Instituto Politécnico de Leiria

Escola Superior de Tecnologia e Gestão

IPv6@ESTG-Leiria

Mecanismos de transição

Cenários

Luís Alberto Esteves Diogo

Óscar Filipe Correia Brilha

Leiria

Setembro de 2006

Instituto Politécnico de Leiria

Escola Superior de Tecnologia e Gestão

IPv6@ESTG-Leiria

Mecanismos de transição

Documento extra da cadeira de Projecto I, do curso de Licenciatura em Engenharia Informática e Comunicações, ano lectivo 2005/2006

Realizado entre Março e Julho de 2006

Autores:

Luís Alberto Esteves Diogo, nº 10429

Óscar Filipe Correia Brilha, nº 9061

Orientador: Prof. Nuno Veiga

Leiria

Setembro de 2006

Índice

Lista de figuras

Resumo

Este documento foca a vertente prática da transição focando os mecanismos de túneis, ou seja, mostra como implementar os diversos tipos de túneis usados para transitar para o novo Protocolo IPv6. O software IOS Cisco não tem suporte para os túneis Teredo logo esse tipo de túnel não será abordado neste documento. Os primeiros cenários apresentados são relativamente simples e têm como objectivo mostrar o formato dos pacotes que circulam entre routers de modo a se perceber como os cabeçalhos IPv4 e IPv6 circulam em conjunto.

Os restantes cenários são já cenários mais elaborados e que se aproximam de situações reais de transição podendo ser usados como exemplos em casos reais de transição.

São abordados também mecanismos de NAT, DNS e DHCPv6 por os consideramos importantes na transição para IPv6

Ao longo do documento são explicadas as configurações usadas em routers e Pcs mas em anexo seguem também as configurações completas copiadas dos routers Cisco de alguns dos cenários.

1. ISATAP

Os túneis ISATAP foram concebidos para fornecerem conectividade IPv6 entre hosts de uma intranet. É um mecanismo muito semelhante ao mecanismo 6over4 (obsoleto) com a particularidade de não usar Multicast, característica que foi a causa do insucesso dos túneis 6over4. São túneis apropriados para usar em intranets. Para poder ligar a uma rede IPv6 exterior a intranet é necessário o uso de túneis 6to4.

Cenário 1

Resumo

Os endereços ISATAP criam numa intranet a sua própria rede, ou seja garantem automaticamente a conectividade entre máquinas Windows sem qualquer configuração em especial. O cenário apresentado tem como objectivo mostrar o formato dos pacotes capturados numa rede em que se usem endereços ISATAP.

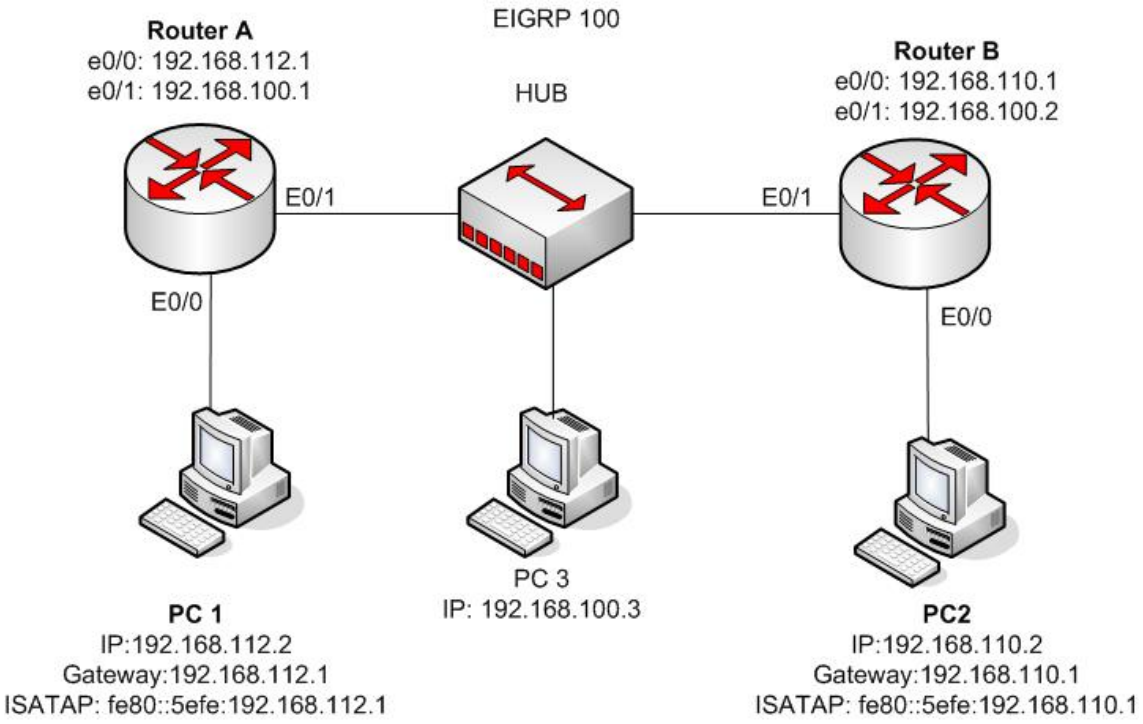

Figura 1.1 – Cenário da implementação de um túnel ISATAP

Foi efectuado um ping do pc 2 para o pc 3 (ping fe80::5efe:192.168.112.1.)

Pacote capturado no PC3:

Echo Request:

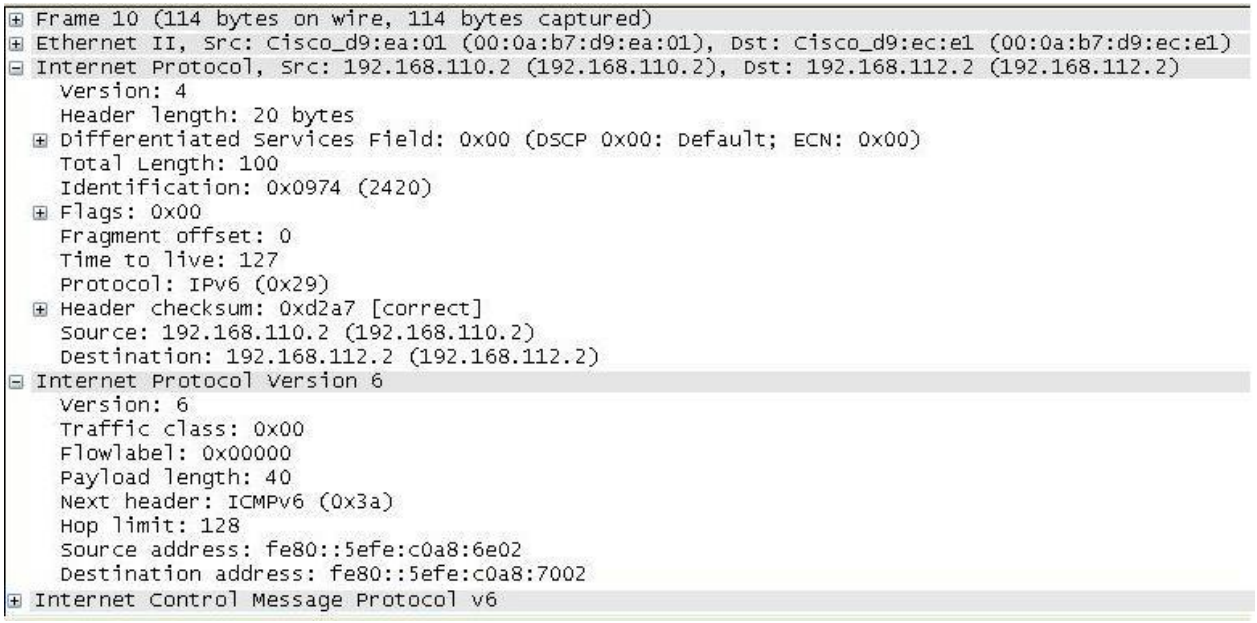

Figura 1.2 – Pacote capturado no PC 3 (echo request)

Echo reply:

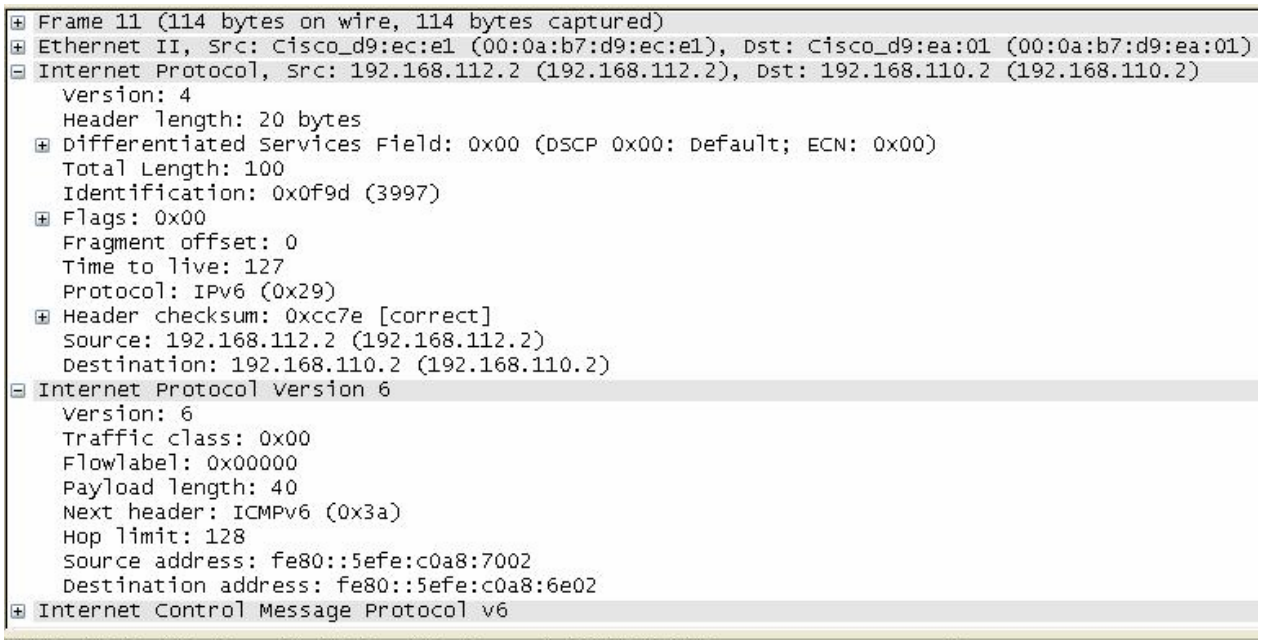

Figura 1.3 – Pacote capturado no PC 3 (echo reply)

Observações

Observando os pacotes capturados é possível ver que o cabeçalho IPv6 é encapsulado dentro do cabeçalho IPv4. Entre os routers é usado o protocolo IPv4 para o encaminhamento. Após a chegada do pacote ao Router é analisado o cabeçalho IPv6 para que se possa encaminhar o pacote para a máquina certa. È analisado o endereço IPv4 embutido no endereço ISATAP e caso exista uma máquina com esse IP, o pacote é enviado para essa máquina.

Nota: apesar de neste cenário se usarem routers os endereços ISATAP devem ser usado numa intranet! Para que se possa fazer encaminhamento para uma rede exterior é necessário o uso de endereços ISATAP globais!

2. Túneis Manuais

Os túneis manualmente configurados são simples de usar e eficazes.
Nestes túneis os endereços IPv6 e IPv4 são manualmente configurados (atribuídos) nas *interface*s dos routers destino e origem.
Este tipo de configuração é usada quando se pretende uma conectividade permanente ou dedicada e requer configuração específica em cada extremidade do túnel. Como é necessária configuração nos routers em cada ponta do túnel requer um bom conhecimento acerca da topologia da rede. Este tipo de configuração ponto a ponto fornece uma segurança bastante robusta a tráfego injectado no troço.

Cenário 1

Resumo

Cenário simples em que se implementa um túnel manual entre dois Routers.

