

IPv6

FEUP

MPR

A Necessidade de um Novo IP

- ◆ IPv4
 - » Espaço de endereçamento pequeno
 - » Usado de forma descontínua
 - » Algumas alterações para resolver estes problemas
 - superredes (CDIR), NAT
 - » Mas, problema aparentemente inultrapassável
- ◆ IETF começa (1991) a desenvolver nova versão do IP → o IPv6
 - » Mesmos princípios arquitectónicos do IPv4
 - » Muitas melhorias
 - » Implica mudança do cabeçalho
- ◆ O IPv6 pode vir a ser relevante para as Comunicações Móveis
 - » 1 endereço IP para cada telefone

IPv6

- ◆ Endereços de 128 bits (16 octetos, 8 *shorts*)
- ◆ Facilita suporte de QoS (serviços de tempo real)
- ◆ Inclui funções de segurança
 - » autenticação das partes; cifragem dos dados
- ◆ Autoconfiguração
 - » *Plug-n-play*
- ◆ Routing melhorado
 - » Em particular no suporte de terminais móveis
- ◆ Multicast
- ◆ Transição suave do IPv4

Espaço de Endereçamento e Notação

- ◆ Sem classes
 - » Prefixo do endereço indica tipo de utilização → unicast, multicast, anycast
 - Anycast: envio dos pacotes para o membro do grupo mais próximo
 - » Prefixos podem ser usados para transformar endereços v4 em endereços v6
- ◆ Representação dos endereços
 - » 8 x 16 bit, hexadecimal, separados por :
 - 47CD:1234:3200:0000:0000:4325:B792:0428 ← Formato preferido
 - » Formato comprimido
 - FF01:0:0:0:0:0:43 → FF01::43
 - » Compatibilidade com IPv4
 - 0:0:0:0:0:0:13.1.68.3 ou ::13.1.68.3
 - » Não há endereços de difusão. Só multicast
 - » Endereço de loopback → ::1
 - » Prefixos com notação de / , como no v4
 - FEDC:BA98:7600::/40 → prefixo de 40 bits

Endereços IPv6

Allocation	Prefix (binary)	Fraction of Address Space
-----	-----	-----
Unassigned	0000 0000	1/256
Unassigned	0000 0001	1/256
Reserved for NSAP Allocation	0000 001	1/128
Unassigned	0000 01	1/64
Unassigned	0000 1	1/32
Unassigned	0001	1/16
Global Unicast	001	1/8
Unassigned	010	1/8
Unassigned	011	1/8
Unassigned	100	1/8
Unassigned	101	1/8
Unassigned	110	1/8
Unassigned	1110	1/16
Unassigned	1111 0	1/32
Unassigned	1111 10	1/64
Unassigned	1111 110	1/128
Unassigned	1111 1110 0	1/512
Link-Local Unicast Addresses	1111 1110 10	1/1024
Site-Local Unicast Addresses	1111 1110 11	1/1024
Multicast Addresses	1111 1111	1/256

Endereços – *Link-Local, Site-Local*

◆ Endereços *Link-Local*

- Usados para endereçamento numa ligação ou LAN
- Configurado automaticamente
- Routers não encaminham pacotes com endereços Link-Local para outras redes

◆ Endereços *Site-Local*

- Usado para endereços dentro de um site/empresa
- Podem ser feitas sub-redes
- Routers não encaminham pacotes com endereços *Site-Local* para fora do Site

