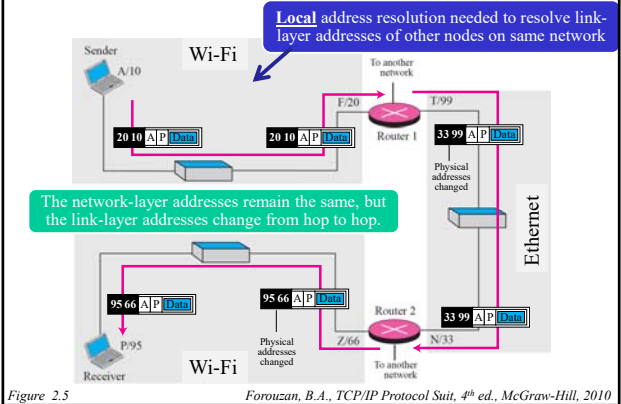


Съпоставяне на адреси (address resolution)

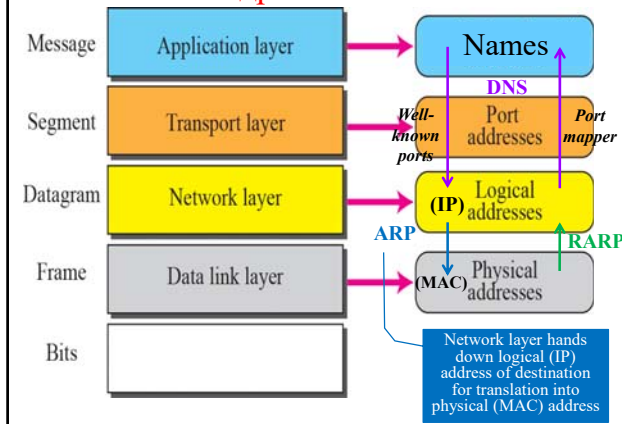
1

Необходимост от съпоставяне на адреси



2

TCP/IP: Адреси и съпоставяне



3

Съпоставяне на адреси: Техники

- **Справка в таблица (table lookup)**
 - Адреси на горния слой (HL), съхранявани в паметта заедно със съответните им адреси на долния слой (LL)
 - Долният слой използва HL адреса като ключ, за да намери съответния му LL адрес.
- **Изчисление (closed-form computation)**
 - HL адрес, базиран на LL адрес.
 - LL извлича LL адреса от HL адреса
- **Динамично съпоставяне (dynamic resolution)**
 - Обмен на съобщения, използвани за адресно съпоставяне, когато е необходимо.
 - HL изпраща заявка за LL адрес
 - Местоназначението (или друг възел) отговаря с необходимия LL адрес

4

Съпоставяне на адреси: Справка в таблица

- Използване на списък, съдържащ HL адрес (напр. IP адрес) и LL адрес (напр. MAC/хардуерен адрес) за всеки мрежов възел
- Претърсване по IP адрес и извличане на съответния хардуерен адрес
- Може да се спести място чрез отпадане на NetID префикса
 - Всички IP адреси в мрежата имат един и същ префикс
- Последователното търсене може да е прекалено бавно
- По-добре е да се използва индексирание или хеширане

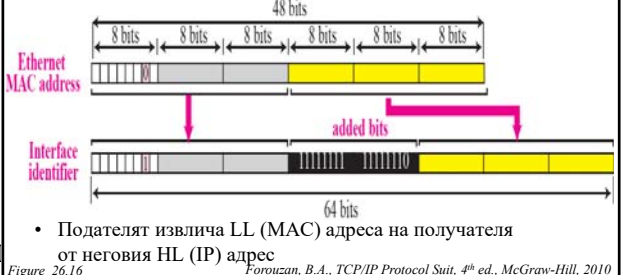
IP Address	Hardware Address
197.15.3.2	0A:07:4B:12:82:36
197.15.3.3	0A:9C:28:71:32:8D
197.15.3.4	0A:11:C3:68:01:99
197.15.3.5	0A:74:59:32:CC:1F
197.15.3.6	0A:04:BC:00:03:28
197.15.3.7	0A:77:81:0E:52:FA

