



Network Layer (Private/Public Addressing, NAT and DHCP)

Redes de Comunicações 1

**Licenciatura em Engenharia de Computadores e
Informática
DETI-UA**

PRIVATE ADDRESSING

2

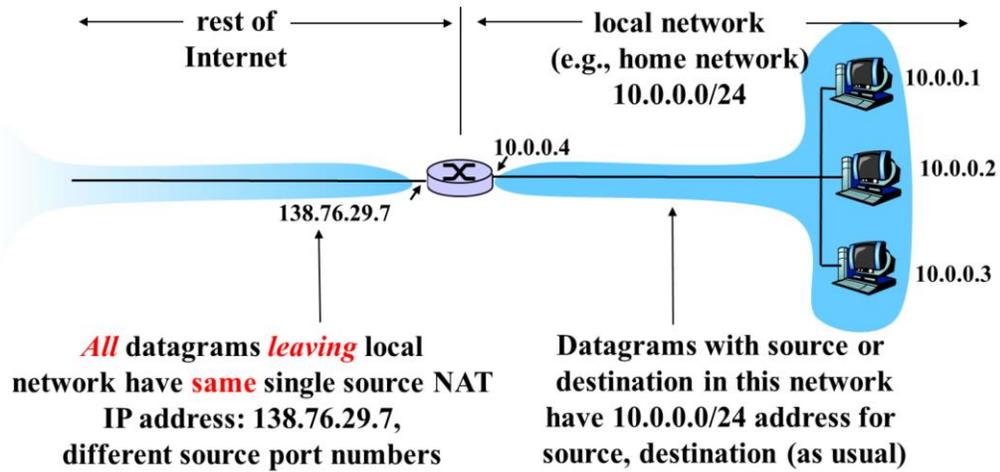
Blocks of private addresses

Prefix	Lowest address	Highest address
10/8	10.0.0.0	10.255.255.255
172.16/12	172.16.0.0	172.31.255.255
192.168/16	192.168.0.0	192.168.255.255
169.254/16	169.254.0.0	169.254.255.255

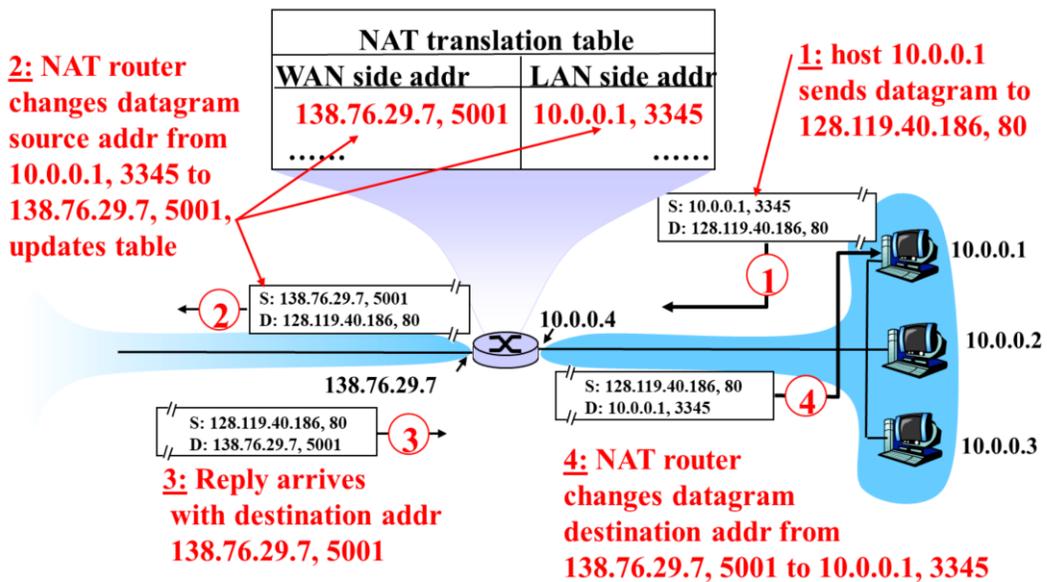
- These addresses can be used freely in private networks
- IP packets with destination addresses belonging to these blocks are not routed in the public network
- For communications with the Internet, the private addresses must be translated into public addresses

3

NAT: Network Address Translation



NAT: Network Address Translation



NAT: Network Address Translation

Implementation: NAT router must:

- *outgoing datagrams: replace* (source IP address, port #) of every outgoing datagram to (NAT IP address, new port #)
 - ... remote clients/servers will respond using (NAT IP address, new port #) as destination addr.
- *remember (in NAT translation table)* every (source IP address, port #) to (NAT IP address, new port #) translation pair
- *incoming datagrams: replace* (NAT IP address, new port #) in dest fields of every incoming datagram with corresponding (source IP address, port #) stored in NAT table

6

NAT: Network Address Translation

- **Motivation:** local network uses just one IP address as far as outside world is concerned:
 - range of addresses not needed from ISP: just one (or a few) IP address for all devices
 - can change addresses of devices in local network without notifying outside world
 - can change ISP without changing addresses of devices in local network
 - devices inside local net not explicitly addressable, visible by outside world (a security additional advantage).

7

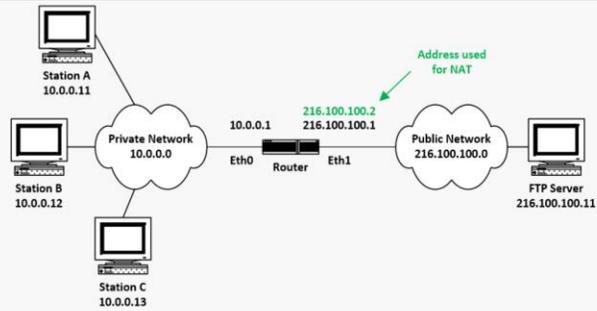
NAT: Network Address Translation

- ❑ More than one public IP address may be available (NAT versus PAT)
- ❑ 16-bit port-number field:
 - more than 60,000 simultaneous connections with a single public IP address!
- ❑ NAT is controversial:
 - routers should only process up to layer 3
 - violates end-to-end argument
 - NAT possibility must be taken into account by application designers, e.g, P2P applications
 - address shortage should instead be solved by IPv6

8

Example (I)

Access of A and B to the FTP server :



No.	Sta	Source Address	DestAddress	Layer	Summary
1	0k	10.0.0.12	216.100.100.11	TCP	1032->File
2	0k	216.100.100.11	10.0.0.12	TCP	File Transf
3	0k	10.0.0.12	216.100.100.11	TCP	1032->File
4	0k	216.100.100.11	10.0.0.12	FTP	220 Serv-U
5	0k	10.0.0.12	216.100.100.11	TCP	1032->File
6	0k	10.0.0.12	216.100.100.11	FTP	USER anonym
7	0k	216.100.100.11	10.0.0.12	FTP	331 User ner

No.	Sta	Source Address	DestAddress	Layer	Summary
12	0k	10.0.0.11	216.100.100.11	TCP	1033->File
13	0k	216.100.100.11	10.0.0.11	TCP	File Transf
14	0k	10.0.0.11	216.100.100.11	TCP	1033->File
15	0k	216.100.100.11	10.0.0.11	FTP	220 Serv-U
16	0k	10.0.0.11	216.100.100.11	TCP	1033->File
17	0k	10.0.0.11	216.100.100.11	FTP	USER anonym
18	0k	216.100.100.11	10.0.0.11	FTP	331 User ner

Captures in private network

No.	Sta	Source Address	DestAddress	Layer	Summary
1	0k	216.100.100.2	216.100.100.11	TCP	1032->File
2	0k	216.100.100.11	216.100.100.2	TCP	File Transf
3	0k	216.100.100.2	216.100.100.11	TCP	1032->File
4	0k	216.100.100.11	216.100.100.2	FTP	220 Serv-U
5	0k	216.100.100.2	216.100.100.11	TCP	1032->File
6	0k	216.100.100.2	216.100.100.11	FTP	USER anonym
7	0k	216.100.100.11	216.100.100.2	FTP	331 User ner

