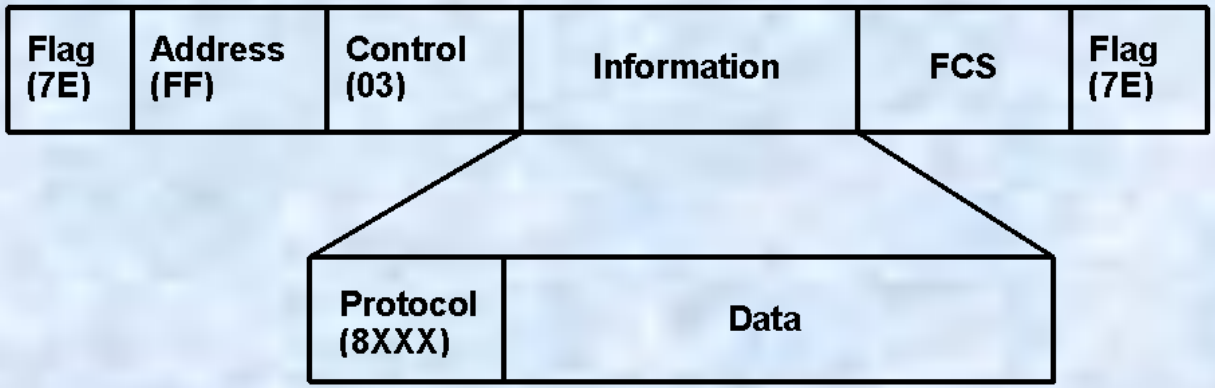
Пример диалога между двумя узлами по протоколу LCP:

```
DLC:  Frame 12 arrived at  11:59:42.4901; frame size
is 18 (0012 hex) bytes
PPP:
PPP:  Protocol = C021 (Link Control)
PPP:  Code = 01 (Configure Request)
PPP:  Identifier = 1
PPP:  Length = 14 bytes
PPP:  Type = 01 (Maximum Receive Unit)
PPP:  Length = 4 bytes
PPP:  Value = 1594 bytes
PPP:  Type = 05 (Magic Number)
PPP:  Length = 6 bytes
PPP:  Value = 129D7DC8
PPP:

DLC:  Frame 13 arrived at  11:59:42.5031; frame size
is 18 (0012 hex) bytes
PPP:
PPP:  Protocol = C021 (Link Control)
PPP:  Code = 02 (Configure Ack)
PPP:  Identifier = 1
PPP:  Length = 14 bytes
PPP:  Type = 01 (Maximum Receive Unit)
PPP:  Length = 4 bytes
PPP:  Value = 1594 bytes
PPP:  Type = 05 (Magic Number)
PPP:  Length = 6 bytes
PPP:  Value = 129D7DC8
PPP:
```

Протокол NCP

- Переговорный процесс, проводимый NCP, определяет, пакеты каких протоколов сетевого уровня будут передаваться в пакетах PPP в рамках данной сессии
- Для каждого протокола сетевого уровня, который поддерживается NCP, существует соответствующий стандарт RFC, определяющий способ инкапсуляции его пакетов в PPP пакет, а также параметры, подлежащие принятию в результате переговоров
- Например, для протокола IP существует спецификация RFC1332 "The PPP Internet Protocol Control Protocol (IPCP)"
- Аналогичные спецификации имеются для протоколов IPX (IPXCP), DECnet (DNCP) и других
- Отдельного протокола NCP не существует, а имеется семейство NCP-протоколов, по одному для каждого протокола сетевого уровня
- Протоколы семейства NCP используют тот же формат кадра, что и протокол LCP

Формат кадра протокола NCP:



Указывает на тип NCP:

8021 - IP
8029 - AT
8025 - XNS, VINES
8031 - Bridge
8027 - DECnet
8023 - OSI

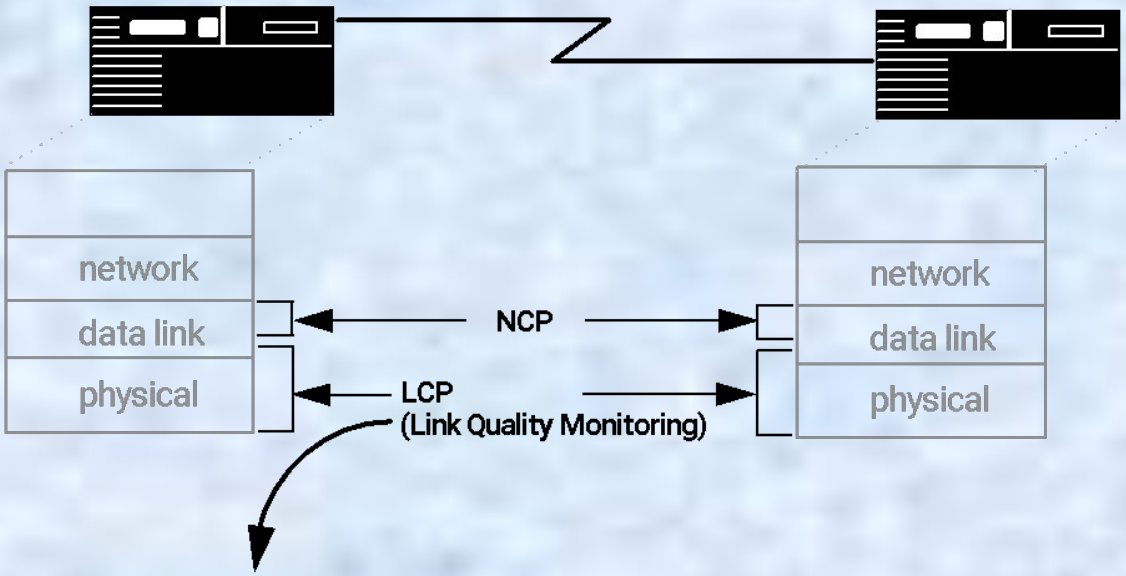
Пример процедуры согласования параметров протокола IP с помощью протокола IPCP