ф. Иван

5

Съпоставяне на адреси: Изчисление

- HL адрес, базиран на LL адрес.
 - Т.е. IPv6 адрес, избран въз основа на MAC адрес (в LANs)
- Пример
 - MAC адресът е (F5-A9-23-14-7A-D2)₁₆
 - Съответният му IPv6 адрес е F7A9:23FF:FE14:7AD2



6

1

Съпоставяне на адреси: Динамично съпоставяне

- Обмен на съобщения с друг/и възел/възли за научаване на LL адреса (на получателя)
- 2 възможности:
 - Чрез сървър (*server-based*)
 - Хостът-подател изпраща съобщение до сървър с цел научаване на необходимия му адрес
 - Списък на съответните сървъри се поддържа в хостовете
 - Напр. *DNS* протокола в TCP/IP
 - Чрез равнопоставен възел (*peer-to-peer*)
 - Хостът-подател изпраща *broadcasts* съобщение-заявка към всички възли в мрежата
 - (Бъдещият) получател отговаря с неговия хардуерен адрес
 - Напр. *ARP* протокола в TCP/IP

7

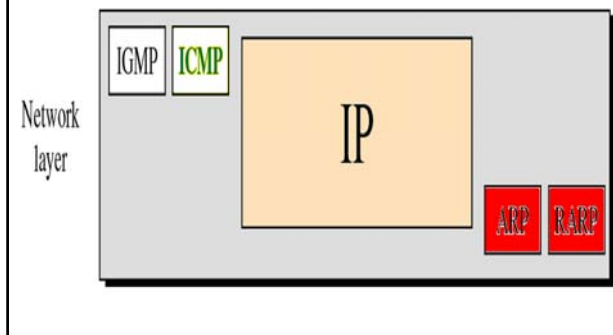
Table Lookup (T), Closed-Form Computation (C), Dynamic Resolution (D): Сравнение

Feature	Type Of Resolution
Useful with any hardware	T
Address change affects all hosts	T
Protocol address independent of hardware address	T, D
Hardware address must be smaller than protocol address	C
Protocol address determined by hardware address	C
Requires hardware broadcast	D
Adds traffic to a network	D
Produces resolution with minimum delay	T, C
Implementation is more difficult	D

Hybrid T+D used on Internet!