No.	Sta	Source Address	DestAddress	Layer	Summary
1	0k	216.100.100.11	216.100.100.11	TCP	1033->File
2	0k	216.100.100.2	216.100.100.11	TCP	File Transf
3	0k	216.100.100.2	216.100.100.11	TCP	1033->File
4	0k	216.100.100.11	216.100.100.2	FTP	220 Serv-U
5	0k	216.100.100.2	216.100.100.11	TCP	1033->File
6	0k	216.100.100.2	216.100.100.11	FTP	USER anonym
7	0k	216.100.100.11	216.100.100.2	FTP	331 User ner

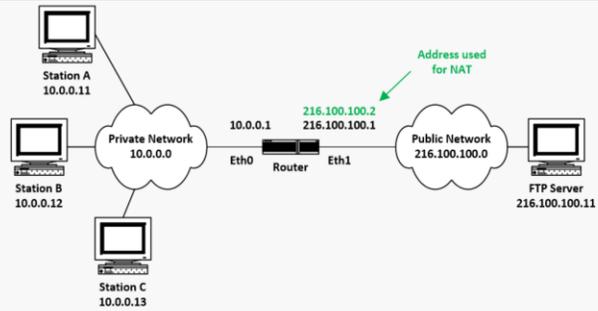
Captures in public network

9

Nesta experiência foi configurado um servidor de FTP na estação D e apenas um endereço público, o 216.100.100.2. A estação A e a estação B vão estabelecer ligações com o servidor de FTP da estação D. Os pacotes do protocolo FTP são encapsulados em pacotes TCP. A estação A abriu a sua ligação com o servidor de FTP no porto 1033, a estação B fê-lo no porto 1032. O router guardou os números dos portos na sua tabela de tradução. Quando chega um pacote do endereço 216.100.100.11 para o endereço 216.100.100.2 e para porto 1032, é enviado para a estação B. Quando chega um pacote do endereço 216.100.100.11 para o endereço 216.100.100.2 e para porto 1033, é enviado para a estação B.

Example (II)

Access of A and B to the FTP server :



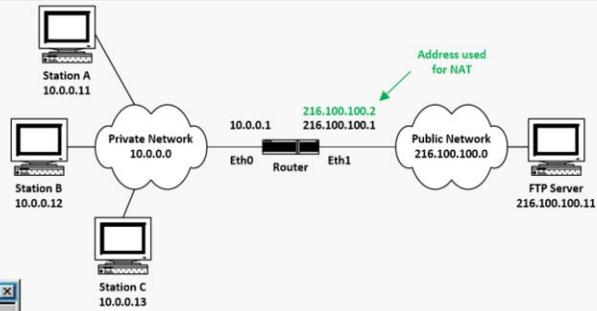
```
Router#show ip nat translation verbose
Pro Inside global      Inside local      Outside local      Outside global
tcp 216.100.100.2:1032 10.0.0.12:1032    216.100.100.11:21 216.100.100.11:21
   create 00:00:35, use 00:00:24, left 23:59:35,
   flags:
   extended, use_count: 0
tcp 216.100.100.2:1033 10.0.0.11:1033    216.100.100.11:21 216.100.100.11:21
   create 00:00:12, use 00:00:06, left 23:59:53,
   flags:
   extended, use_count: 0
```

10

Tabela de traduções do router correspondente à experiência anterior.

Example (III)

Second access of B to the FTP server :



No.	Sta	Source Address	Dest Address	Layer	Summary
1	Ok	10.0.0.12	216.100.100.11	TCP	1033->File
2	Ok	216.100.100.11	10.0.0.12	TCP	File Transf
3	Ok	10.0.0.12	216.100.100.11	TCP	1033->File
4	Ok	216.100.100.11	10.0.0.12	FTP	220 Serv-U
5	Ok	10.0.0.12	216.100.100.11	TCP	1033->File
6	Ok	10.0.0.12	216.100.100.11	FTP	USER anonym
7	Ok	216.100.100.11	10.0.0.12	FTP	331 User ne

private network

No.	Sta	Source Address	Dest Address	Layer	Summary
1	Ok	216.100.100.2	216.100.100.11	TCP	1024->File
2	Ok	216.100.100.11	216.100.100.2	TCP	File Transf
3	Ok	216.100.100.2	216.100.100.11	TCP	1024->File
4	Ok	216.100.100.11	216.100.100.2	FTP	220 Serv-U
5	Ok	216.100.100.2	216.100.100.11	TCP	1024->File
6	Ok	216.100.100.2	216.100.100.11	FTP	USER anonym
7	Ok	216.100.100.11	216.100.100.2	FTP	331 User ne

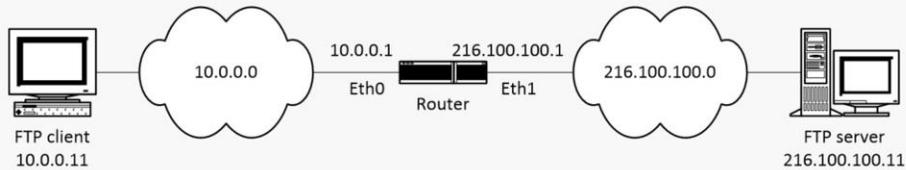
public network

```
Router#show ip nat translation verbose
Pro Inside global   Inside local   Outside local   Outside global
top 216.100.100.2:1024 10.0.0.12:1033 216.100.100.11:21 216.100.100.11:21
   create 00:00:49, use 00:00:42, left 23:59:17,
   flags:
extended, use count: 0
top 216.100.100.2:1032 10.0.0.12:1032 216.100.100.11:21 216.100.100.11:21
   create 00:02:42, use 00:02:31, left 23:57:28,
   flags:
extended, use count: 0
top 216.100.100.2:1033 10.0.0.11:1033 216.100.100.11:21 216.100.100.11:21
   create 00:02:18, use 00:02:13, left 23:57:46,
   flags:
extended, use count: 0
```

11

Este processo é transparente para as estações. Portanto elas podiam ter usado o mesmo número de porta TCP. Vamos abrir mais uma ligação entre a estação B e o servidor de FTP. Desta vez a estação B vai usar o porto 1033, o mesmo que a estação A. O router colocou mais uma entrada na tabela de tradução. Esta entrada relaciona o endereço privado 10.0.0.12 e o porto 1033, com o endereço público 216.100.100.2 e o porto 1024. Ou seja, o router além de traduzir o endereço privado da estação B, traduz também a porta TCP. Pode ver-se que os pacotes são enviados pela estação B para a rede privada a partir do endereço 10.0.0.12 e da porta 1033, e são enviados para a rede pública pelo router como tendo origem no endereço 216.100.100.2 e na porta 1024. Quando a estação D responde, fá-lo para a porta 1024. O router recebe um pacote para a porta 1024 e sabe que o deve enviar para o endereço 10.0.0.12 para a porta 1033.

Example (IV)



Translation of IP address in FTP message (Layer 5):

```

ETHER-II: 00-60-97-9B-9E-07 ==> 00-D0-58-A9-3E-37
Internet Protocol
  Version(MSB 4 bits): 4
  Header length(LSB 4 bits): 5 (32-bit word)
  Service type: Preced-Routine,Delay-Normal,Thrput-Normal,Reli-Normal
  Total length: 61 (Octets)
  Fragment ID: 14595
  Flags: Do not fragment,Last fragment,Offset=0 (0x00)
  Time to live: 128 seconds/hops
  IP protocol type: TCP (0x06)
  Checksum: 0x7B3D
  No option
  IP address 10.0.0.11 ->216.100.100.11
TCP: 1036->File Transfer (Control),S=82367,A=171360,W=86
File Transfer Protocol
  PORT 10,0,0,11,4,13
  Sliced Packet( Data Length = 75)
  
```

private network

```

ETHER-II: 00-D0-58-A9-3E-38 ==> 00-60-97-D4-9F-9A
Internet Protocol
  Version(MSB 4 bits): 4
  Header length(LSB 4 bits): 5 (32-bit word)
  Service type: Preced-Routine,Delay-Normal,Thrput-Normal,Reli-Normal
  Total length: 65 (Octets)
  Fragment ID: 14595
  Flags: Do not fragment,Last fragment,Offset=0 (0x00)
  Time to live: 127 seconds/hops
  IP protocol type: TCP (0x06)
  Checksum: 0x49DD
  No option
  IP address 216.100.100.2 ->216.100.100.11
TCP: 1036->File Transfer (Control),S=82367,A=171360,W=8610
File Transfer Protocol
  PORT 216,100,100,2,4,13
  Sliced Packet( Data Length = 79)
  