Согласуется IP-адрес

```
DLC:  Frame 34 arrived at  12:00:09.4456; frame size is 14
(000E hex) bytes
PPP:
PPP:  Protocol = 8021 (Internet Protocol Control)
PPP:  Code = 01 (Configure Request)
PPP:  Identifier = 9
PPP:  Length = 10 bytes
PPP:  Type = 03 (IP Address)
PPP:  Length = 6 bytes
PPP:  IP address = [200.5.5.1]
PPP:
```

```
DLC:  Frame 35 arrived at  12:00:09.4626; frame size is 14
(000E hex) bytes
PPP:
PPP:  Protocol = 8021 (Internet Protocol Control)
PPP:  Code = 02 (Configure Ack)
PPP:  Identifier = 9
PPP:  Length = 10 bytes
PPP:  Type = 03 (IP Address)
PPP:  Length = 6 bytes
PPP:  IP address = [200.5.5.1]
PPP:
```

Протокол Link Quality Monitoring (LQM)



Протокол LQM использует для слежения два типа процедур:

- Link Quality Report (LQR)— определяет качество линии связи, основываясь на проценте успешно переданных пакетов
- Echo Request/Reply - определяет качество линии с помощью передачи служебных пакетов

Протоколы аутентификации PAP и CHAP

- Описаны в RFC 1334
- В этом стандарте определено два протокола:
  - ⇒ Password Authentication Protocol (PAP)
  - ⇒ Challenge Handshake Authentication Protocol (CHAP)
- **Протокол PAP**
  - ⇒ Использует при аутентификации передачу идентификатора партнера и его пароля по глобальному каналу в виде открытого текста
  - ⇒ Если аутентификатор обнаруживает совпадение идентификатора и пароля с записью, имеющейся у него в базе легальных пользователей, то процесс аутентификации считается успешно завершенным

- **Протокол CHAP**

⇒ **Ключ** *(secret)* имеется как у аутентификатора, так и у партнера

⇒ **Слово-вызов** (*challenge*)
генерируется аутентификатором
и передается в виде пакета типа Challenge партнеру

⇒ Партнер, получив слово-вызов, зашифровывает его с помощью односторонней хэш-функции MD5

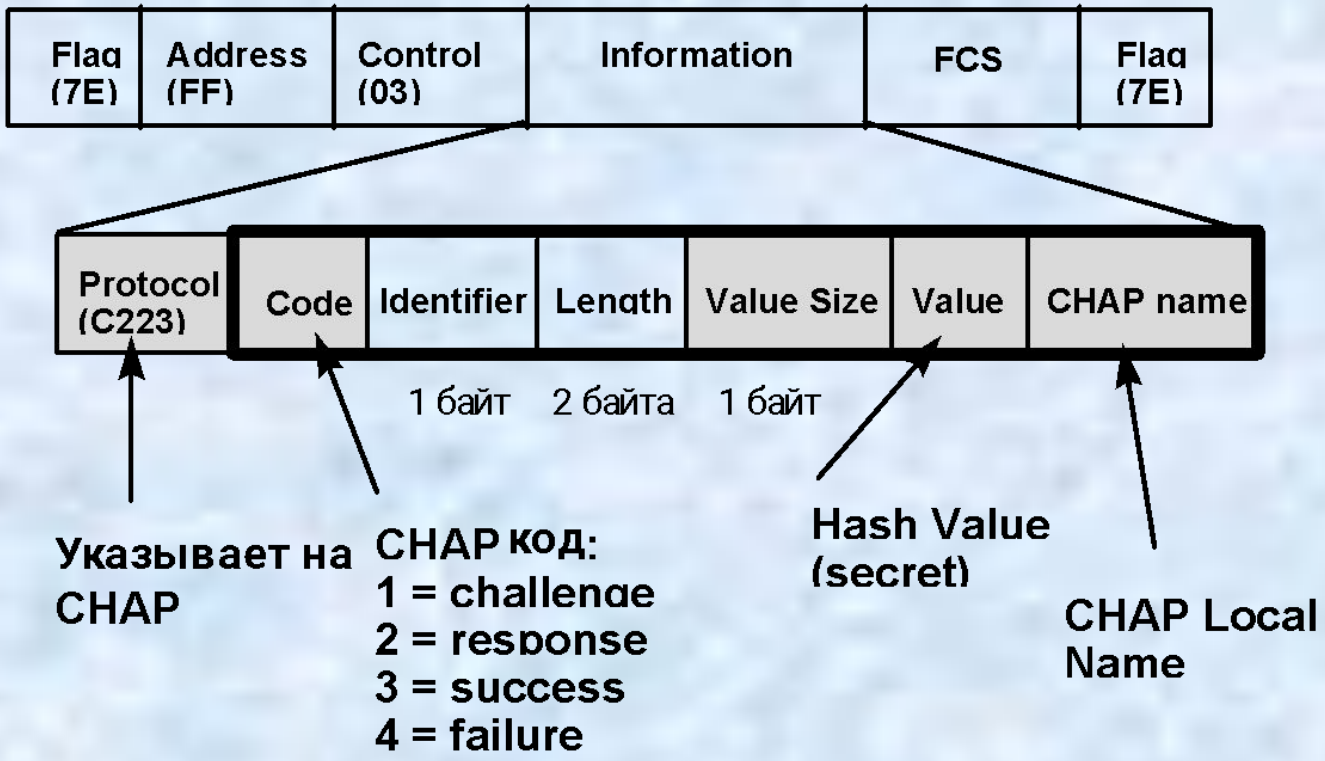
⇒ Результат работы хэш-функции возвращается аутентификатору в виде пакета типа Response