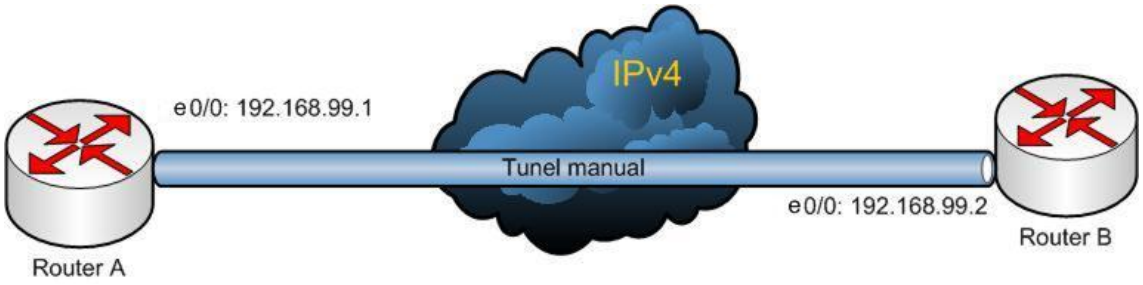

Figura 2.1 – Cenário de implementação de um túnel manual

Configuração Do Router A

```
interface ethernet 0
ip address 192.168.99.1 255.255.255.0

interface tunnel 0
ipv6 address 3ffe:b00:c18:1::3/127
tunnel source Fastethernet 0/0
tunnel destination 192.168.99.2
tunnel mode ipv6ip
```

Configuração do Router B

```
interface ethernet 0
ip address 192.168.99.2 255.255.255.0

interface tunnel 0
ipv6 address 3ffe:b00:c18:1::2/127
tunnel source ethernet 0/0
tunnel destination 192.168.99.1
tunnel mode ipv6ip
```

Observações

Os túneis manuais são muito simples de implementar, a grande desvantagem deste tipo de túnel é o facto de ter de se saber e configurar o endereço IP do Router de destino. Esta característica pode limitar seriamente a escalabilidade de uma rede.

3. Túneis 6to4

O objectivo dos túneis 6to4 é o de permitir a comunicação de vários domínios IPv6 (ilhas IPv6) sobre um *Backbone* IPv4 sem ser necessário configurar o túnel nos routers de fronteira situados nesses domínios IPv6. Desta forma, este tipo de túneis foram desenvolvidos para permitir que os routers de fronteira descubram automaticamente o endereço IPv4 da outra extremidade do túnel.
Os túneis 6to4 estabelecem o endereço das extremidades do túnel com base no endereço IPv4. O principal objectivo e propósito dos túneis 6to4 é o de permitir a comunicação entre máquinas IPv6 sem a configuração explícita de túneis, usando para isso *relay router*s.

Cenário 1

Resumo

Cenário simples onde é implementado um túnel 6to4 entre dois routers para que se possa encaminhar tráfego IPv6

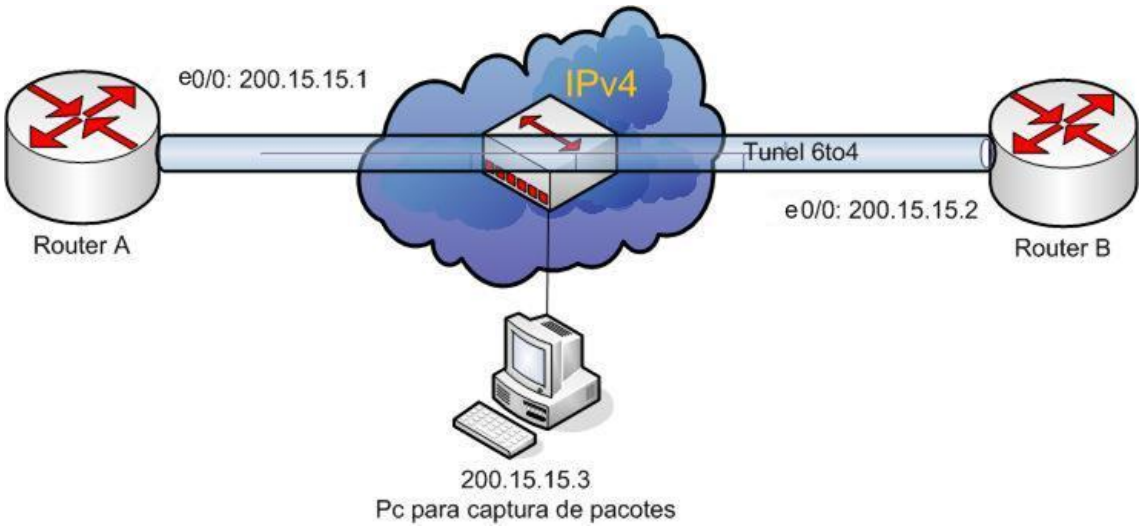

Figura 3.1 – Cenário de implementação de um túnel 6to4

Configurações do Router A

```
interface Ethernet0
description IPv4 uplink
ip address 200.15.15.1 255.255.255.0

interface Tunnel0
description IPv6 uplink
no ip address
ipv6 address 2002:C80F:F01::1/128
tunnel source Ethernet 0
tunnel mode ipv6ip 6to4

ipv6 route 2002::/16 tunnel 0
```

Configurações do Router B

```
interface Ethernet0
description IPv4 uplink
ip address 200.15.15.2 255.255.255.0

interface Tunnel0
description IPv6 uplink
no ip address
ipv6 address 2002:C80F:F02::1/128
tunnel source Ethernet 0
tunnel mode ipv6ip 6to4

ipv6 route 2002::/16 tunnel 0
```

Notas: De salientar que neste tipo de túneis não é especificado o destino do túnel (Tunnel Destination). Este é determinado com base no endereço IPv4 de destino uma vez que à saída do Router o encaminhamento é efectuado em IPv4.

Ping do router A para o B:

routerA# ping 2002:C80F:F02::1

```
⊞ Frame 7 (134 bytes on wire, 134 bytes captured)
⊞ Ethernet II, Src: Cisco_7d:f8:40 (00:13:19:7d:f8:40), Dst: Cisco_59:ef:80 (00:13:19:59:ef:80)
⊟ Internet Protocol, Src: 200.15.15.1 (200.15.15.1), Dst: 200.15.15.2 (200.15.15.2)
    Version: 4
    Header length: 20 bytes
  ⊞ Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
    Total Length: 120
    Identification: 0x0025 (37)
  ⊞ Flags: 0x00
    Fragment offset: 0
    Time to live: 255
    Protocol: IPv6 (0x29)
  ⊞ Header checksum: 0x0d16 [correct]
    Source: 200.15.15.1 (200.15.15.1)
    Destination: 200.15.15.2 (200.15.15.2)
⊟ Internet Protocol Version 6
    Version: 6
    Traffic class: 0x00
    Flowlabel: 0x00000
    Payload length: 60
    Next header: ICMPv6 (0x3a)
    Hop limit: 64
    Source address: 2002:c80f:f01::1
    Destination address: 2002:c80f:f02::1
⊟ Internet Control Message Protocol v6
    Type: 128 (Echo request)
    Code: 0
    Checksum: 0xf82e [correct]
    ID: 0x0c8b
    Sequence: 0x0000
    Data (52 bytes)
```

Figura 3.2 – Pacote capturado entre os dois routers

Observações

Através do pacote capturado entre os dois routers é possível ver o formato dos endereços 6to4. Também aqui o encaminhamento entre os routers é efectuado em IPv4 e só no Router é que o cabeçalho IPv6 é analisado.