```

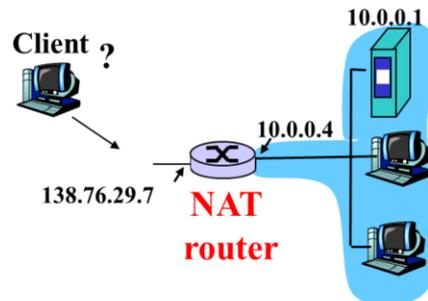
public network

12

Mostra-se um pacote do protocolo FTP capturado na rede pública e na rede privada. Quando o pacote circula na rede privada tem como endereço origem 10.0.0.11. Quando circula na rede pública o endereço destino é 216.100.100.2. O router traduziu o endereço do cliente quando enviou o pacote para a rede pública. Para além disso, o router traduziu os endereços IP transportados no interior dos pacotes FTP. Analisando o pacote FTP vemos que os primeiros quatro octetos são o endereço IP do cliente. Pode ver-se que quando o pacote está na rede privada o endereço do pacote FTP é o endereço privado, quando passou para a rede pública o endereço foi traduzido para o endereço público.

NAT traversal problem

- client wants to connect to server with address 10.0.0.1
 - server address 10.0.0.1 local to LAN (client cannot use it as destination address)
 - only one externally visible NATted address: 138.76.29.7
- solution 1: statically configure NAT to forward incoming connection requests at given port to server
 - e.g., (138.76.29.7, port 2500) always forwarded to 10.0.0.1 port 25000

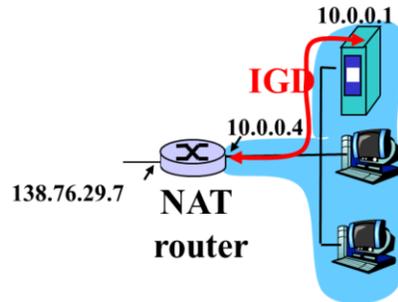


NAT traversal problem

□ solution 2: Universal Plug and Play (UPnP) Internet Gateway Device (IGD) Protocol. Allows NATted host to:

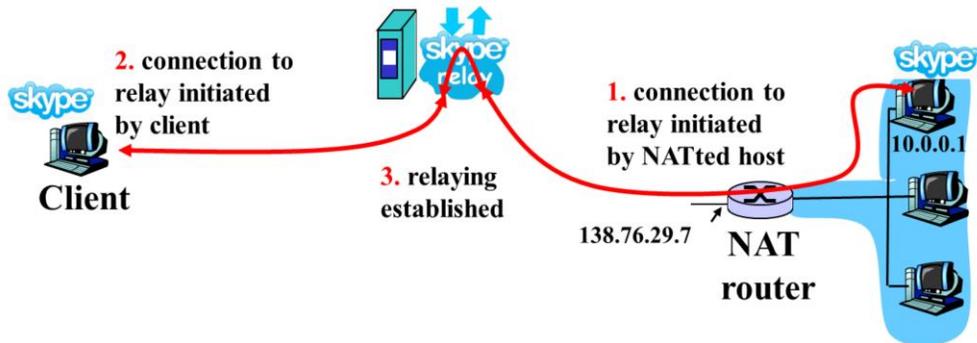
- Application running on NATted host requests mapping between (private IP address, private port number) and (public IP address, public port number)
- Application can advertise (public IP address, public port number)

i.e., automate static NAT port map configuration



NAT traversal problem

- solution 3: relaying (used in Skype)
 - NATed server establishes connection to Relay
 - External client connects to Relay
 - Relay bridges packets between two connections



15

DHCP

16

DHCP: Dynamic Host Configuration Protocol

Goal: allow host to *dynamically* obtain its IP address from network server when it joins network

Can renew its lease on address in use

Allows reuse of addresses (only hold address while connected and "on")

Support for mobile users who want to join network

Also allows client to learn subnet mask, default gateway, local DNS server

DHCP overview:

- host broadcasts "DHCP discover" message
- DHCP server responds with "DHCP offer" message
- host requests IP address: "DHCP request" message
- DHCP server sends address: "DHCP ack" message

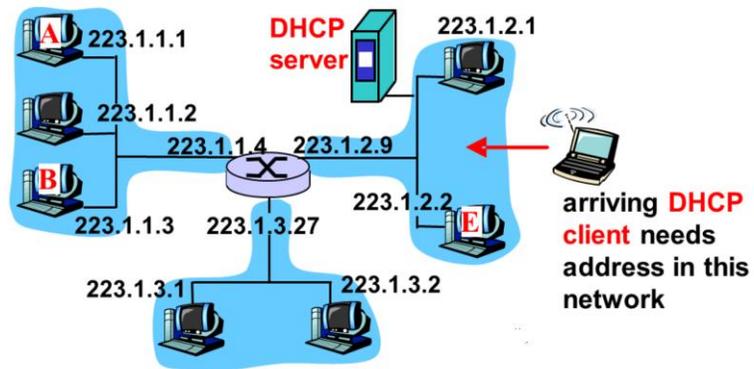
Configuration of a DHCP server

- ❑ Lease address range
 - Pool of IP addresses to be leased defined by the first and the last address
- ❑ Excluding address range
 - Pool of IP addresses inside the lease address range that are not to be leased
- ❑ Reserved addresses
 - IP addresses to be assigned statically for specific MACs
- ❑ Lease time
 - Time duration of the lease of an address

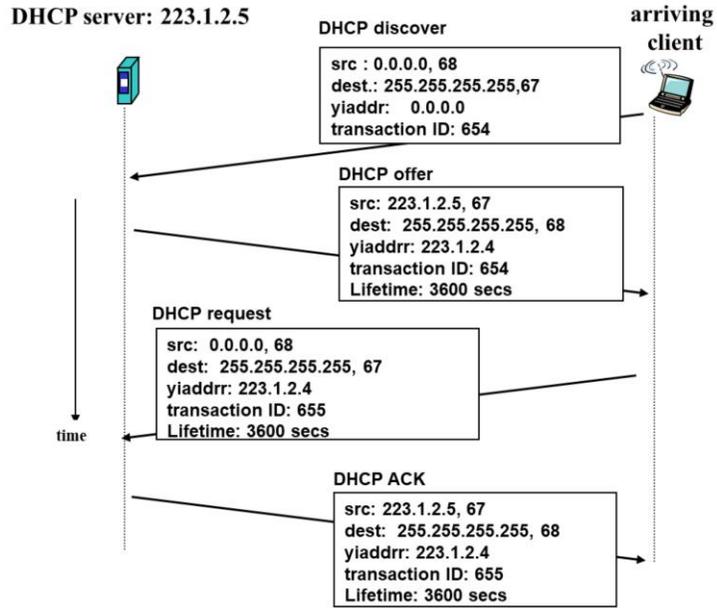
18

Demonstração da configuração de um servidor de DHCP

DHCP client-server scenario



DHCP client-server scenario



DHCP versus BOOTP

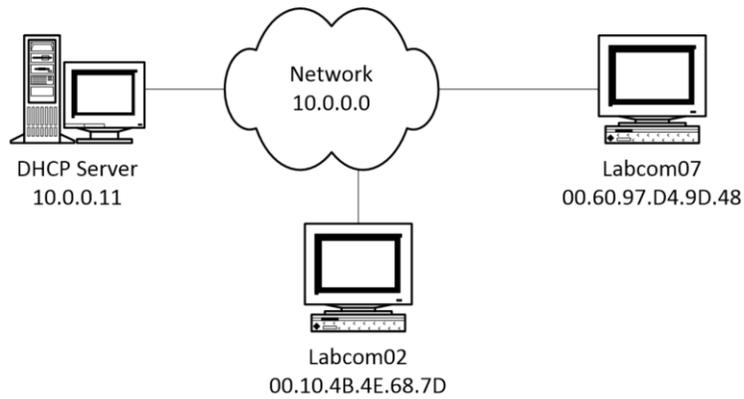
- Extension of *Bootstrap Protocol*, BOOTP, (RFC 1542)
 - Run over UDP
 - Server port number: 67
 - Client port number: 68
 - Originally, BOOTP enables a diskless terminal to find its IP address, a server address and a configuration filename to be requested to the server and locally executed.