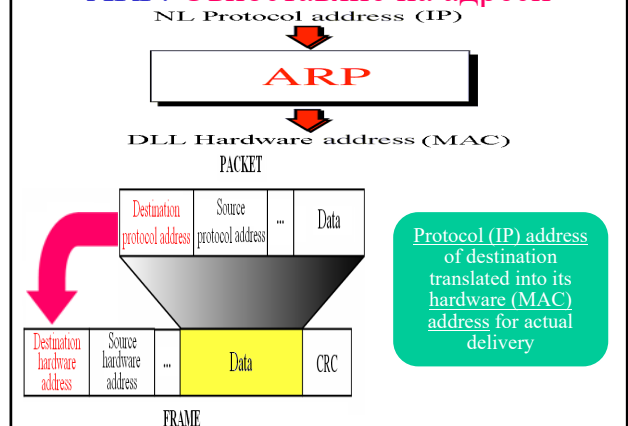
8

Протоколи (v4) на мрежовия слой за съпоставяне на адреси в TCP/IP: ARP и RARP



9

ARP: Съпоставяне на адреси



10

ARP: Функциониране

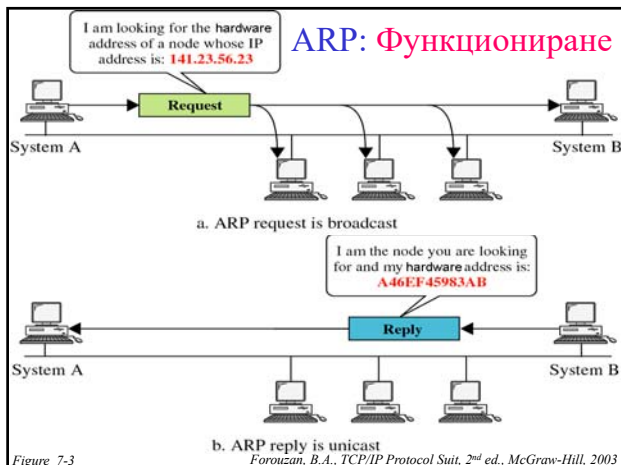


Figure 7-3

Forouzan, B.A., TCP/IP Protocol Suit, 2nd ed., McGraw-Hill, 2003

11

ARP: Формат на пакета

Hardware Type		Protocol Type
Hardware length	Protocol length	Operation Request 1, Reply 2
Sender hardware address (For example, 6 bytes for Ethernet)		
Sender protocol address (For example, 4 bytes for IP)		
Target hardware address (For example, 6 bytes for Ethernet) (It is not filled in a request)		
Target protocol address (For example, 4 bytes for IP)		

Figure 8.3

Forouzan, B.A., TCP/IP Protocol Suit, 4th ed., McGraw-Hill, 2010

12

ARP формат на пакета: Пример

0x0001 (Ethernet)		0x0800 (IPv4)
0x06 (Ethernet)	0x04 (IPv4)	Operation Request 1, Reply 2
Sender hardware address (For example, 6 bytes for Ethernet)		
Sender protocol address (For example, 4 bytes for IP)		
Target hardware address (For example, 6 bytes for Ethernet) (It is not filled in a request)		
Target protocol address (For example, 4 bytes for IP)		

13

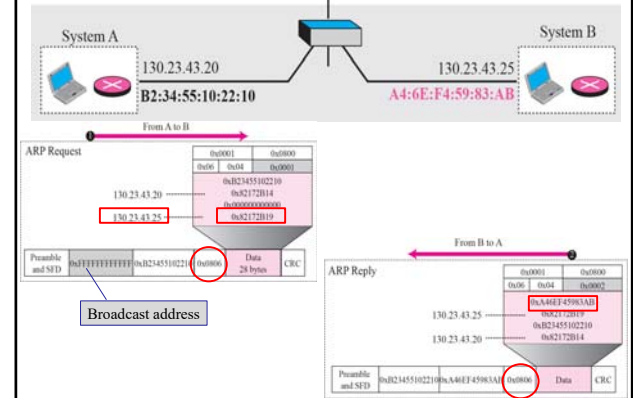
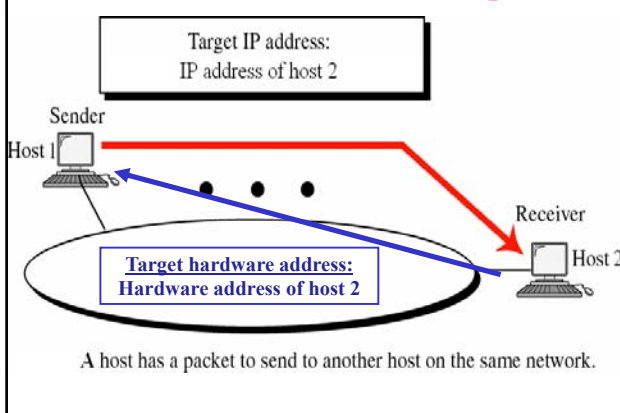
ARP заявка & отговор: Пример