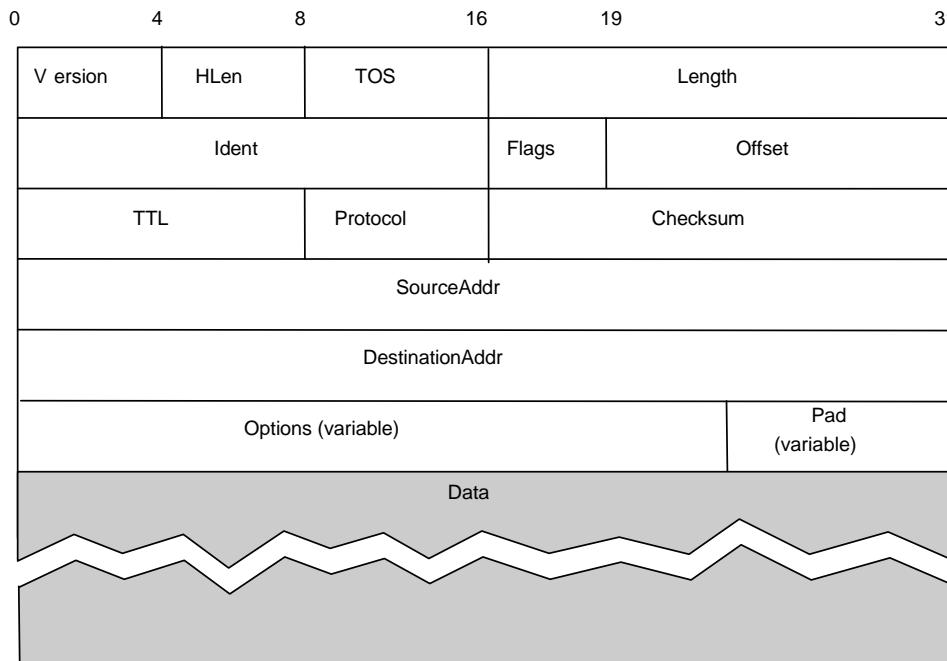
Formato dos Endereços

	n bits		m bits		128-n-m bits		Global Unicast Address (2000:: 3)</td
	001 global rout prefix		subnet ID		interface ID		
	10 bits		54 bits		64 bits		Link-Local Unicast address (fe80:: 10)</td
	1111111010		0		interface ID		
	10 bits		54 bits		64 bits		Site-Local Unicast address (fec0:: 10)</td
	1111111011		subnet ID		interface ID		
	n bits					128-n bits	Anycast address
	subnet prefix					00000000000000	
	8		4		4		112 bits
	11111111		flgs		scop		group ID
							Multicast address grupo permanente ou não Scope - link, site, global, ... (ff:: 8)</td

Endereços – Global Unicast

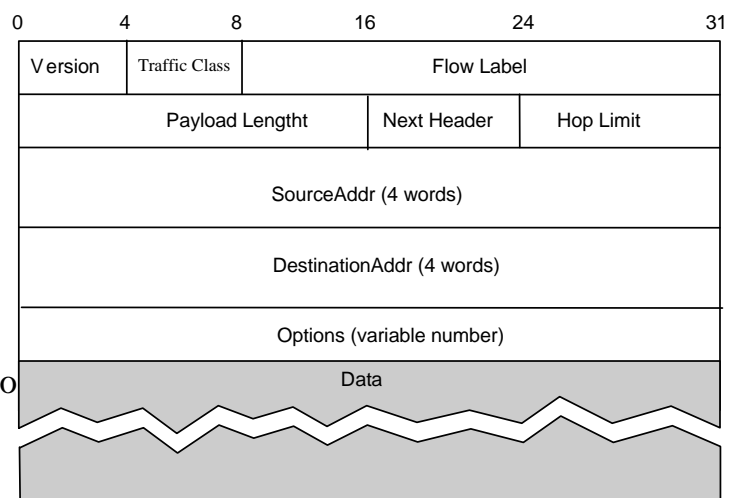
- ◆ Semelhante ao CIDR
 - » 001 → 3 bits mais significativos do endereço (2000:: - » Endereço → prefixo de rede + identificador do computador
 - » Prefixos estruturados → agregação de subredes em redes maiores
 - Menos entradas nas tabelas de encaminhamento
- ◆ Endereços Anycast → tratados como endereços unicast