⇒ Аутентификатор сравнивает этот ответ с тем значением, которое он получил, локально применив хэш-функцию к слову-вызову

⇒ Если результаты совпадают, то аутентификация считается успешной и партнеру посылается пакет типа Success - успех

⇒ Для защиты от перехвата ответа аутентификатор должен использовать различные значения последовательности символов при каждой последовательной аутентификации

Формат пакетов протокола CHAP



- Поле **Code** определяет тип пакета
- Поле **Identifier** необходимо для отождествления ответов (Response) и вызовов (Challenge)
- Поле **Length** содержит общую длину пакета CHAP (Code Identifier) вместе с полями и
- Поле **Value Size** определяет длину слова-вызова (в пакетах Challenge) или хэш-значения (в пакетах Response)
- Поле **Value** предназначено для передачи слова-вызова или хэш-значения

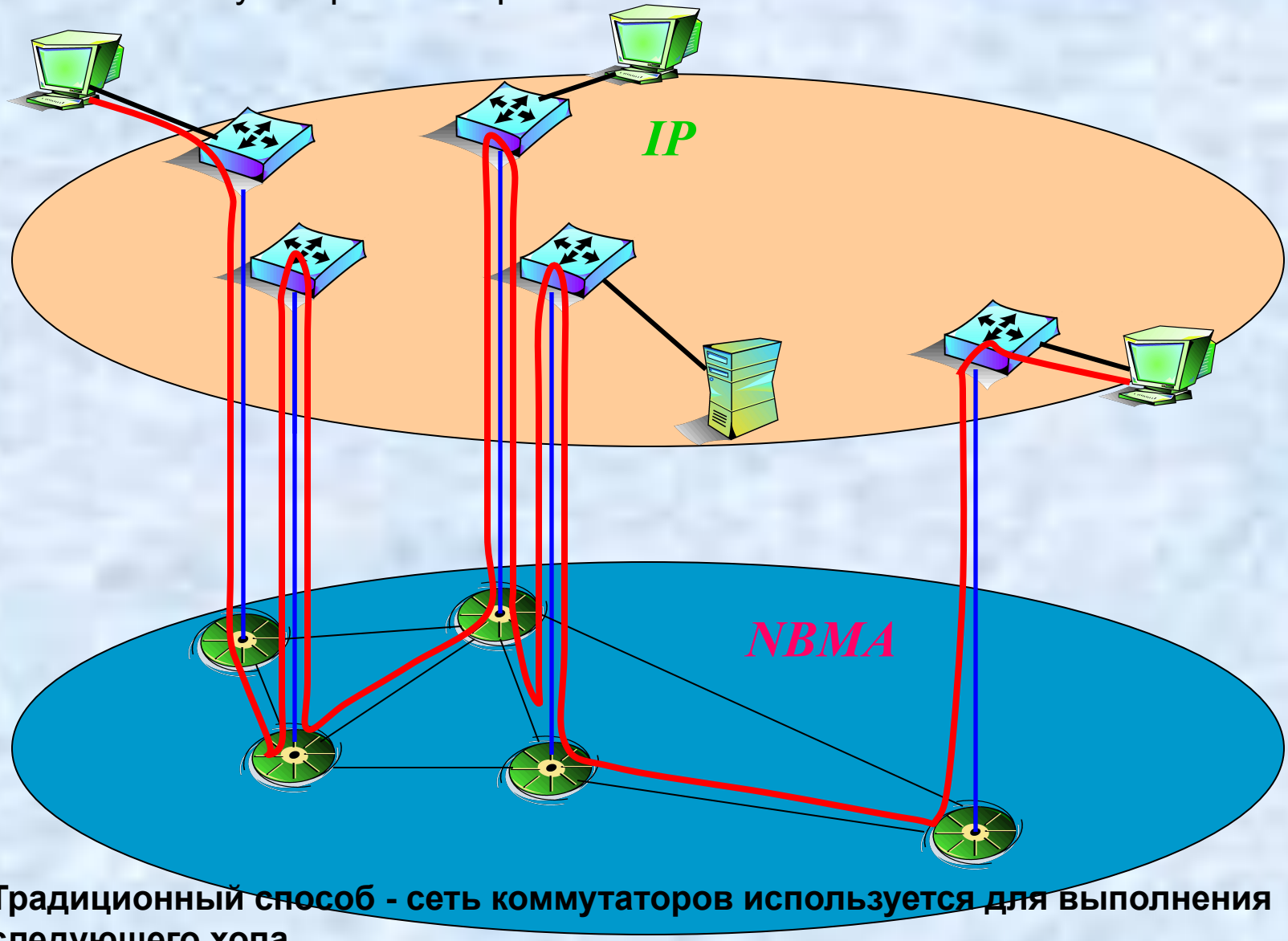
- Поле **CHAP name** - имя аутентификатора или партнера, в зависимости от того, кто является отправителем пакета. Имя партнера нужно аутентификатору для того, чтобы знать, какой ключ нужно использовать при аутентификации

Пример аутентификации узла с именем chicago аутентификатора с именем paris:

```
----------------Frame 24 (Challenge)-------------
DLC:  Frame 24 arrived at  12:50:02.1940; frame size is 18 (0012 hex) bytes
PPP:  Protocol = C223 (Challange Handshake Authentication)

ADDR  HEX          ID          |
0000  FF 03 C2 23 01 02 00 0E  04 10 6C 02 F7 70 61 72  □.B#.....l.wpar
0010  69 73                                            |  14  |          is


----------------Frame 25 (Response)-------------
DLC:  Frame 25 arrived at  12:50:02.2184; frame size is 32 (0020 hex) bytes
PPP:  Protocol = C223 (Challange Handshake Authentication)

ADDR  HEX          ID        |MD5   (
0000  FF 03 C2 23 02 02 00 1C  10 47 A4 0C 5D 45 4D EF  □.B#.....G$.]EMo
0010  5D 29 66 B2 13 17 1A F6  4B 63 68 69 63 61 67 6F  ])f2...vKchicago


----------------Frame 26 (Success)---------------
DLC:  Frame 26 arrived at  12:50:02.2257; frame size is 8 (0008 hex) bytes
PPP:  Protocol = C223 (Challange Handshake Authentication)

ADDR  HEX                                  ASCII
0000  FF 03 C2 23 03 02 00 04              □.B#...
```