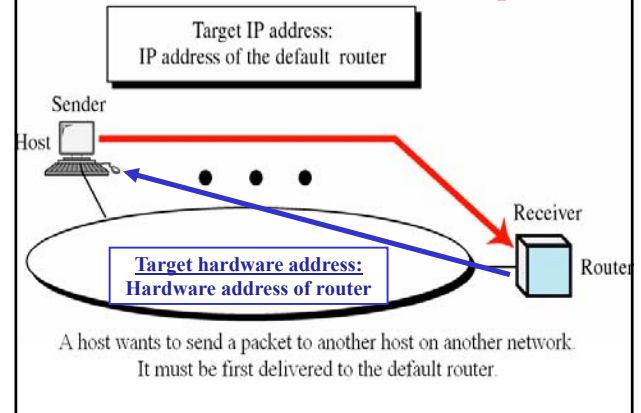
Figure 8.6

Forouzan, B.A., TCP/IP Protocol Suite, 4th ed., McGraw-Hill, 2010

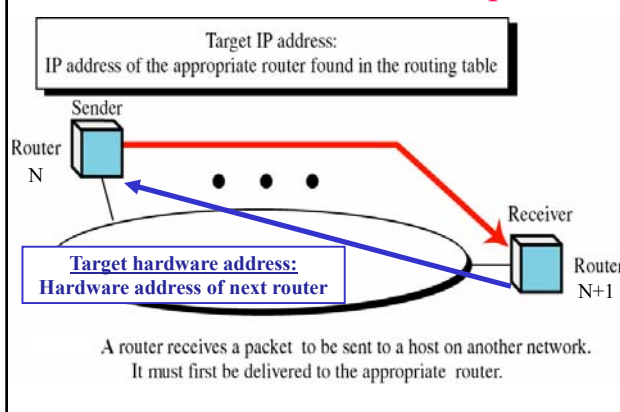
14

Използване на ARP: Сценарий 1

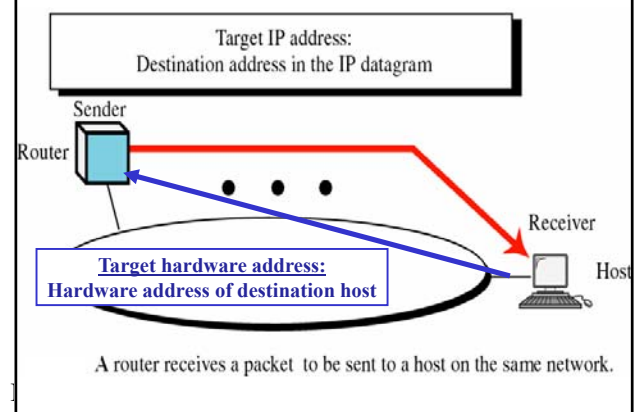
16

Използване на ARP: Сценарий 2

17

Използване на ARP: Сценарий 3

18

Използване на ARP: Сценарий 4

19

ф. Иван

3

Proxy ARP

Използва се за създаване на подмрежа без да се налага промяна в цялата система с цел разпознаване на подмрежови адреси