21

O cliente DHCP não tem que saber nada sobre a rede quando arranca. Por isso, no primeiro pacote que envia para rede, não conhece nem o seu endereço IP, nem o endereço da rede onde está, nem o endereço IP do servidor DHCP.

A primeira mensagem DHCP enviada é do tipo DHCP Discover. É encapsulada num pacote do tipo pedido BootP. Como o nome indica esta mensagem serve para descobrir os servidores de DHCP que existem na rede. O cliente pode também indicar neste pacote qual o endereço IP que pretende alugar. As máquinas Windows NT 4.0 tentam alugar sempre o último endereço que usaram. Os servidores de DHCP que recebem o pedido respondem com uma mensagem do tipo DHCP Offer. Essa mensagem é encapsulada num pacote do tipo resposta BootP. O pacote BootP tem que ser enviado para o endereço de broadcast, uma vez que o cliente ainda não possui endereço IP. Nessa resposta cada servidor diz ao cliente que endereço IP lhe pode fornecer. Os servidores tentam sempre respeitar a preferência do cliente por um determinado endereço. O cliente recebe as respostas dos servidores. Se algum dos servidores lhe oferecer o endereço IP que pediu, escolhe-o. Senão escolhe qualquer um dos servidores (provavelmente o primeiro que responder). O cliente envia então um pacote do tipo DHCP Request ao servidor que ofereceu o endereço IP que ele escolheu. Nesse pacote o cliente pede ao servidor que lhe seja fornecido o endereço. Quando o servidor recebe o pedido reserva o endereço IP para o cliente e responde com um DHCP Acknowledge.

Example

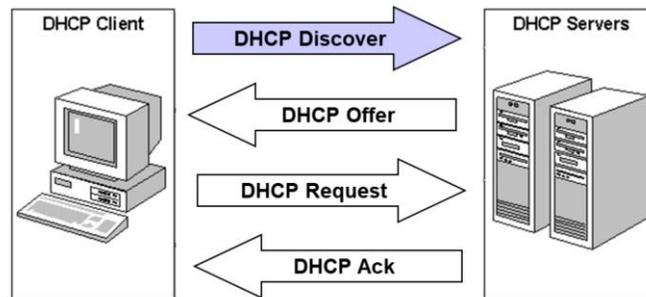


22

Nesta experiência montámos uma rede onde existe um servidor de DHCP e alguns clientes. Um desses clientes, a estação labcom07, está a arrancar quando a experiência começa. Com a estação labcom02 capturamos os pacotes trocados entre a estação labcom07 e o servidor DHCP.

DHCP Discover

***DHCP Discover* message is encapsulated on a *BootP Request* message. It is sent to discover available DHCP servers. The client can also indicate an IP address.**

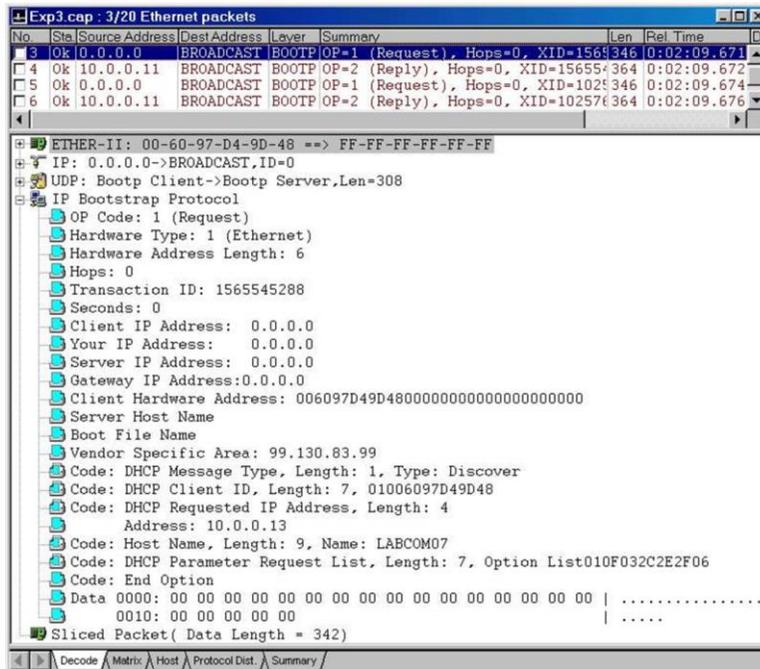


23

O cliente DHCP não tem que saber nada sobre a rede quando arranca. Por isso, no primeiro pacote que envia para rede, não conhece nem o seu endereço IP, nem o endereço da rede onde está, nem o endereço IP do servidor DHCP.

A primeira mensagem DHCP enviada é do tipo DHCP Discover. É encapsulada num pacote do tipo pedido BootP. Como o nome indica esta mensagem serve para descobrir os servidores de DHCP que existem na rede. O cliente pode também indicar neste pacote qual o endereço IP que pretende alugar. As máquinas Windows NT 4.0 tentam alugar sempre o último endereço que usaram. Os servidores de DHCP que recebem o pedido respondem com uma mensagem do tipo DHCP Offer. Essa mensagem é encapsulada num pacote do tipo resposta BootP. O pacote BootP tem que ser enviado para o endereço de broadcast, uma vez que o cliente ainda não possui endereço IP. Nessa resposta cada servidor diz ao cliente que endereço IP lhe pode fornecer. Os servidores tentam sempre respeitar a preferência do cliente por um determinado endereço. O cliente recebe as respostas dos servidores. Se algum dos servidores lhe oferecer o endereço IP que pediu, escolhe-o. Senão escolhe qualquer um dos servidores (provavelmente o primeiro que responder). O cliente envia então um pacote do tipo DHCP Request ao servidor que ofereceu o endereço IP que ele escolheu. Nesse pacote o cliente pede ao servidor que lhe seja fornecido o endereço. Quando o servidor recebe o pedido reserva o endereço IP para o cliente e responde com um DHCP Acknowledge.

DHCP Discover

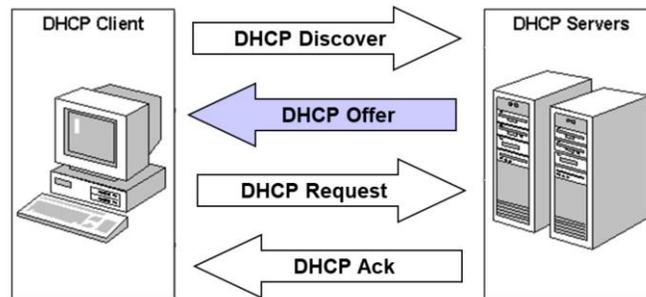


24

A primeira medida tomada pelo cliente é o envio de uma mensagem do tipo DHCP Discover. Essa mensagem não leva endereço IP de origem, porque o cliente ainda não conhece o seu endereço IP. Essa mensagem é transportada por um pacote do tipo BootP request. Numa das células pode ver-se que o cliente pede o endereço IP 10.0.0.13, ele sabe qual foi o último endereço IP com que trabalhou, portanto tenta continuar com o mesmo. Este pacote BootP tem que ser enviado para o endereço de broadcast, porque a estação não conhece o endereço IP do servidor.

DHCP Offer

***DHCP Offer* message is encapsulated on a *BootP Reply* message. Each server offers one IP address to lease (if possible, servers offer the IP address indicated by the client).**



25

O cliente DHCP não tem que saber nada sobre a rede quando arranca. Por isso, no primeiro pacote que envia para rede, não conhece nem o seu endereço IP, nem o endereço da rede onde está, nem o endereço IP do servidor DHCP.