3. Нешироковещательные сети с множественным доступом – Non-Broadcast with Multuple Access, NBMA

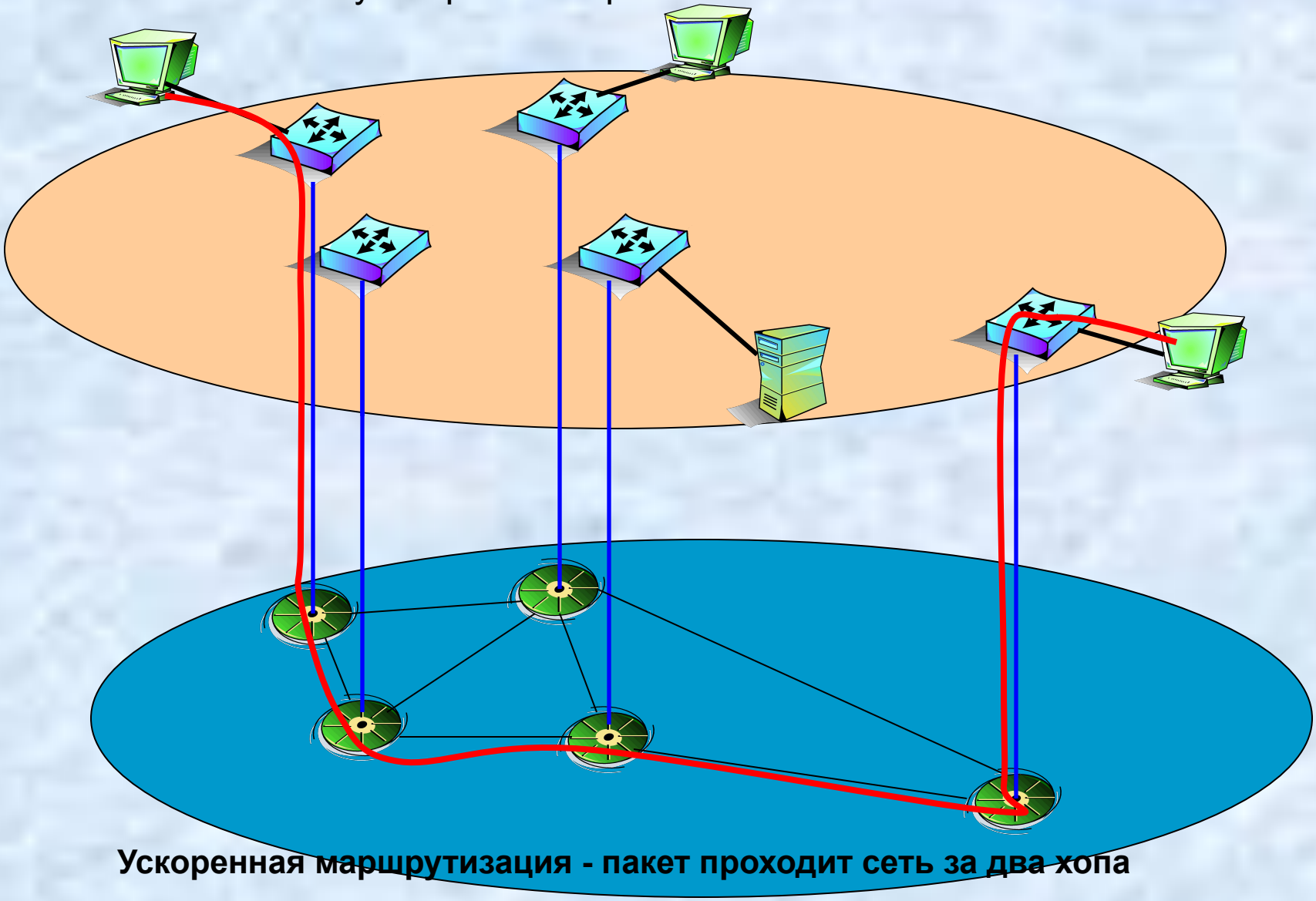
Взаимодействие слоев маршрутизаторов и коммутаторов в современных сетях



Традиционный способ - сеть коммутаторов используется для выполнения следующего хопа