Cenário 2

Resumo

Foram acrescentados PC´s de modo a simular intranets. Foram usados exclusivamente endereços públicos na atribuição de endereços ás máquinas.

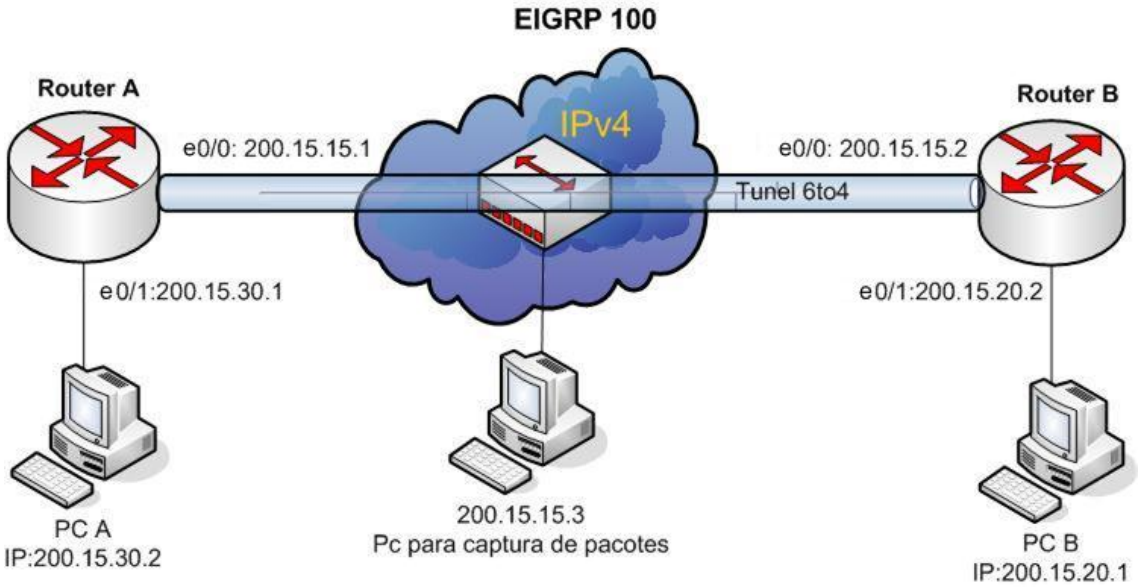

Figura 3.3 – Cenário com túnel 6to4 e PCs

Configurações do Router A

```
interface Ethernet0/0
        description IPv4 uplink
        ip address 200.15.15.1 255.255.255.0
interface Ethernet0/1
        description IPv4 uplink
        Ip address 200.15.35.1 255.255.255.0

interface Tunnel0
        description IPv6 uplink
        no ip address
        ipv6 address 2002:C80F:F01::1/128
        tunnel source Ethernet 0
        tunnel mode ipv6ip 6to4

ipv6 route 2002::/16 tunnel 0

router EIGRP 100
        network 200.15.15.0
        network 200.15.30.0
```

Configurações do Router B

```
interface Ethernet0
        description IPv4 uplink
        ip address 200.15.15.2 255.255.255.0

interface Ethernet0/1
        description IPv4 uplink
        ip address 200.15.20.2 255.255.255.0
```

```
interface Tunnel0
        description IPv6 uplink
        no ip address
        ipv6 address 2002:C80F:F02::1/128
        tunnel source Ethernet 0
        tunnel mode ipv6ip 6to4

ipv6 route 2002::/16 tunnel 0

router EIGRP 100
        network 200.15.15.0
        network 200.15.20.0
```

Ping do router A ao pc B:

routerA# ping 2002:C80F:1401::C80f:1401

Pacote capturado no PC ligado ao hub:

```
⊞ Frame 12 (134 bytes on wire, 134 bytes captured)
⊞ Ethernet II, Src: Cisco_7d:f8:40 (00:13:19:7d:f8:40), Dst: Cisco_59:ef:80 (00:13:19:59:ef:80)
⊟ Internet Protocol, Src: 200.15.15.1 (200.15.15.1), Dst: 200.15.20.1 (200.15.20.1)
    Version: 4
    Header length: 20 bytes
  ⊞ Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
    Total Length: 120
    Identification: 0x0048 (72)
  ⊞ Flags: 0x00
    Fragment offset: 0
    Time to live: 255
    Protocol: IPv6 (0x29)
  ⊞ Header checksum: 0x07f4 [correct]
    Source: 200.15.15.1 (200.15.15.1)
    Destination: 200.15.20.1 (200.15.20.1)
⊟ Internet Protocol Version 6
    Version: 6
    Traffic class: 0x00
    Flowlabel: 0x00000
    Payload length: 60
    Next header: ICMPv6 (0x3a)
    Hop limit: 64
    Source address: 2002:c80f:f01::1
    Destination address: 2002:c80f:1401::c80f:1401
⊟ Internet Control Message Protocol v6
    Type: 128 (Echo request)
    Code: 0
    Checksum: 0x983f [correct]
    ID: 0x22ff
    Sequence: 0x0004
    Data (52 bytes)
```

Figura 3.4 – Pacote capturado no PC do Hub

Pacote capturado no pc B:

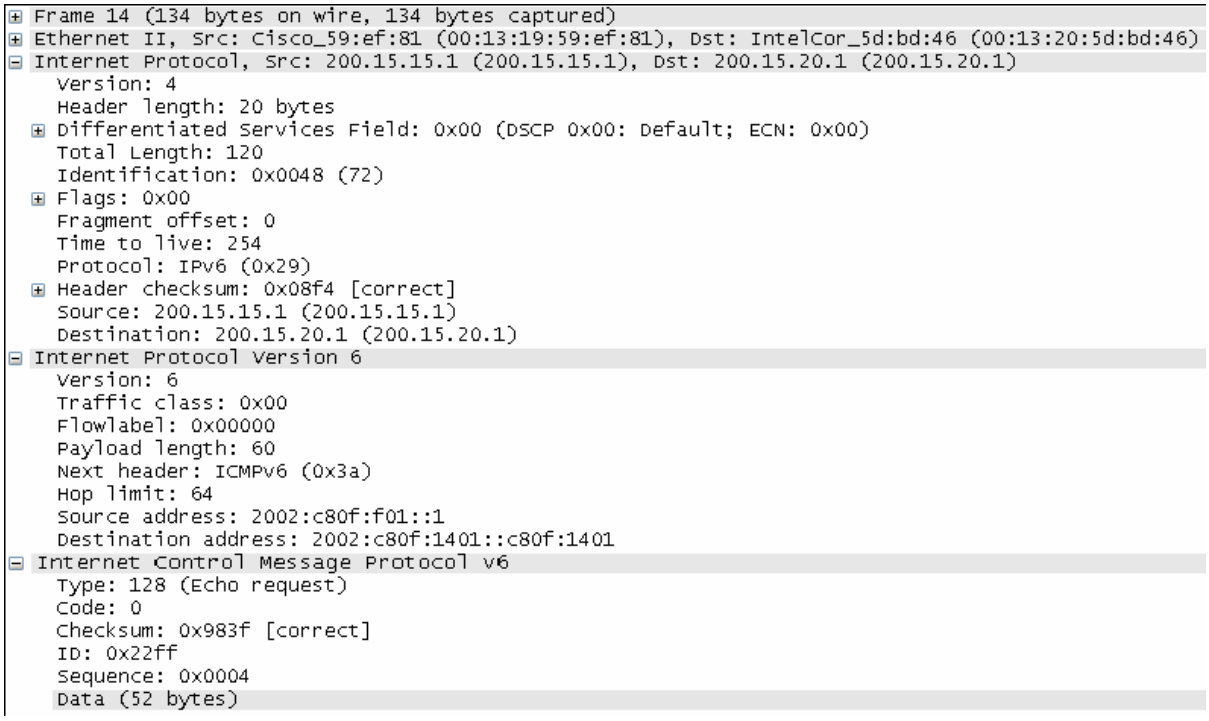

Figura 3.5 – Pacote capturado no PC B

Observações

Os pacotes têm o mesmo formato, a parte IPv6 vai encapsulada após a parte IPv4. O router B ao receber o pacote vai verificar se existe alguma máquina com o endereço IPv6 referenciado.

Cenário 3

Resumo

6to4 com ligação ao Backbone IPv6 (IPv6 nativo).

Este cenário pretende simular o que acontece quando um anfitrião algures na Internet se liga a um Router 6to4 para obter conexão ao Backbone Ipv6.

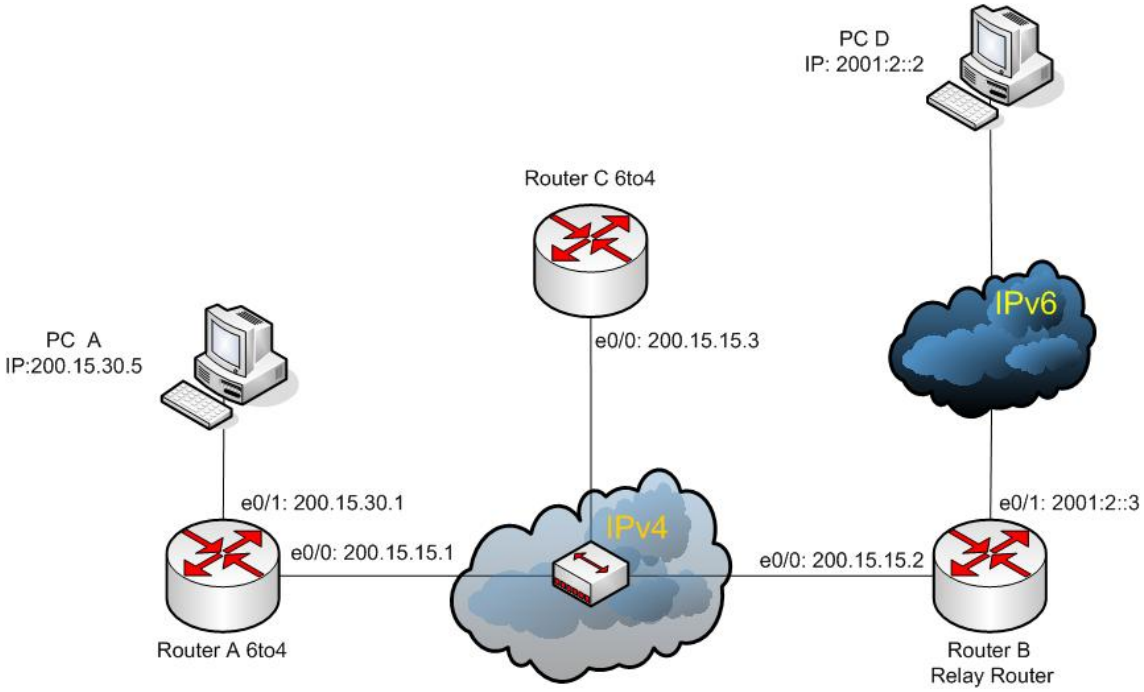

Figura 3.6 – Cenário com um túnel 6to4 e uma ligação a uma rede IPv6 nativa

Configurações do Router A (6to4)

```
interface Ethernet0/0
        description IPv4 uplink
        ip address 200.15.15.1 255.255.255.0
interface Ethernet0/1
        description IPv4 uplink
        Ip address 200.15.35.1 255.255.255.0
interface Tunnel0
        description IPv6 uplink
        no ip address
        ipv6 address 2002:C80F:F01::1/128
        tunnel source Ethernet 0
        tunnel mode ipv6ip 6to4

ipv6 route 2002::/16 tunnel 0
ipv6 route 2001::/16 2002:C80F:F01::2/128 (ver nota 1)
```

Nota 1: esta rota serve para garantir a conectividade do router A ao backbone IPv6, caso não seja colocada, um pc ligado a este router teria conectividade na mesma ao backbone IPv6 uma vez que tem conhecimento do endereço IPv4 (200.15.15.2) do relay router.

Configurações do Router C

```
interface Ethernet0
        description IPv4 uplink
        ip address 200.15.15.3 255.255.255.0
interface Tunnel0
        description IPv6 uplink
        no ip address
        ipv6 address 2002:C80F:F03::1/128
        tunnel source Ethernet 0
        tunnel mode ipv6ip 6to4

ipv6 route 2002::/16 tunnel 0
ipv6 route 2001::/16 2002:C80F:F01::2/128  (ver nota 2)
```

Nota 2: esta rota serve para garantir a conectividade do router C ao backbone IPv6, caso não seja colocada, um PC ligado a este router teria conectividade na mesma ao backbone IPv6 uma vez que tem conhecimento do endereço IPv4 (200.15.15.2) do relay router.