Datagrama IPv4

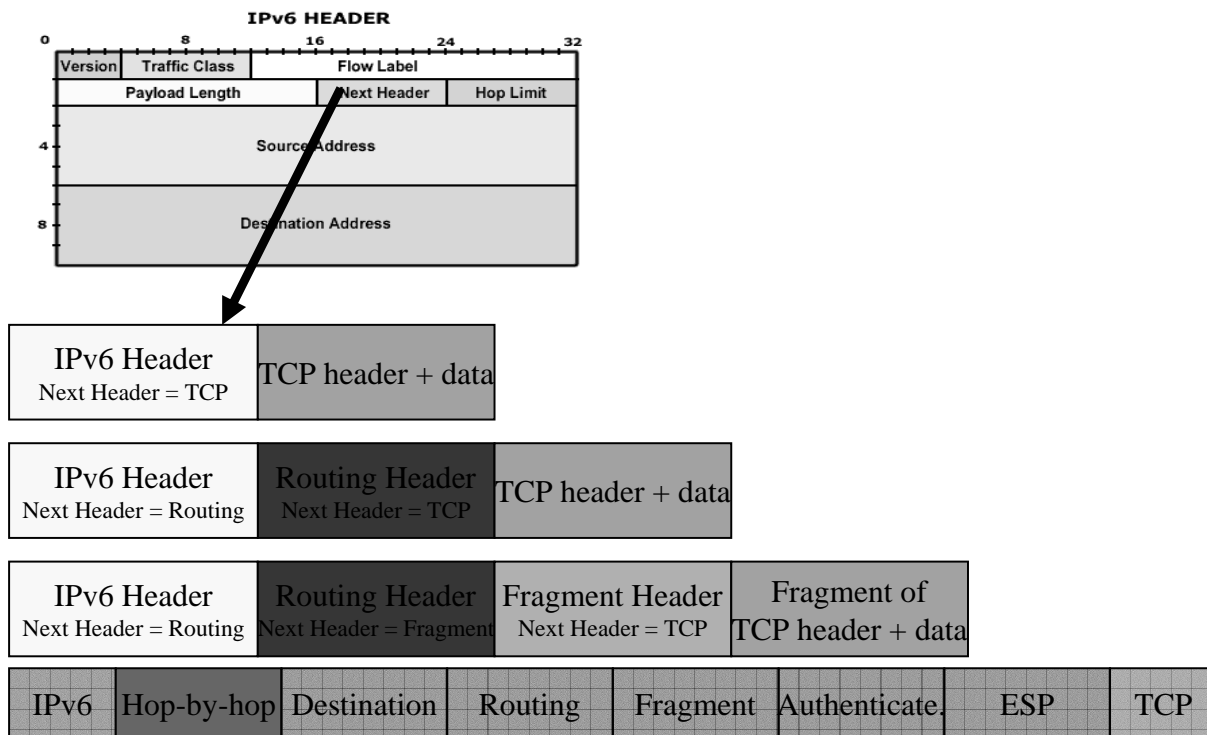


Formato do Pacote IPv6

- ◆ Mais simples que v4
- ◆ Version = 6. Traffic class = ToS, em v4
- ◆ Flow label → identifica fluxo pacotes
 - » QoS, reserva de recursos
 - » Pacotes do fluxo com tratamento idêntico
- ◆ Payload length
 - » não inclui cabeçalho
 - » máximo = 64 kbytes
- ◆ Hop limit = TTL, em v4
- ◆ Next header → opções, próximo protocolo
- ◆ Options → cabeçalho de extensão
 - » 6 tipos comuns
 - » encaminhamento, fragmentação
 - » autenticação, cifragem, ...



Cabeçalhos de Extensão



Tipos de Extensão

- ◆ Hop-by-hop
 - » Informação adicional, examinada por todos os nós atravessados por pacote
 - Examinado em todos os routers; os outros headers só são examinados no destino
- ◆ Destination
 - » Informação para o computador de destino
- ◆ Routing
 - » Lista de nós a visitar pelo pacote
- ◆ Fragmentation
 - » Feita pelo emissor do pacote, que deve descobrir o MTU
- ◆ Authentication (segurança)
 - » Autenticação (assinatura) do cabeçalho do pacote
- ◆ ESP (segurança)
 - » Informação sobre criptografia dados

Cabeçalho IPv6 – Principais Diferenças

- ◆ Não há checksum → erros detectados noutra camada,
 - Recurso ao pseudo header no TCP ou UDP
- ◆ Cabeçalho de comprimento fixo
 - » Tempo de processamento do pacote diminui
- ◆ Não há informação de fragmentação/reassemblagem
 - » Pacote de comprimento incorrecto → eliminado
 - Mensagem enviada para emissor, para reduzir comprimento pacote
- ◆ Computador responsável pela descoberta do MTU
- ◆ Segmentação possível em cabeçalho de extensão

Extensão de Fragmentação

- » Semelhante à fragmentação IPv4
 - Implementado como cabeçalho de extensão. Colocado entre cabeçalho IPv6 e dados
- » Offset de 13 bit
- » M → indica último fragmento do pacote
- » ID, Identificador do pacote. Maior que em v4
- » Fragmentação feita nos equipamentos terminais da ligação