Результат - медленное продвижение пакета - большое число хопов

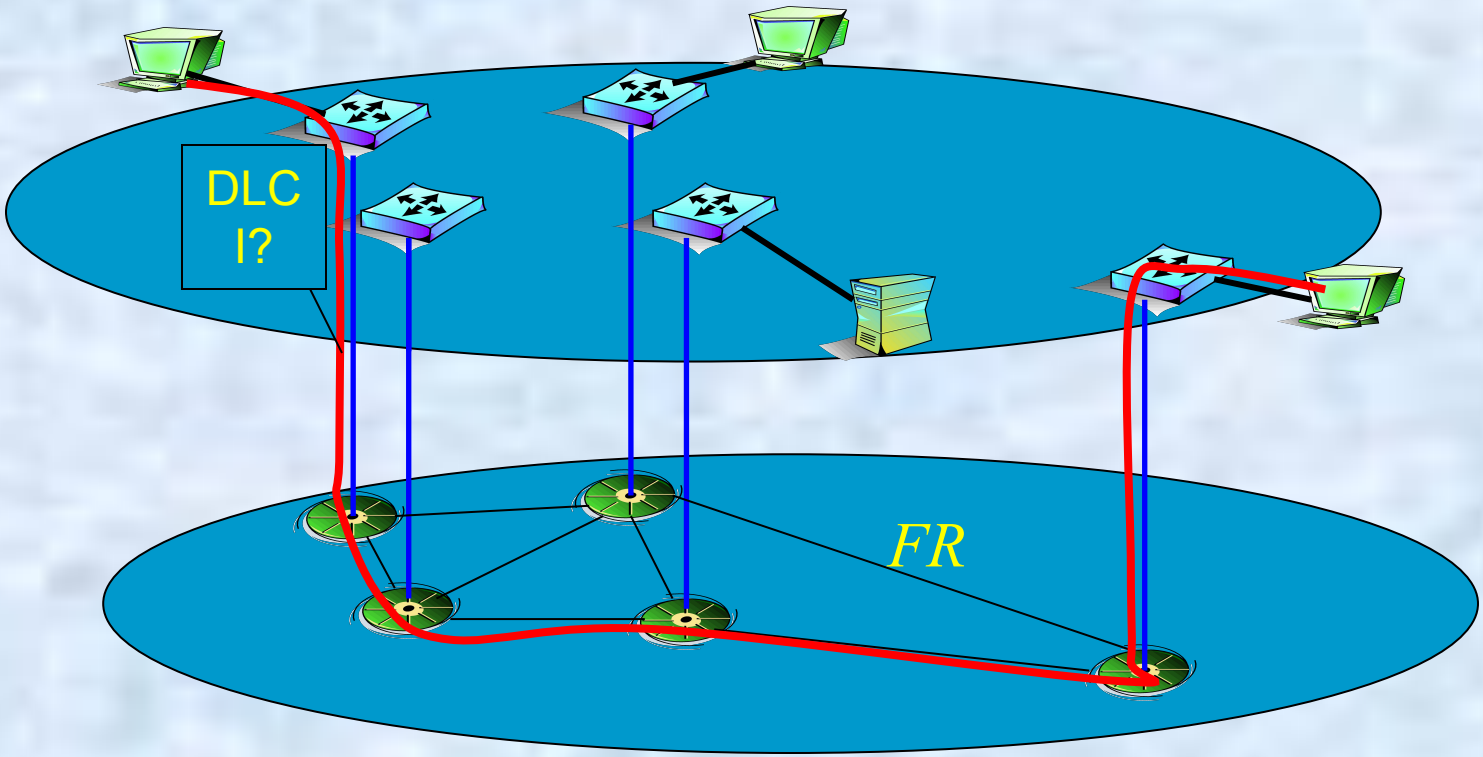
Взаимодействие слоев маршрутизаторов и коммутаторов в современных сетях



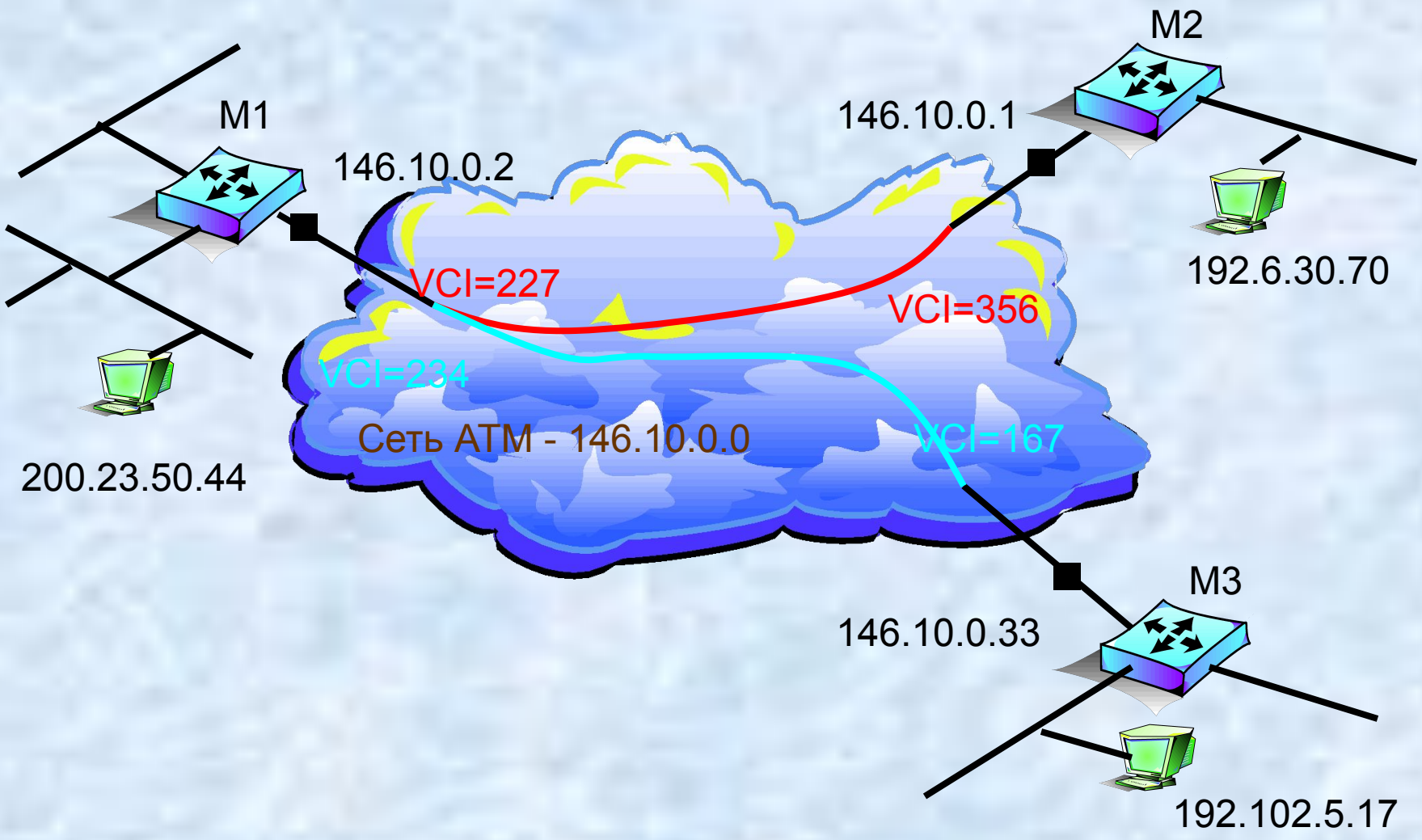
Ускоренная маршрутизация - пакет проходит сеть за два хопа