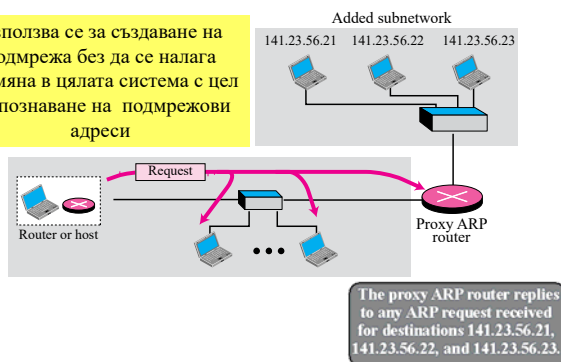


Figure 8.7

Forouzan, B.A., TCP/IP Protocol Suit, 4th ed., McGraw-Hill, 2010

20

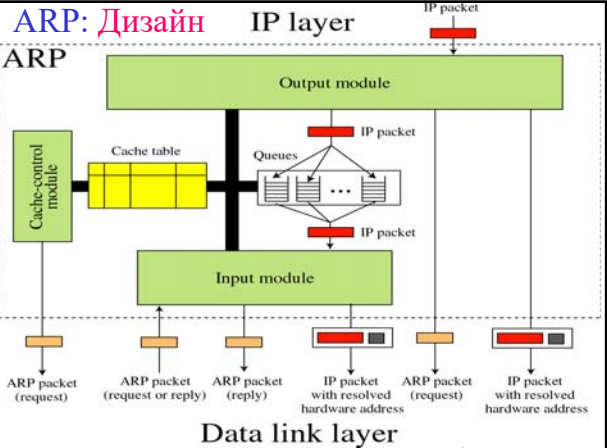


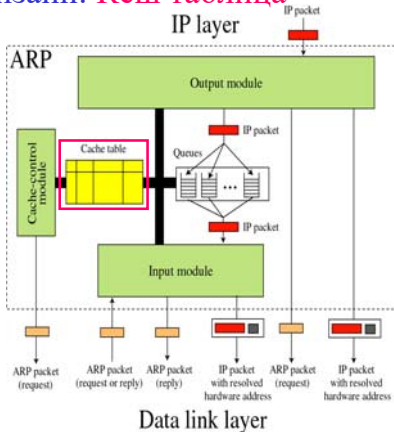
Figure 8.13

Forouzan, B.A., TCP/IP Protocol Suit, 4th ed., McGraw-Hill, 2010

21

ARP дизайн: Кеш таблица

- Неефективност при използване на ARP за всеки IP пакет, предназначен за един и същ възел-получател.
 - 2 допълнителни пакета за всеки IP пакет
- Възниква кеширан ARP отговорите
 - В кеш таблица
 - Прочитана при всяко стартиране на системата
 - Изграждана с течение на времето
 - Претърсвана преди изпращането на всеки IP пакет
 - Ограничено пространство в кеша
 - Съпоставянията на адреси остават в кеша за ограничено време (20-30 минути)
 - Трябва да се потвърждават периодично
 - В противен случай се премахват



22

ARP кеш таблица: Пример

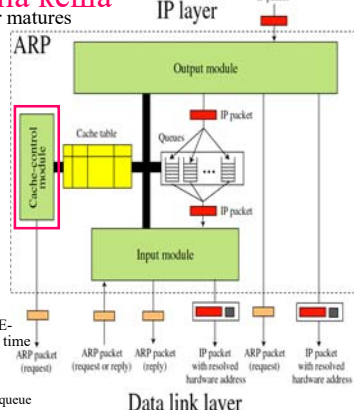
State	Queue	Attempts	Time-out	Protocol address	Hardware address
FREE					
PENDING	2	1		180.3.6.2	
RESOLVED	14		900	180.3.6.1	AC:AE:32:34:73:42

- Реализира се като масив от записи с полета:
 - State: free, pending, resolved
 - Queue number: to enqueue packets waiting for same destination
 - Attempts: No. of ARP requests sent
 - Time-out: lifetime in sec.
 - Etc. corresponding to ARP packet's fields

23

ARP дизайн: Модул за управление на кеша

- Sleep until periodic timer matures
- For every entry
 - If (state is FREE)
 - Continue
 - If (state is PENDING)
 - Increment value of attempts by 1
 - If (attempts > MAX)
 - Change state to FREE
 - Destroy corresponding queue
 - Else
 - Send ARP request
 - Continue
 - If (state is RESOLVED)
 - Decrement value of TIME-OUT by value of elapsed time
 - If (TIME-OUT ≤ 0)
 - Change state to FREE
 - Destroy corresponding queue

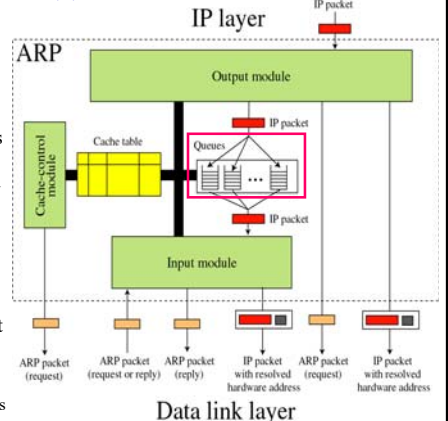


24

ф. Иван

ARP дизайн: Опашки

- 1 queue for each destination
- Holds IP packets while ARP tries to resolve hardware address
- Output module sends unresolved IP packets into corresponding queue
- Input module removes IP packet from queue and send it to DLL for transmission
 - With resolved hardware address