Configurações do Router B (relay router)

```
ipv6 unicast-routing (ver nota 3)
interface Ethernet0/0
        description IPv4 uplink
        ip address 200.15.15.2 255.255.255.0
interface FastEthernet0/1
         ip address 200.15.20.2 255.255.255.0
         ipv6 address 2001:2::3/128
         speed auto
interface Tunnel0
        description 6to4 relay
        no ip address
        no ip redirects
        ipv6 address 2002:C80F:F02::1/128
        tunnel source Ethernet 0/0
        tunnel mode ipv6ip 6to4
ipv6 route 2002::/16 tunnel 0
```

Nota 3: sem este comando o Router não encaminharia tráfego IPv6. Este comando permite o envio de "Router Advertisements", ou seja anuncia as máquinas os endereços 6to4 a usar assim como a existência do Router como Relay Router. Com este comando a máquina Windows é configurada automaticamente com uma rota para este Router.

Configurações do PC D (Windows XP)

- **Digitar na linha de comandos:**

```
>IPv6 adu 4/2001:2::2

Ou

>netsh interface IPv6 add address 4 2001:2::2
```

Adiciona um endereço global IPv6 ao PC (de modo a similar o Backbone IPv6).

```
>IPv6 rtu ::/0 4/2001:2::3
```

Adiciona uma rota do PC ao Router de saída, basicamente especifica o Gateway IPv6 que o PC D irá usar para enviar tráfego IPv6.

Configurações no PC A (Windows XP)

O PC A recebe um anúncio do Relay Router indicando ao PC o endereço do Relay Router, ou seja o PC A sabe automaticamente para que endereço enviar tráfego IPv6 que tenha como destino o Backbone IPv6.

É possível ver a rota criada para o Relay Router através do comando:

```
>netsh interface IPv6 show route
```

Observações

O cenário implementado poderia simular o que se passa actualmente na Internet. O Router A seria um utilizador algures no mundo que procura ligação ao Backbone IPv6.

No caso das máquinas Windows XP o sistema operativo configura automaticamente um Relay Router (no caso do Windows XP é usado o 6to4.ipv6.microsoft.com por defeito e caso exista uma ligação à Internet). Este Relay Router é que atribui automaticamente o endereço 6to4 que as máquinas Windows usam. È possível no entanto alterar este Relay Router com o comando:

```
>netsh interface IPv6 6to4 set relay <IP do relay router>
```

Para o caso do cenário seria:

```
>netsh interface IPv6 6to4 set relay 200.15.15.2
```

4. 6to4 + ISATAP

Para pôr máquinas de uma intranet a comunicar com o Backbone IPv6 uma das soluções encontradas é o uso de túneis ISATAP juntamente com túneis 6to4. O túnel ISATAP garante a conectividade do PC ao Gateway (router Cisco) e o túnel 6to4 garante a conectividade ao Backbone IPv6.

Cenário

Resumo

Com este cenário pretende-se simular a conectividade entre a rede nativa IPv6 e uma qualquer rede Intranet. Neste cenário uma vez que existem endereços públicos e privados, foi necessário o uso de dois tipos diferentes de túneis, os túneis ISATAP e os túneis 6to4.

Em relação à rede da ESTG esta poderia ser uma solução que permitisse a conexão dos PC´s da intranet da escola ao Backbone da IPv6. Não é no entanto uma solução simples e rápida de implementar necessitando de algumas configurações nem sempre simples.

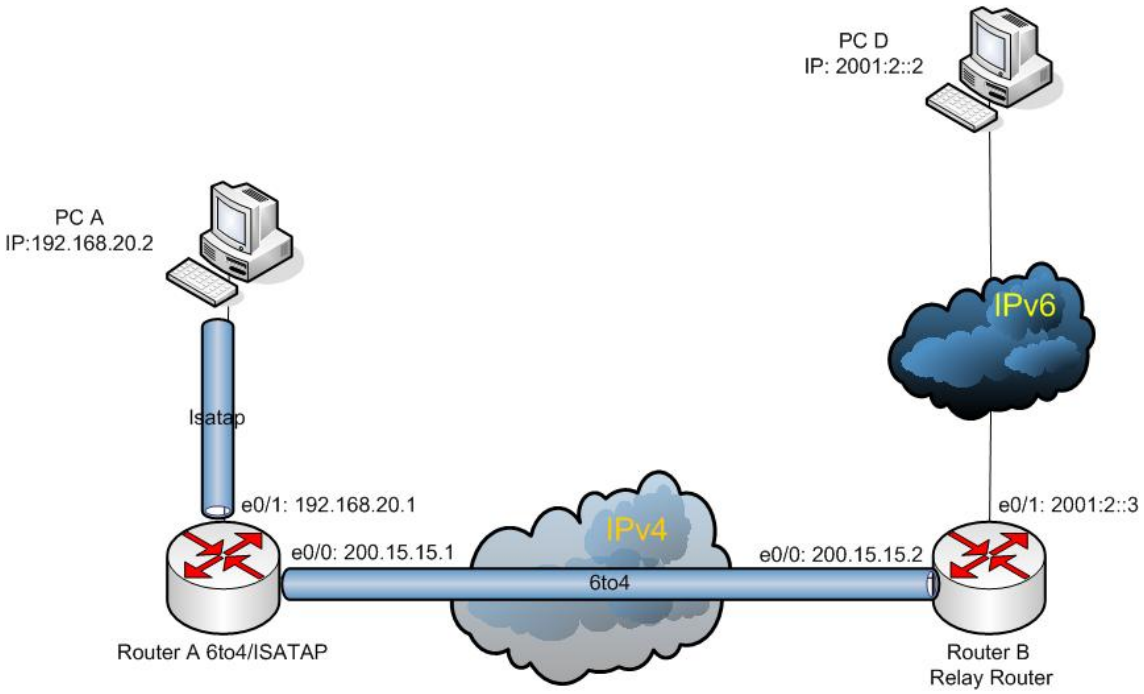

Figura 4.1 – Cenário com um túnel ISATAP e um túnel 6to4

Configurações do Router A (6to4 / ISATAP)

```
ipv6 unicast-routing

interface Tunnel0  (6to4)
 no ip address
 no ip redirects
```

```
 ipv6 address 2002:C80F:F01::1/64
 no ipv6 redirects
 tunnel source FastEthernet0/0
 tunnel mode ipv6ip 6to4

interface Tunnel1 (ISATAP)
 no ip address
 no ip redirects
 ipv6 address 2002:C832:3201::/64 eui-64
 no ipv6 nd suppress-ra
 tunnel source FastEthernet0/1
 tunnel mode ipv6ip isatap

interface FastEthernet0/0
 ip address 200.15.15.1 255.255.255.0
 duplex auto
 speed auto

interface FastEthernet0/1
 ip address 192.168.20.1 255.255.255.0
 duplex auto
 speed auto

router rip
 network 192.168.20.0
 network 200.15.15.0

ipv6 route 2002:C80F:F01::/48 Tunnel1
ipv6 route 2002::/16 Tunnel0
```