A primeira mensagem DHCP enviada é do tipo DHCP Discover. É encapsulada num pacote do tipo pedido BootP. Como o nome indica esta mensagem serve para descobrir os servidores de DHCP que existem na rede. O cliente pode também indicar neste pacote qual o endereço IP que pretende alugar. As máquinas Windows NT 4.0 tentam alugar sempre o último endereço que usaram. Os servidores de DHCP que recebem o pedido respondem com uma mensagem do tipo DHCP Offer. Essa mensagem é encapsulada num pacote do tipo resposta BootP. O pacote BootP tem que ser enviado para o endereço de broadcast, uma vez que o cliente ainda não possui endereço IP. Nessa resposta cada servidor diz ao cliente que endereço IP lhe pode fornecer. Os servidores tentam sempre respeitar a preferência do cliente por um determinado endereço. O cliente recebe as respostas dos servidores. Se algum dos servidores lhe oferecer o endereço IP que pediu, escolhe-o. Senão escolhe qualquer um dos servidores (provavelmente o primeiro que responder). O cliente envia então um pacote do tipo DHCP Request ao servidor que ofereceu o endereço IP que ele escolheu. Nesse pacote o cliente pede ao servidor que lhe seja fornecido o endereço. Quando o servidor recebe o pedido reserva o endereço IP para o cliente e responde com um DHCP Acknowledge.

DHCP Offer

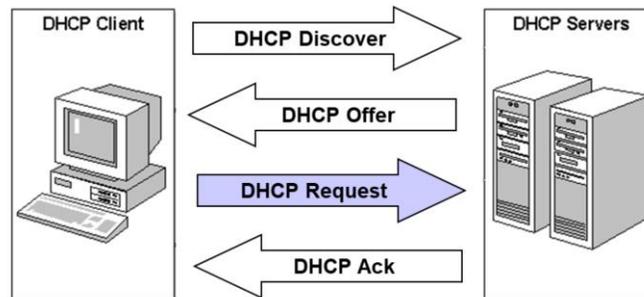
```
ETHER-II: 00-60-97-9B-9B-3F ==> FF-FF-FF-FF-FF-FF
IP: 10.0.0.11->BROADCAST,ID=4250
UDP: Bootp Server->Bootp Client,Len=326
IP Bootstrap Protocol
  OP Code: 2 (Reply)
  Hardware Type: 1 (Ethernet)
  Hardware Address Length: 6
  Hops: 0
  Transaction ID: 1565545288
  Seconds: 0
  Client IP Address: 0.0.0.0
  Your IP Address: 10.0.0.13
  Server IP Address: 10.0.0.11
  Gateway IP Address:0.0.0.0
  Client Hardware Address: 006097D49D4800000000000000000000
  Server Host Name
  Boot File Name
  Vendor Specific Area: 99.130.83.99
  Code: DHCP Message Type, Length: 1, Type: Offer
  Code: Subnet Mask, Length: 4 Address:255.0.0.0
  Code: DHCP Renewal (T1) Time, Length: 4, Value:150
  Code: DHCP Rebinding (T2) Time, Length: 4, Value:262
  Code: DHCP IP Address Lease Time, Length: 4, Value:300
  Code: DHCP Server ID, Length: 4
  Address: 10.0.0.11
  Code: Domain Name, Length: 17, Name: labcom.det.ua.pt
  Code: Router, Length: 4
  Address: 10.0.0.1
  Code: NetBIOS Name Server, Length: 8
  Address: 193.136.173.202
  Address: 193.136.173.203
  Code: NetBIOS over TCP/IP, Length: 1, Node Type:0x8 H-node
  Code: Domain Name Server, Length: 4
  Address: 10.0.0.11
  Code: End Option
Sliced Packet( Data Length = 360)
```

26

O único servidor que existe na rede recebe esta mensagem DHCP Discover e verifica que pode oferecer ao cliente o endereço IP que ele pretende. Envia então ao cliente uma mensagem DHCP Offer onde indica que lhe pode oferecer o endereço IP 10.0.0.13. Essa mensagem é transportada por um pacote BootP. Ainda não é agora que o endereço IP vai ficar alugado, mas mesmo assim este servidor envia toda a informação sobre a rede que está configurado para enviar.

DHCP Request

DHCP Request message is encapsulated on a ***BootP Request*** message. After selecting one of the offers, the client indicates the selected IP address.



27

O cliente DHCP não tem que saber nada sobre a rede quando arranca. Por isso, no primeiro pacote que envia para rede, não conhece nem o seu endereço IP, nem o endereço da rede onde está, nem o endereço IP do servidor DHCP.

A primeira mensagem DHCP enviada é do tipo DHCP Discover. É encapsulada num pacote do tipo pedido BootP. Como o nome indica esta mensagem serve para descobrir os servidores de DHCP que existem na rede. O cliente pode também indicar neste pacote qual o endereço IP que pretende alugar. As máquinas Windows NT 4.0 tentam alugar sempre o último endereço que usaram. Os servidores de DHCP que recebem o pedido respondem com uma mensagem do tipo DHCP Offer. Essa mensagem é encapsulada num pacote do tipo resposta BootP. O pacote BootP tem que ser enviado para o endereço de broadcast, uma vez que o cliente ainda não possui endereço IP. Nessa resposta cada servidor diz ao cliente que endereço IP lhe pode fornecer. Os servidores tentam sempre respeitar a preferência do cliente por um determinado endereço. O cliente recebe as respostas dos servidores. Se algum dos servidores lhe oferecer o endereço IP que pediu, escolhe-o. Senão escolhe qualquer um dos servidores (provavelmente o primeiro que responder). O cliente envia então um pacote do tipo DHCP Request ao servidor que ofereceu o endereço IP que ele escolheu. Nesse pacote o cliente pede ao servidor que lhe seja fornecido o endereço. Quando o servidor recebe o pedido reserva o endereço IP para o cliente e responde com um DHCP Acknowledge.

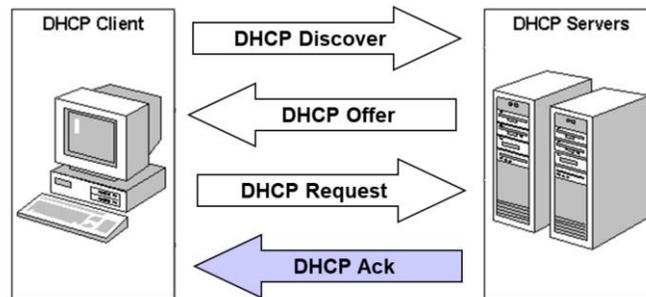
DHCP Request

```
ETHER-II: 00-60-97-D4-9D-48 ==> FF-FF-FF-FF-FF-FF
IP: 0.0.0.0->BROADCAST, ID=256
UDP: Bootp Client->Bootp Server, Len=308
IP Bootstrap Protocol
  OP Code: 1 (Request)
  Hardware Type: 1 (Ethernet)
  Hardware Address Length: 6
  Hops: 0
  Transaction ID: 1025766432
  Seconds: 0
  Client IP Address: 0.0.0.0
  Your IP Address: 0.0.0.0
  Server IP Address: 0.0.0.0
  Gateway IP Address: 0.0.0.0
  Client Hardware Address: 006097D49D4800000000000000000000
  Server Host Name
  Boot File Name
  Vendor Specific Area: 99.130.83.99
  Code: DHCP Message Type, Length: 1, Type: Request
  Code: DHCP Client ID, Length: 7, 01006097D49D48
  Code: DHCP Requested IP Address, Length: 4
  Address: 10.0.0.13
  Code: DHCP Server ID, Length: 4
  Address: 10.0.0.11
  Code: Host Name, Length: 9, Name: LABCOM07
  Code: DHCP Parameter Request List, Length: 7, Option List 010F032C2E2F06
  Code: End Option
  Data 0000: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 | .....
Sliced Packet( Data Length = 342)
```

Quando o cliente recebe a oferta do servidor resolve aceitá-la. Envia então uma mensagem do tipo DHCP Request, para tentar alugar o endereço. Nesta rede só existe um servidor de DHCP, mas podiam existir mais. Se fosse o caso cada um teria feito a sua proposta. Por isso, na mensagem DHCP Request, além do endereço que está a pedir, o cliente indica também qual o servidor que deve atender ao pedido. Mesmo conhecendo já o endereço IP do servidor, este pacote tem que ser enviado para o endereço de Broadcast. Podem ter sido vários os servidores de DHCP a responder ao cliente com ofertas. Esses servidores devem ser informados de que a sua oferta foi rejeitada. Enviando a mensagem DHCP Request todos os servidores ficam a saber qual a oferta que foi aceite. Assim, os servidores cuja oferta não foi aceite, podem libertar o endereço que ofereceram para outro cliente.