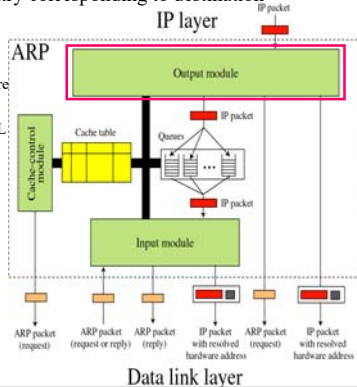


25

4

ARP дизайн: Изходящ модул

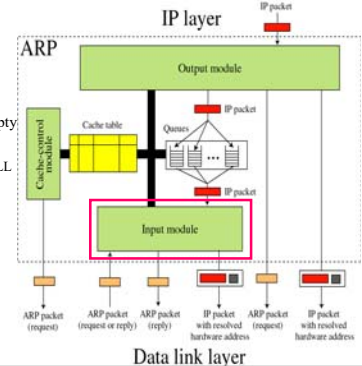
1. Sleep until IP packet received from IP entity
2. Check cache table for entry corresponding to destination IP address
3. If (found)
 1. If (state is RESOLVED)
 1. Extract value of hardware address from entry
 2. Send IP packet & hardware address to DLL
 3. Return
 2. If (state is PENDING)
 1. Enqueue IP packet to corresponding queue
 2. Return
4. If (NOT found)
 1. Create cache entry with state set to PENDING & ATTEMPTS set to 1
 2. Create queue
 3. Enqueue IP packet
 4. Send ARP request
5. Return



26

ARP дизайн: Входящ модул

1. Sleep until ARP packet (request or reply) arrives
2. Check cache table to find entry corresponding to this ARP packet
3. If (found)
 1. Update entry & reset TIME-OUT field
 2. If (state is PENDING)
 1. Change state to RESOLVED
 2. While queue is NOT empty
 1. Dequeue 1 IP packet
 2. Send IP packet & hardware address to DLL
4. If (NOT found)
 1. Create entry
 2. Add entry to cache table
 3. Set state to RESOLVED & set TIME-OUT
5. If (ARP request)
 1. Send ARP reply
6. Return



27

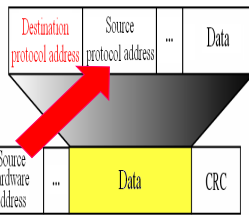
RARP: Съпоставяне на адреси

NL Protocol address (IP)

RARP

DLL Hardware address (MAC)

PACKET



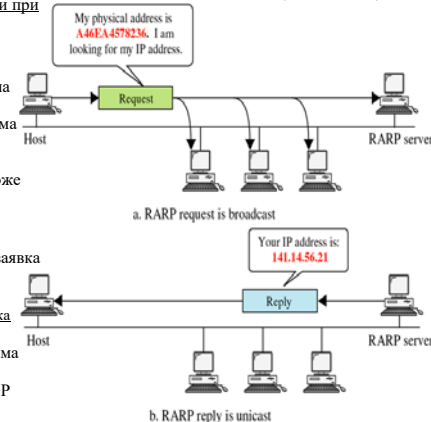
Hardware (MAC) address of source translated into its protocol (IP) address

FRAME

28

Reverse ARP (RARP)

- Например, използван от безжични компютри при стартиране.
- Осигурява превод от MAC към IP адрес
- Пример: Зареждане на бинарен образ на операционната система (без IP адреси) от отдалечен сървър на безжичен PC; как може той да научи своя IP адрес?
- Изпращане (чрез broadcast) на RARP заявка в мрежата
 - 1 RARP сървър е необходим във всяка мрежа
- RARP сървърът приема заявката и изпраща обратно съответния IP адрес



29

RARP: Формат на пакета

Hardware type		Protocol type
Hardware length	Protocol length	Operation Request 3, Reply 4
Sender hardware address (For example, 6 bytes for Ethernet)		
Sender protocol address (For example, 4 bytes for IP) (It is not filled for request)		
Target hardware address (For example, 6 bytes for Ethernet) (It is not filled for request)		
Target protocol address (For example, 4 bytes for IP) (It is not filled for request)		

Figure 7-11

Forouzan, B.A., TCP/IP Protocol Suit, 2nd ed., McGraw-Hill, 2003

ф. Иван

30

RARP: Ограничения

- Използва **DLL broadcasting** за достигане на RARP сървъра
 - Broadcast разпръскванията се спират от маршрутизаторите
 - RARP сървърът трябва да бъде в една и съща физическа мрежа като хоста, изпратил запитване към него.
 - Необходимост от разполагане на RARP сървър във всяка (под)мрежа
- НЕ предоставя цялата информация, необходима на хостовете за минимум IP конфигурация.
 - Например, не предоставя подмрежова маска, IP адрес на маршрутизатор, IP адрес на DNS сървър.
- **Остарял!**
- Заменен от протоколи на приложния слой за преодоляване на тези ограничения
 - **BOOTP, DHCP**
 - Използват *Relay Agents* за преминаване през маршрутизатори

32

5