Configurações do Router B

```
ipv6 unicast-routing
interface Ethernet 0/0
        description IPv4 uplink
        ip address 200.15.15.2 255.255.255.0
interface Ethernet 0/1
        ip address 200.15.20.2 255.255.255.0
        ipv6 address 2001:2::3/128
        speed auto
interface tunnel 0
        description 6to4 relay
        no ip address
        no ip redirects
        ipv6 address 2002:C80F:F02::1/128
        tunnel source Ethernet 0/0
        tunnel mode ipv6ip 6to4
ipv6 route 2002::/16 tunnel 0
```

Configurações do PC A (Windows XP)

No PC A é necessário acrescentar um Gateway à interface ISATAP, neste caso o Gateway será a interface Ethernet 0/1 do Router A. O endereço que será inserido no PC será o IP da Ethernet 0/1 no formato ISATAP, ou seja, o IP: 192.168.20.1 será inserido através do comando:

```
>Ipv6 rtu ::/0 2/2002:c832:3201::5efe:192.168.20.1
```

Tratando-se de um PC de uma intranet com endereçamento privado, é necessário encontrar uma forma de identificar a máquina de modo único.

A solução encontrada foi a de criar um endereço global ISATAP que junte o prefixo 6to4 usado no túnel 6to4 (deriva de um endereço publico logo é único) com o endereço privado da máquina e o Id. da interface.

Este endereço é configurado através do seguinte comando:

```
>Ipv6 adu 2/2002:c80f:f01:2:0:5efe:192.168.20.2
```

Onde:

2002:c80f:f01: prefixo 6to4 (tunnel 0)

2: interface ISATAP do Windows

0:5EFE: indica que se trata de um endereço ISATAP

192.168.20.2: endereço privado do pc

Resumindo:

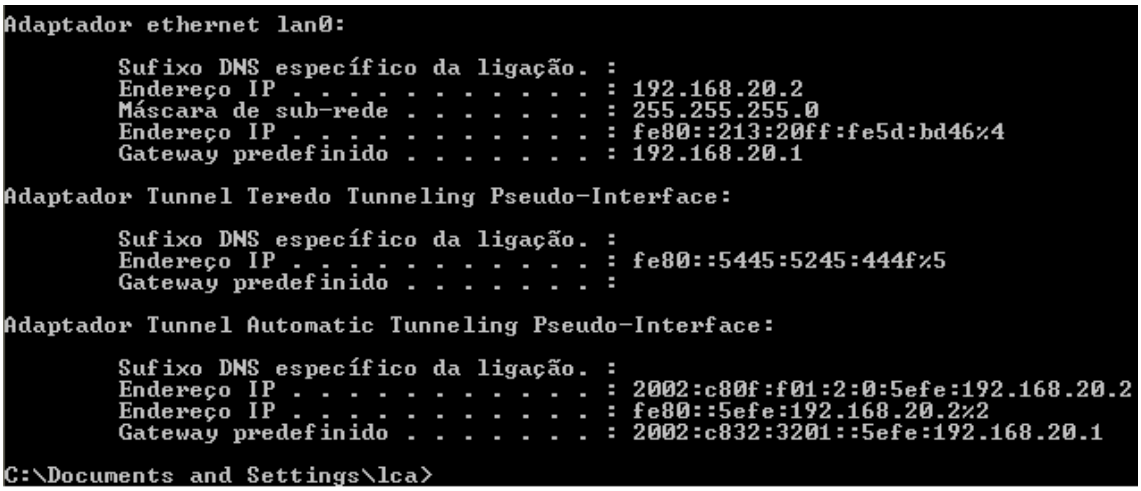

Figura 4.2 – Resultado do comando ipconfig no Windows

Deste modo o PC fica com um endereço único que permite a comunicação com a rede exterior à intranet! De notar que o endereço ISATAP Link-local também fica presente mas para encaminhamento para o exterior da intranet é o endereço global ISATAP que é usado.

Configurações do PC D (Windows XP)

Linha de comandos:

```
>IPv6 adu 4/2001:2::2

Ou

>netsh interface IPv6 add address 4 2001:2::2
```

Adiciona um endereço global IPv6 ao PC (de modo a similar o Backbone IPv6).

```
>IPv6 rtu ::/0 4/2001:2::3
```

Adiciona uma rota do PC ao Router de saída, basicamente especifica o Gateway IPv6 que o PC D irá usar para enviar tráfego IPv6.

Ping do PC_A para o PC_D:

Echo request

```
⊞ Frame 230 (114 bytes on wire, 114 bytes captured)
⊞ Ethernet II, Src: Cisco_7d:f8:40 (00:13:19:7d:f8:40), Dst: Cisco_59:ef:80 (00:13:19:59:ef:80)
⊟ Internet Protocol, Src: 200.15.15.1 (200.15.15.1), Dst: 200.15.15.2 (200.15.15.2)
    Version: 4
    Header length: 20 bytes
  ⊞ Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
    Total Length: 100
    Identification: 0x001c (28)
  ⊞ Flags: 0x00
    Fragment offset: 0
    Time to live: 255
    Protocol: IPv6 (0x29)
  ⊞ Header checksum: 0x0d33 [correct]
    Source: 200.15.15.1 (200.15.15.1)
    Destination: 200.15.15.2 (200.15.15.2)
⊟ Internet Protocol Version 6
    Version: 6
    Traffic class: 0x00
    Flowlabel: 0x00000
    Payload length: 40
    Next header: ICMPv6 (0x3a)
    Hop limit: 127
    Source address: 2002:c80f:f01:2:0:5efe:c0a8:1402
    Destination address: 2001:2::2
⊞ Internet Control Message Protocol v6
```

Figura 4.3 – pacote capturado (Echo request)