DHCP Ack

***DHCP Ack* message is encapsulated in a *BootP Reply* message. The server acknowledges positively the lease of the IP address indicating also other information of interest.**



29

O cliente DHCP não tem que saber nada sobre a rede quando arranca. Por isso, no primeiro pacote que envia para rede, não conhece nem o seu endereço IP, nem o endereço da rede onde está, nem o endereço IP do servidor DHCP.

A primeira mensagem DHCP enviada é do tipo DHCP Discover. É encapsulada num pacote do tipo pedido BootP. Como o nome indica esta mensagem serve para descobrir os servidores de DHCP que existem na rede. O cliente pode também indicar neste pacote qual o endereço IP que pretende alugar. As máquinas Windows NT 4.0 tentam alugar sempre o último endereço que usaram. Os servidores de DHCP que recebem o pedido respondem com uma mensagem do tipo DHCP Offer. Essa mensagem é encapsulada num pacote do tipo resposta BootP. O pacote BootP tem que ser enviado para o endereço de broadcast, uma vez que o cliente ainda não possui endereço IP. Nessa resposta cada servidor diz ao cliente que endereço IP lhe pode fornecer. Os servidores tentam sempre respeitar a preferência do cliente por um determinado endereço. O cliente recebe as respostas dos servidores. Se algum dos servidores lhe oferecer o endereço IP que pediu, escolhe-o. Senão escolhe qualquer um dos servidores (provavelmente o primeiro que responder). O cliente envia então um pacote do tipo DHCP Request ao servidor que ofereceu o endereço IP que ele escolheu. Nesse pacote o cliente pede ao servidor que lhe seja fornecido o endereço. Quando o servidor recebe o pedido reserva o endereço IP para o cliente e responde com um DHCP Acknowledge.

DHCP Ack

```
ETHER-II: 00-60-97-9B-9B-3F ==> FF-FF-FF-FF-FF-FF
IP: 10.0.0.11->BROADCAST, ID=4251
UDP: Bootp Server->Bootp Client, Len=326
IP Bootstrap Protocol
  OP Code: 2 (Reply)
  Hardware Type: 1 (Ethernet)
  Hardware Address Length: 6
  Hops: 0
  Transaction ID: 1025766432
  Seconds: 0
  Client IP Address: 0.0.0.0
  Your IP Address: 10.0.0.13
  Server IP Address: 0.0.0.0
  Gateway IP Address: 0.0.0.0
  Client Hardware Address: 006097D49D4800000000000000000000
  Server Host Name
  Boot File Name
  Vendor Specific Area: 99.130.83.99
  Code: DHCP Message Type, Length: 1, Type: Ack
  Code: DHCP Renewal (T1) Time, Length: 4, Value:150
  Code: DHCP Rebinding (T2) Time, Length: 4, Value:262
  Code: DHCP IP Address Lease Time, Length: 4, Value:300
  Code: DHCP Server ID, Length: 4
    Address: 10.0.0.11
  Code: Subnet Mask, Length: 4    Address:255.0.0.0
  Code: Domain Name, Length: 17, Name: labcom.det.ua.pt
  Code: Router, Length: 4
    Address: 10.0.0.1
  Code: NetBIOS Name Server, Length: 8
    Address: 193.136.173.202
    Address: 193.136.173.203
  Code: NetBIOS over TCP/IP, Length: 1, Node Type:0x8 H-node
  Code: Domain Name Server, Length: 4
    Address: 10.0.0.11
  Code: End Option
Sliced Packet( Data Length = 360)
```

30

O servidor recebeu o pedido do endereço 10.0.0.13 do cliente e responde com uma mensagem DHCP Acknowledge. Ficou então alugado o endereço 10.0.0.13 ao cliente. No pacote BootP que transporta a mensagem DHCP Acknowledge vão também indicados todos os parâmetros de configuração cedidos pelo servidor. São eles a máscara da rede, o endereço do router que deve servir de default gateway, o endereço IP do servidor de DNS, o nome do domínio DNS e os endereços IP dos servidores de WINS.

Existem na mensagem DHCP Acknowledge algumas células DHCP muito importantes. São elas o tempo de lease, o tempo de renovação T1 e o tempo de renovação T2. O tempo de lease indica o período de tempo que o cliente pode usar o endereço IP. Os campos instante de renovação T1 e instante de renovação T2 indicam ao cliente quando é que ele deve tentar renovar o endereço, se estiver interessado. Esses instantes indicam o que vem estabelecido na norma. O instante T1 é 50% do tempo de lease e o instante T2 é 87.5% do tempo de lease.

Address leasing

- *Renewal Time (50% of Lease Time)*
 - at this time, the client should try to renew the lease in the server that has assigned its IP address
- *Rebinding Time (85% of Lease Time)*
 - at this time, the client should try to renew the lease of its IP address (if it didn't succeed before) in any available server
- *Lease Time*
 - if the lease could not be renewed, at this time, the client stop using the IP address

31

O cliente DHCP não tem que saber nada sobre a rede quando arranca. Por isso, no primeiro pacote que envia para rede, não conhece nem o seu endereço IP, nem o endereço da rede onde está, nem o endereço IP do servidor DHCP.

A primeira mensagem DHCP enviada é do tipo DHCP Discover. É encapsulada num pacote do tipo pedido BootP. Como o nome indica esta mensagem serve para descobrir os servidores de DHCP que existem na rede. O cliente pode também indicar neste pacote qual o endereço IP que pretende alugar. As máquinas Windows NT 4.0 tentam alugar sempre o último endereço que usaram. Os servidores de DHCP que recebem o pedido respondem com uma mensagem do tipo DHCP Offer. Essa mensagem é encapsulada num pacote do tipo resposta BootP. O pacote BootP tem que ser enviado para o endereço de broadcast, uma vez que o cliente ainda não possui endereço IP. Nessa resposta cada servidor diz ao cliente que endereço IP lhe pode fornecer. Os servidores tentam sempre respeitar a preferência do cliente por um determinado endereço. O cliente recebe as respostas dos servidores. Se algum dos servidores lhe oferecer o endereço IP que pediu, escolhe-o. Senão escolhe qualquer um dos servidores (provavelmente o primeiro que responder). O cliente envia então um pacote do tipo DHCP Request ao servidor que ofereceu o endereço IP que ele escolheu. Nesse pacote o cliente pede ao servidor que lhe seja fornecido o endereço. Quando o servidor recebe o pedido reserva o endereço IP para o cliente e responde com um DHCP Acknowledge.

Other DHCP messages

- ❑ **DHCP Decline:**
 - The client rejects the offer of a server and restarts the acquisition of an IP address
- ❑ **DHCP Nack:**
 - The server informs that it cannot renew the lease of an IP address
- ❑ **DHCP Release:**
 - The client informs the server that it is no longer interested on an IP address
- ❑ **DHCP Inform:**
 - The client requests additional information (in this case, the client has an IP address but requires, for example, the DNS server address)

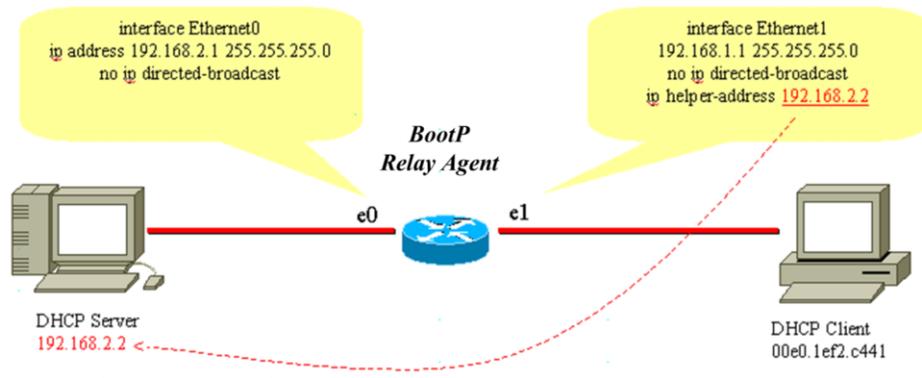
32

O cliente DHCP não tem que saber nada sobre a rede quando arranca. Por isso, no primeiro pacote que envia para rede, não conhece nem o seu endereço IP, nem o endereço da rede onde está, nem o endereço IP do servidor DHCP.