0	8	16	29	31
next header	reserved	offset	reserved	M
ID				

Extensão de Routing

- ◆ Sem este cabeçalho, encaminhamento igual a v4
- ◆ Com este cabeçalho, semelhante ao Source Routing, do v4
 - » Lista de nós a visitar
- ◆ Header length → em palavras de 64 bit
- ◆ Podem ser incluídos até 24 endereços

0	8	16	24
31			
Next header	Hd. Ext. Len	0	Segmnts left
1 – 24 addresses			

Routing Header - Encaminhamento do Pacote^{IPv6 16}

Pacote enviado de S para D, por I1, I2, I3

As the packet travels from S to I1:

Source Address = S	Hdr Ext Len = 6
Destination Address = I1	Segments Left = 3
	Address[1] = I2
	Address[2] = I3
	Address[3] = D

As the packet travels from I1 to I2:

Source Address = S	Hdr Ext Len = 6
Destination Address = I2	Segments Left = 2
	Address[1] = I1
	Address[2] = I3
	Address[3] = D

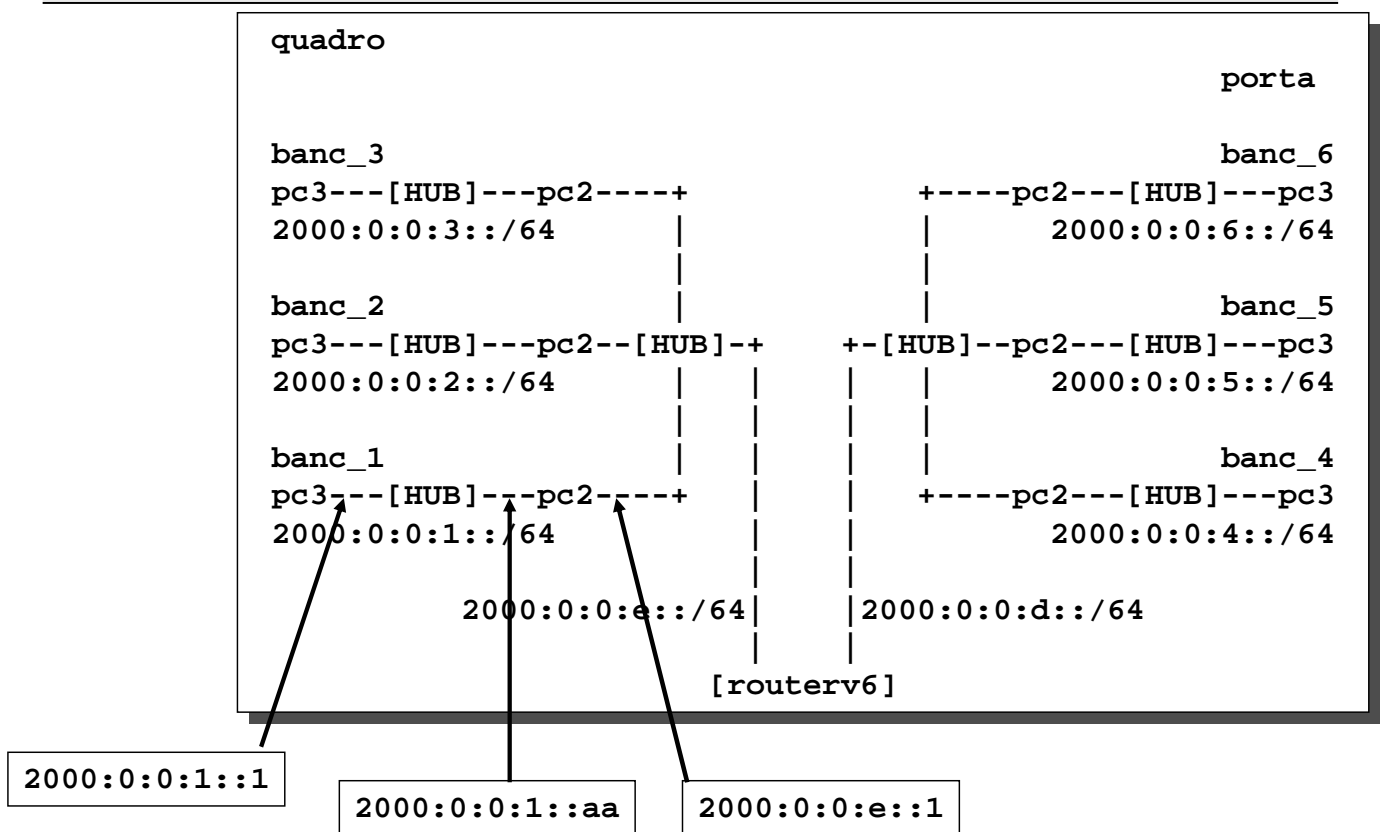
As the packet travels from I2 to I3:

Source Address = S	Hdr Ext Len = 6
Destination Address = I3	Segments Left = 1
	Address[1] = I1
	Address[2] = I2
	Address[3] = D

As the packet travels from I3 to D:

Source Address = S	Hdr Ext Len = 6
Destination Address = D	Segments Left = 0
	Address[1] = I1
	Address[2] = I2
	Address[3] = I3

Exemplo – Rede no Lab de Redes



Configuração da bancada 1, pc3 (tux13)

```

tux13:~# /sbin/ifconfig eth0 inet6 add 2000:0:0:1::1/64
tux13:~# ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 00:C0:DF:08:D5:99
          inet addr:172.16.1.13  Bcast:172.16.1.255  Mask:255.255.255.0
          inet6 addr: 2000:0:0:1::1/64 Scope:Global
          inet6 addr: fe80::2c0:dfff:fe08:d599/10 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:81403 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2429 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:100
          RX bytes:4981344 (4.7 MiB)  TX bytes:260692 (254.5 KiB)
          Interrupt:5
tux13:~# /sbin/route -A inet6 add 2000::/3 gw 2000:0:0:1::aa
tux13:~# route -A inet6
Kernel IPv6 routing table
Destination      NextHop          Flags Metric Ref Use Iface
::1/128          ::               U      0     0  0  lo
2000:0:0:1::1/128  ::               U      0     0  0  lo
2000:0:0:1::/64   ::               UA     256   0  0  eth0
2000::/3         2000:0:0:1::aa UG     1     0  0  eth0
fe80::2c0:dfff:fe08:d599/128  ::               U      0     0  0  lo
fe80::/10        ::               UA     256   0  0  eth0
ff00::/8         ::               UA     256   0  0  eth0
::/0             ::               UDA    256   0  0  eth0
  
```

Identificador IEEE EUI-64

Method to create a IEEE EUI-64 identifier from an IEEE 48bit MAC identifier. This is to insert two octets, with hexadecimal values of 0xFF and 0xFE, in the middle of the 48 bit MAC (between the company_id and vendor supplied id). For example, the 48 bit IEEE MAC with global scope:

```

| 0           1 | 1           3 | 3           4 |
| 0           5 | 6           1 | 2           7 |
+-----+-----+-----+
| ccccc0gccccccc | cccccccmmmmmmmmmm | mmmmmmmmmmmmmmmmmmm |
+-----+-----+-----+

```

00:C0:DF:08:D5:99

where "c" are the bits of the assigned company_id, "0" is the value of the universal/local bit to indicate global scope, "g" is individual/group bit, and "m" are the bits of the manufacturer-selected extension identifier. The interface identifier would be of the form:

```

| 0           1 | 1           3 | 3           4 | 4           6 |
| 0           5 | 6           1 | 2           7 | 8           3 |
+-----+-----+-----+-----+
| ccccc1gccccccc | ccccccc11111111 | 11111110mmmmmmmmmm | mmmmmmmmmmmmmmmmmmm |
+-----+-----+-----+-----+

```

fe80::2c0:dfff:fe08:d599

Transição do v4 para o v6

- ◆ Computador opera em dupla pilha de protocolos (RFC2893)
 - Computador v6 funciona também em v4
 - Versão indica stack a usar
- ◆ Dois computadores v6 podem usar rede v4 (RFC2893)
 - Adição de 96 0s a um endereço v4 de 32-bit. Ex. **::10.0.0.1**
 - 2 computadores falam v6 entre si, mesmo que a rede seja toda v4
 - túnel (v4 contendo v6) entre computadores
- ◆ Computador v6 consegue falar com computador v4
 - Usado endereço v6 mapeável em endereço v4
 - Adição de 2 bytes de 1s ao endereço v4 + extensão de zeros
 - Ex **::ffff:10.0.0.1**
 - O pacote circula na rede em v4. É convertido para v6 internamente