Происходит «прокол» сети коммутаторов до ближайшего к узлу назначения маршрутизатору

Основная проблема - как определить канальный адрес ближайшего маршрутизатора – как будет работать ARP ?



Традиционный вариант работы IP over NBMA



Ручное конфигурирование ARP-таблицы

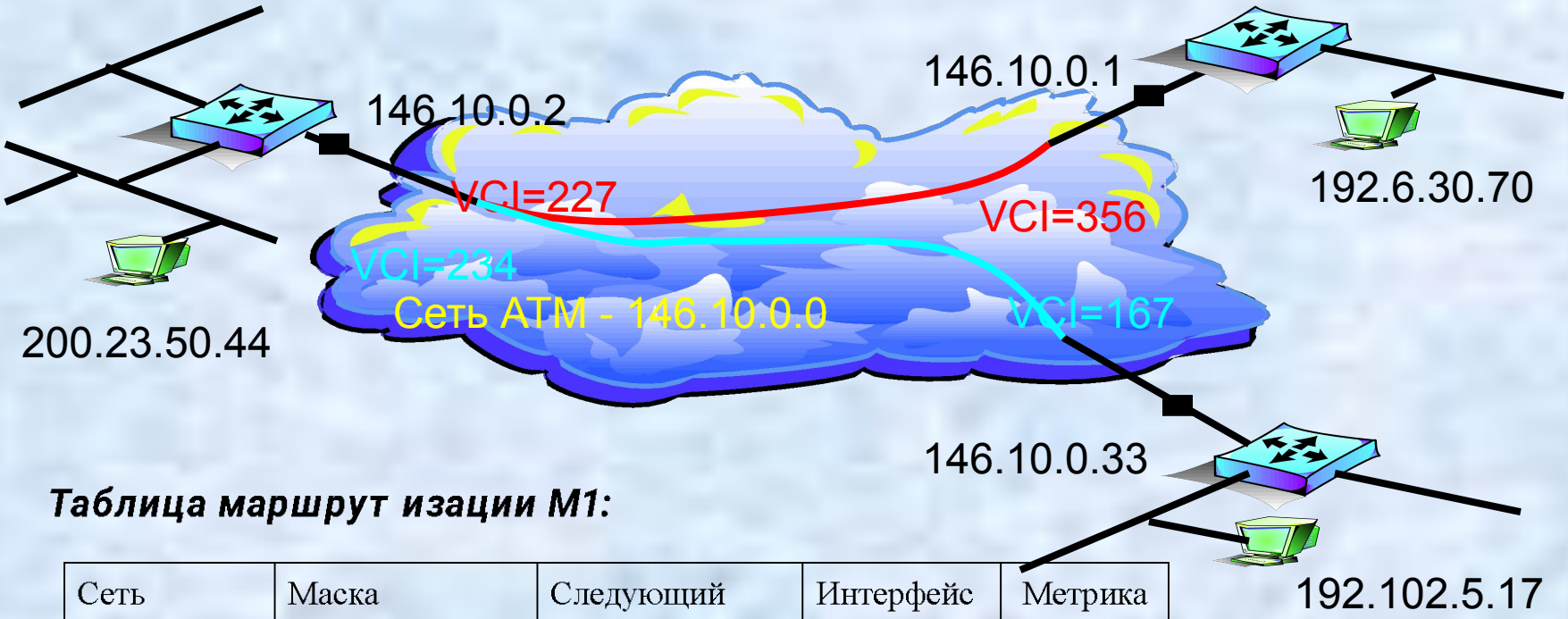


Таблица маршрут изации М1:

Сеть	Маска	Следующий маршрутизатор	Интерфейс	Метрика
146.10.0.0	255.255.0.0	146.10.0.2	146.10.0.2	1
200.23.50.0	255.255.255.0	146.10.0.2	146.10.0.2	1
192.6.30.0	255.255.255.0	146.10.0.1	146.10.0.2	2
195. 102.5.0	255.255.255.0	146.10.0.33	146.10.0.2	2

ARP-таблица маршрутизатора 1:

146.10.0.1	227
146.10.0.33	234

Специальные команды конфигурирования маршрутизаторов Cisco для сетей NBMA

`neighbor ip-address` – задает соседа по NBMA

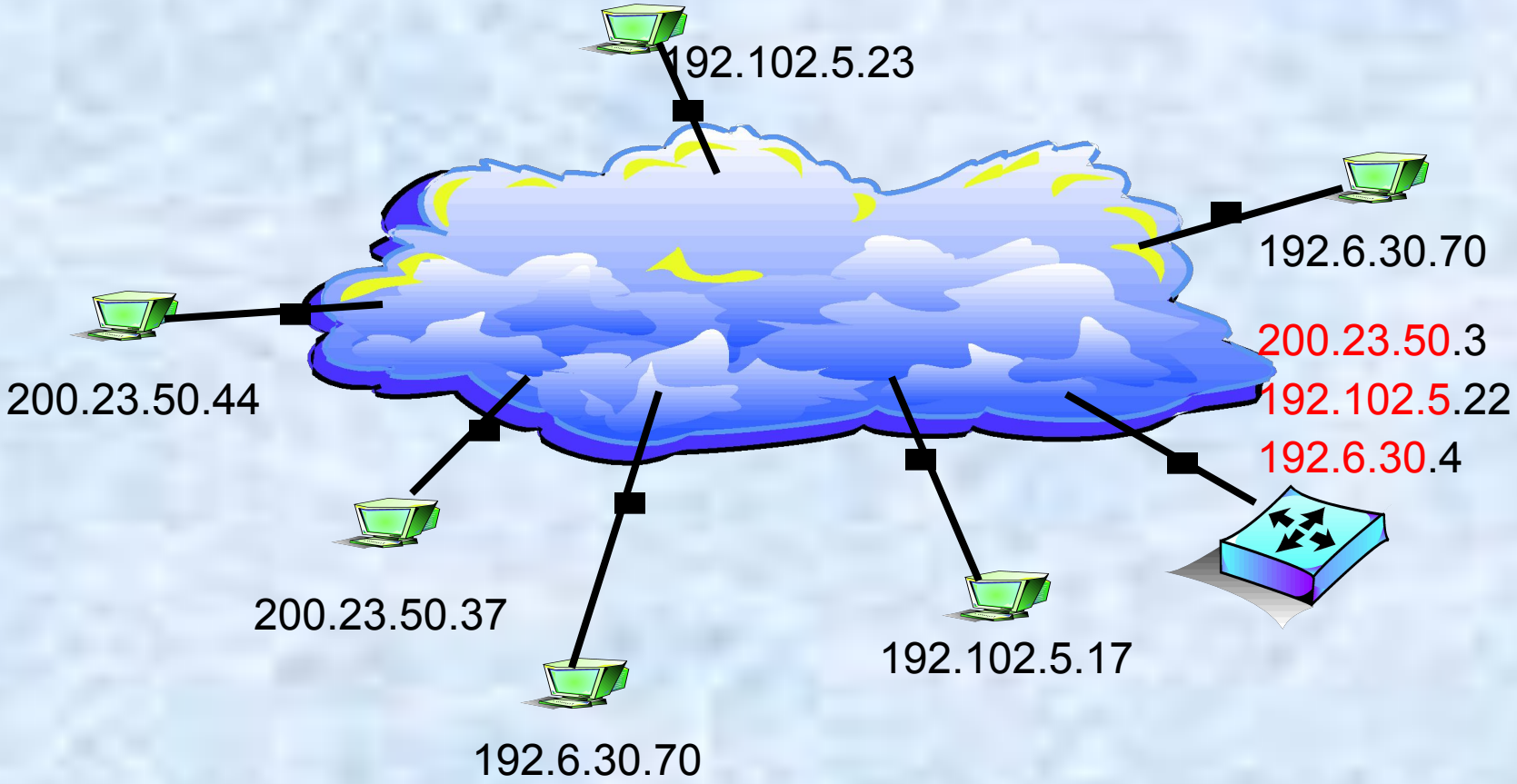
`frame-relay map ip 194.12.200.5 201`-

отображает IP-адрес 194.12.200.5 на номер PVC 201

`no ip split-horizon`

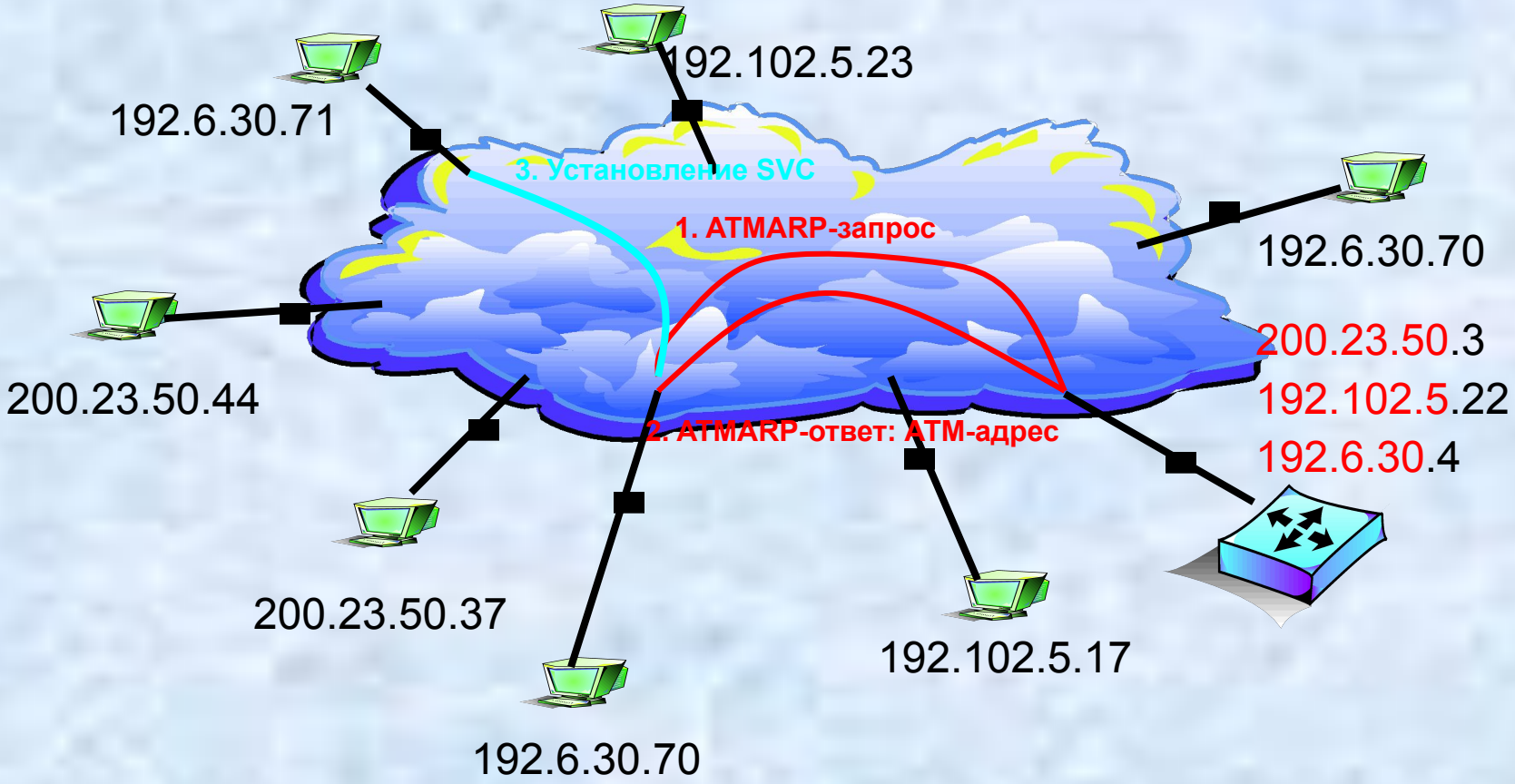
запрещает использование техники `split horizon` для интерфейсов с несколькими IP-адресами