A primeira mensagem DHCP enviada é do tipo DHCP Discover. É encapsulada num pacote do tipo pedido BootP. Como o nome indica esta mensagem serve para descobrir os servidores de DHCP que existem na rede. O cliente pode também indicar neste pacote qual o endereço IP que pretende alugar. As máquinas Windows NT 4.0 tentam alugar sempre o último endereço que usaram. Os servidores de DHCP que recebem o pedido respondem com uma mensagem do tipo DHCP Offer. Essa mensagem é encapsulada num pacote do tipo resposta BootP. O pacote BootP tem que ser enviado para o endereço de broadcast, uma vez que o cliente ainda não possui endereço IP. Nessa resposta cada servidor diz ao cliente que endereço IP lhe pode fornecer. Os servidores tentam sempre respeitar a preferência do cliente por um determinado endereço. O cliente recebe as respostas dos servidores. Se algum dos servidores lhe oferecer o endereço IP que pediu, escolhe-o. Senão escolhe qualquer um dos servidores (provavelmente o primeiro que responder). O cliente envia então um pacote do tipo DHCP Request ao servidor que ofereceu o endereço IP que ele escolheu. Nesse pacote o cliente pede ao servidor que lhe seja fornecido o endereço. Quando o servidor recebe o pedido reserva o endereço IP para o cliente e responde com um DHCP Acknowledge.

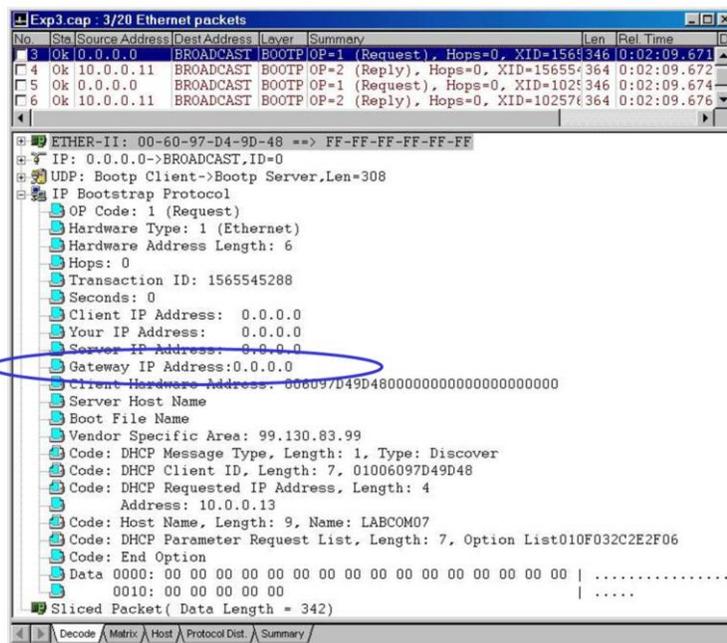
Client and server in different subnets

- It is necessary to make routers behave as *BootP Relay Agents*
- Routers reroute *BootP Request* messages to the IP address of the DHCP server inserting the IP address of the receiver interface in the *Gateway IP* address field of the DHCP messages
- DHCP server sends *BootP Reply* messages to the *Gateway IP* address



33

Gateway IP address field



34

A primeira medida tomada pelo cliente é o envio de uma mensagem do tipo DHCP Discover. Essa mensagem não leva endereço IP de origem, porque cliente ainda não conhece o seu endereço IP. Essa mensagem é transportada por um pacote do tipo BootP request. Numa das células pode ver-se que o cliente pede o endereço IP 10.0.0.13, ele sabe qual foi o último endereço IP com que trabalhou, portanto tenta continuar com o mesmo. Este pacote BootP tem que ser enviado para o endereço de broadcast, porque a estação não conhece o endereço IP do servidor.

Bibliography to study

- J. Kurose, K. Ross, "Computer Networking: A Top-Down Approach", Addison-Wesley, 4th Edition
 - Section 4.4.2 "IPv4 Addressing"

IPv6

IPv6 Features

- Larger address space enabling:
 - Global reachability, flexibility, aggregation, multihoming, autoconfiguration, “plug and play” and renumbering
- Simpler header enabling:
- Routing efficiency, performance and forwarding rate scalability
- Improved option support

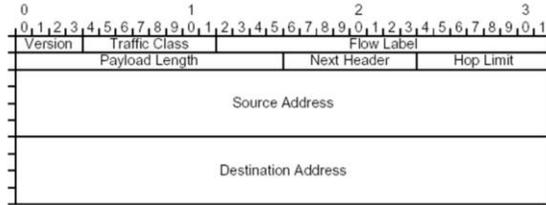
IPv6 Addressing

- IPv4: 4bytes/32 bits
 - ~ 4,294,967,296 possible addresses
- IPv6: 16bytes/128 bits
 - 340,282,366,920,938,463,463,374,607,431,768,211,456 possible addresses
- Representation
 - 16-bit hexadecimal numbers
 - Hex numbers are not case sensitive
 - Numbers are separated by (:)
 - Abbreviations are possible
 - Leading zeros in contiguous block could be represented by (::)
 - Example:
 - 2001:0db8:0000:130F:0000:0000:087C:140B = 2001:0db8:0:130F::87C:140B
 - Double colon only appears once in the address
 - Address's prefix is represented as: prefix/mask_number_of_bits

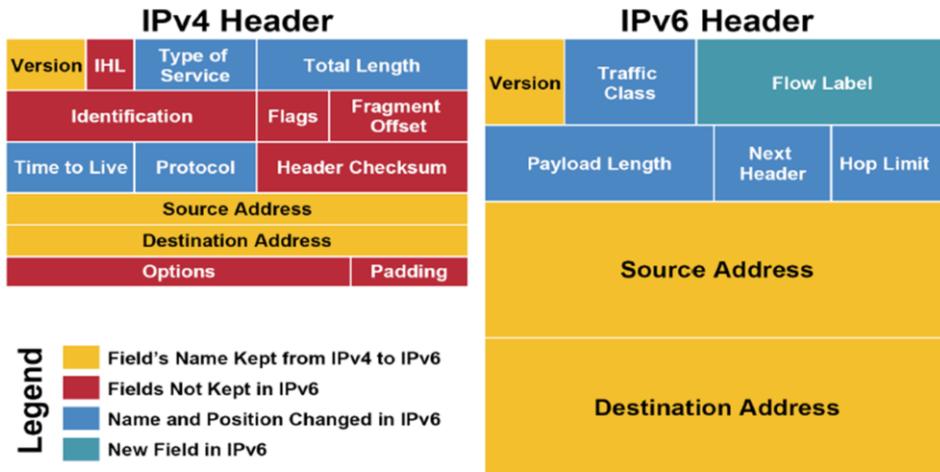
38

Header

- ❑ Fixed-length
40 byte header
- ❑ Checksum removed
- ❑ No options in base header
- ❑ New flow label field (use is currently not defined, actually several uses exist for each purpose)
- ❑ Faster packet processing (with hardware support)
- ❑ Options in flexible and extensible extension headers (can be transparent for transit nodes)



IPv6 Header compared to IPv4 Header



Extension Headers

- ❑ Hop-by-hop (various, e.g. discard packet w/o ICMP)
- ❑ Routing (address list → source routing)
- ❑ Fragment (fragmentation only done by source)
- ❑ Destination options (various , e.g. discard packet w/o ICMP)
- ❑ Authentication (integrity and data origin auth.)
- ❑ Encapsulating security payload (confidentiality)

IPv6 address format

2001:0DA8:E800:0000:0260:3EFF:FE47:0001

- 8 groups of 4 hexadecimal digits
 - Each group represents 16 bits
 - Separator is ":"
 - Case-independent
- Addresses in IPv6 very complex
 - Auto configured
 - Local addresses

IPv6 Addressing

Type	Binary	Hexadecimal
<i>Global Unicast Address</i>	0010	2
<i>Link-Local Unicast Address</i>	1111 1110 10	FE80::/10
<i>Unique-Local Unicast Address</i>	1111 1100 1111 1101	FC00::/8 FD00::/8
<i>Multicast Address</i>	1111 1111	FF00::/16

43

Examples

- ❑ 2200:A:A::1/64
- ❑ 2001:db8:a0b:12f0::1/64
- ❑ 2731:54:65fe:2::a7/64
- ❑ fe80::19b3:fddb:75f9:740/64

- ❑ My PC
 - IPv6 address:
2001:0:9d38:90d7:30a6:16a4:3f57:e09e
 - Link local IPv6
address:fe80::30a6:16a4:3f57:e09e

44

IPv6 Addressing Scheme

□ Interface have multiple addresses

□ Addresses have scope:

○ Link Local

- Valid within the same LAN or link

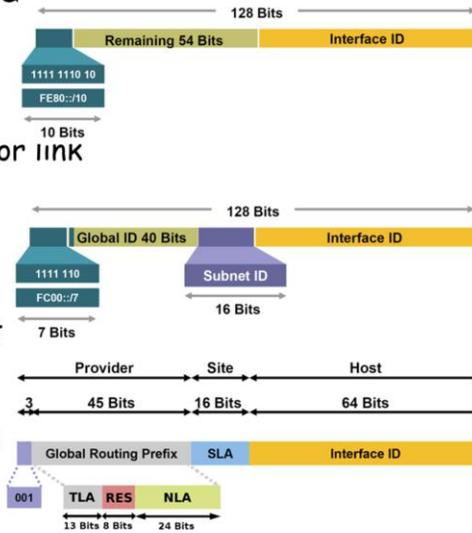
○ Unique Local

- Valid within the same private domain
- Can not be used in Internet

○ Global

□ Addresses have lifetime

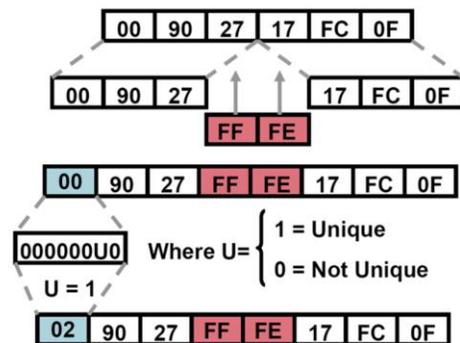
- Valid and preferred lifetime



IPv6 Interface Identifier

□ Lowest-Order 64-Bit field of any address:

- Auto-configured from a 64-bit EUI-64, or expanded from a 48-bit MAC address (e.g. Ethernet address)
- Auto-generated pseudo-random number
- Assigned via DHCP
- Manually configured



Auto-configuration

- ❑ Important concept in the IPv6 birth
 - Terminal needs to automatically obtain its configuration information;
 - Network configuration elements can be changed and automatically propagated to all terminals.
- ❑ Auto-configuration methods
 - Stateless: configuration determined by the network;
 - Stateful: configuration determined by the network management.
- ❑ Process:
 - Link-local is created;
 - Verify the uniqueness of the link-local address (DAD-duplicate address detection)
 - Selection of the configuration method;
 - Determine the information to be auto-configured (addresses, gateways, ...)

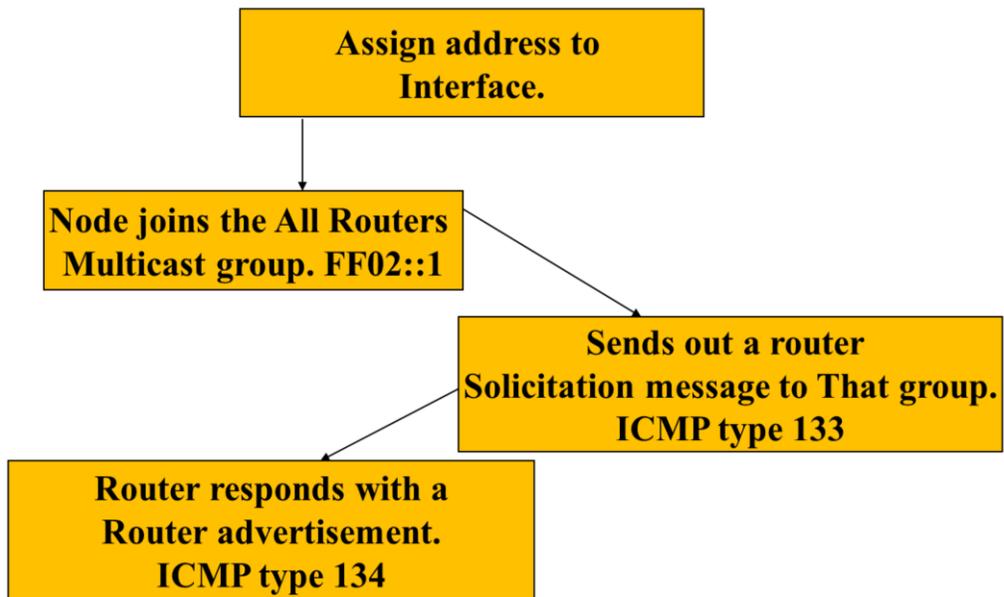
47

Stateless auto-configuration

- Terminal generates its own address. It combines
 - Local information (e.g. MAC address);
 - Information is advertised by routers (prefix defines the local sub-network).
- Advantages:
 - No manual configuration of the terminals;
 - Minimal configuration in the routers;
 - No additional servers.
- If there are no routers, terminal creates its link-local address
 - This address is sufficient to allow a communication in the same local network segment.

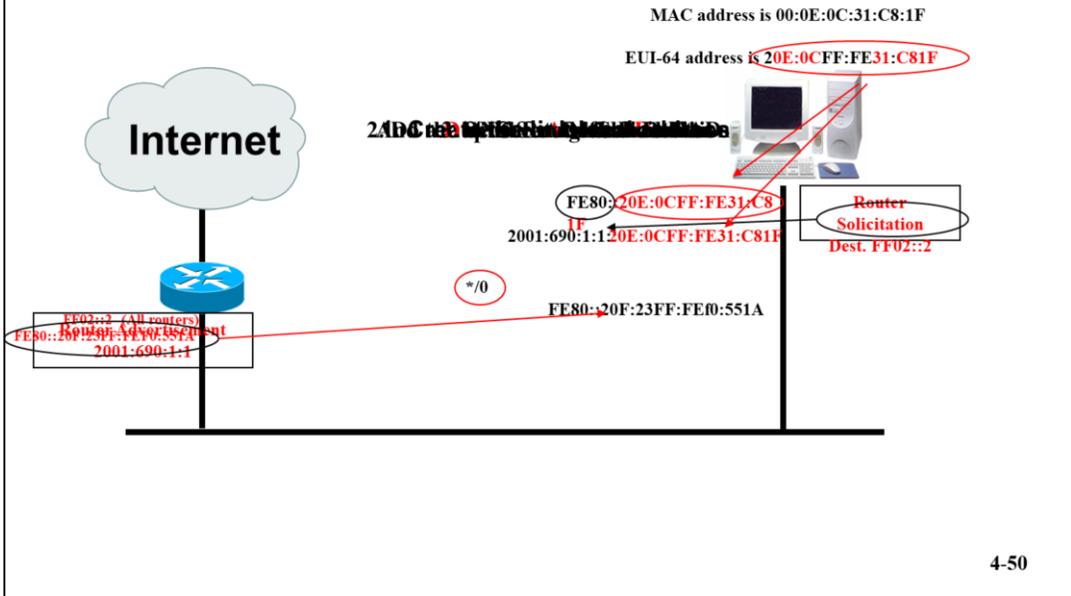
48

Stateless auto-configuration



49

Stateless auto-configuration



Statefull auto-configuration

- ❑ Configuration based on a client-server model
 - Use the Dynamic Host Control Protocol (DHCPv6).
- ❑ DHCPv6 allows:
 - Allocation of IPv6 addresses to the terminals;
 - Delivery of specific configuration information of each terminal.
- ❑ Guarantees larger configuration control (no DADs required);
- ❑ Supports IPv6 concepts of automatic configuration
 - E.g. automatic change of addresses.

51