Логические подсети - Logical IP Subnetwork



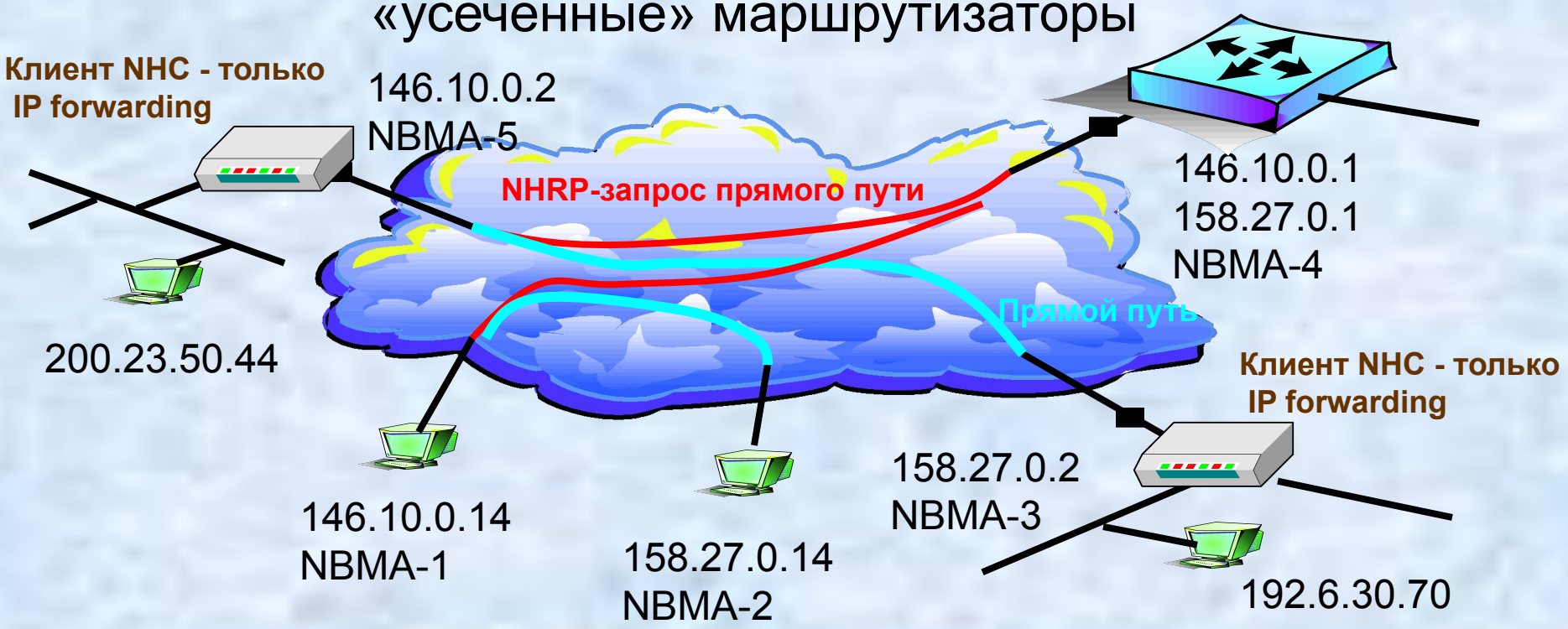
- Узлы различных IP-подсетей образуют логические IP подсети - LIS (Logical IP Subnetwork)
- Узлы различных LIS не могут взаимодействовать через сеть ATM непосредственно, только - через маршрутизатор
- Маршрутизатор присоединен ко всем LIS

Протокол Classical IP - поддержка LIS в ATM



- В каждой LIS должен быть свой ARP-сервер (обычно, это маршрутизатор)
- ATM-адрес ARP-сервера формируется в узлах вручную
- Каждый клиент должен установить соединение с ARP-сервером и зарегестрировать IP- и ATM-адреса
- Разрешение IP-адреса автоматическое, по стандартному ARP-запросу, направляемому ARP-серверу
- Взаимодействие между LIS – обычным способом, через маршрутизатор

NHRP - кратчайшая связь между LIS через «усеченные» маршрутизаторы



Нахождение прямого пути между сетями:

1. **Клиент NHC - серверу NHS: Запрос (без маршрутизации, по протоколу NBMA) на следующий хоп к узлу 192.6.30.70**
2. **Сервер NHS - клиенту NHC: Следующий хоп - адрес NBMA-3**
3. **Клиент устанавливает прямой путь к узлу NBMA-3 и передает ему пакет**
4. **Узел NBMA-3 – усеченный маршрутизатор. Он продвигает пакет узлу 192.6.30.70 обычным способом**