Echo reply

```
⊞ Frame 231 (114 bytes on wire, 114 bytes captured)
⊞ Ethernet II, Src: Cisco_59:ef:80 (00:13:19:59:ef:80), Dst: Cisco_7d:f8:40 (00:13:19:7d:f8:40)
⊟ Internet Protocol, Src: 200.15.15.2 (200.15.15.2), Dst: 200.15.15.1 (200.15.15.1)
    Version: 4
    Header length: 20 bytes
  ⊞ Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
    Total Length: 100
    Identification: 0x0022 (34)
  ⊞ Flags: 0x00
    Fragment offset: 0
    Time to live: 255
    Protocol: IPv6 (0x29)
  ⊞ Header checksum: 0x0d2d [correct]
    Source: 200.15.15.2 (200.15.15.2)
    Destination: 200.15.15.1 (200.15.15.1)
⊟ Internet Protocol Version 6
    Version: 6
    Traffic class: 0x00
    Flowlabel: 0x00000
    Payload length: 40
    Next header: ICMPv6 (0x3a)
    Hop limit: 63
    Source address: 2001:2::2
    Destination address: 2002:c80f:f01:2:0:5efe:c0a8:1402
⊞ Internet Control Message Protocol v6
```

Figura 4.4 – Pacote capturado (Echo reply)

Ping do PC_D para o PC_A:

Echo request

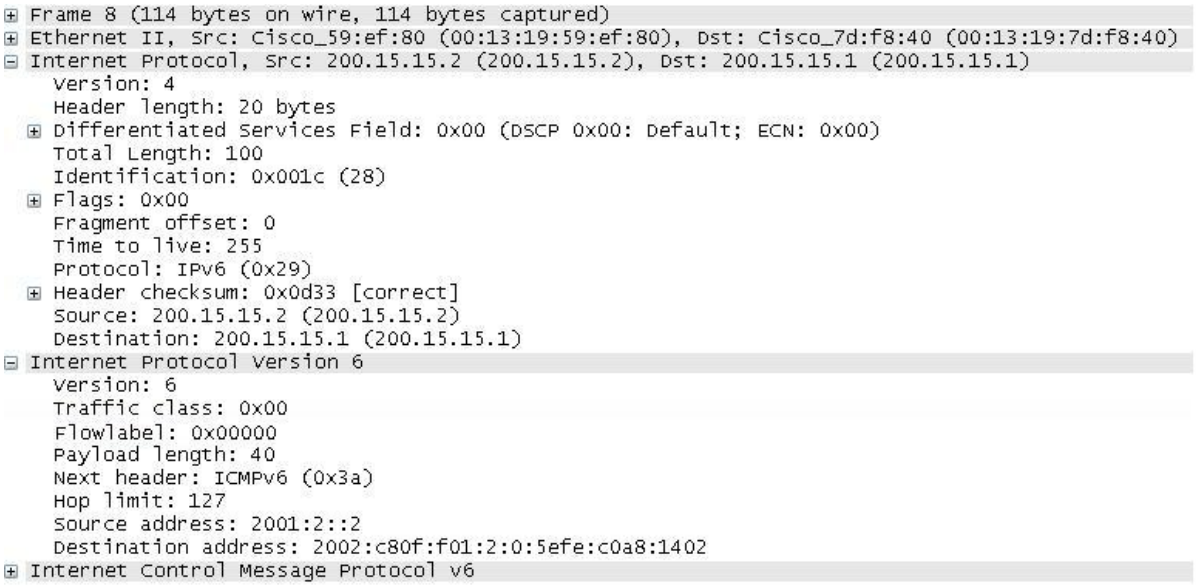

Figura 4.5 – Pacote capturado (Echo request)

Echo reply

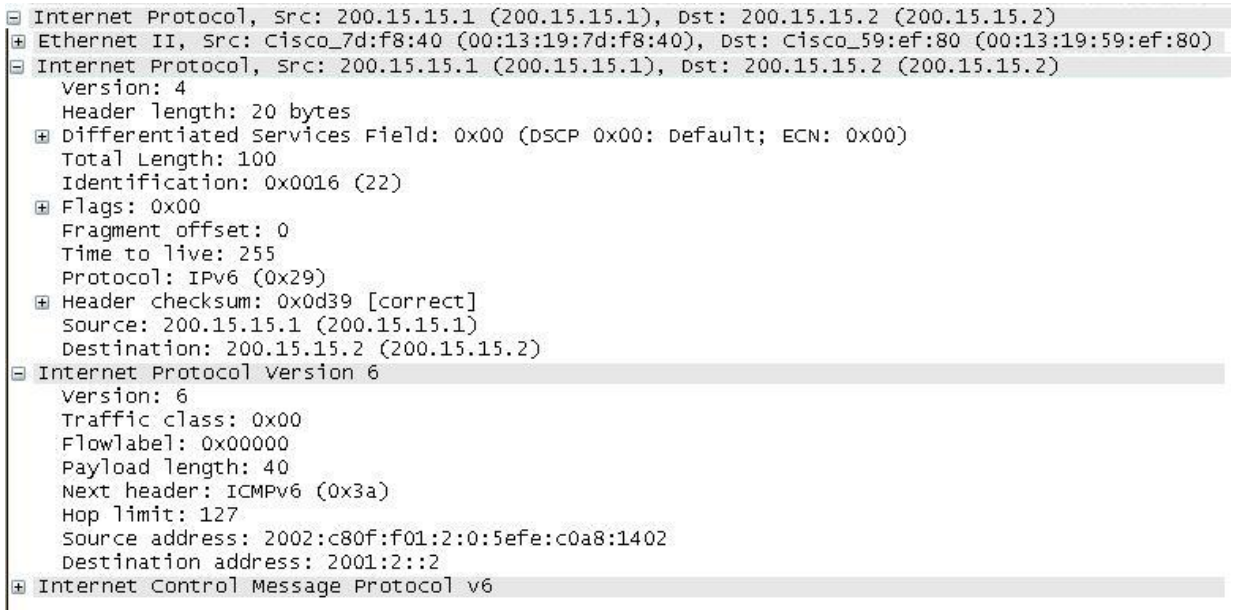

Figura 4.6 – Pacote capturado (Echo reply)

Obervações

Neste cenário existem dois Túneis: um ISATAP e um 6to4.

O túnel ISATAP garante a conexão entre o PC e o router A através do uso de endereços globais ISATAP, a partir do momento que os pacotes chegam ao router A estes são encaminhados pelo túnel 6to4. O inverso também ocorre, ou

seja, os pacotes enviados para o router A com o PC A como destino são encaminhados de um router para o outro pelo túnel 6to4 e ao chegarem ao router B são encaminhados pelo túnel ISATAP até ao PC D.

5. Rede ESTG

Durante a realização deste projecto pôs-se a hipótese de aplicar o cenário anterior à rede nativa IPv6 da ESTG. Na figura seguinte vemos a topologia da rede da ESTG e a sua ligação para o exterior:

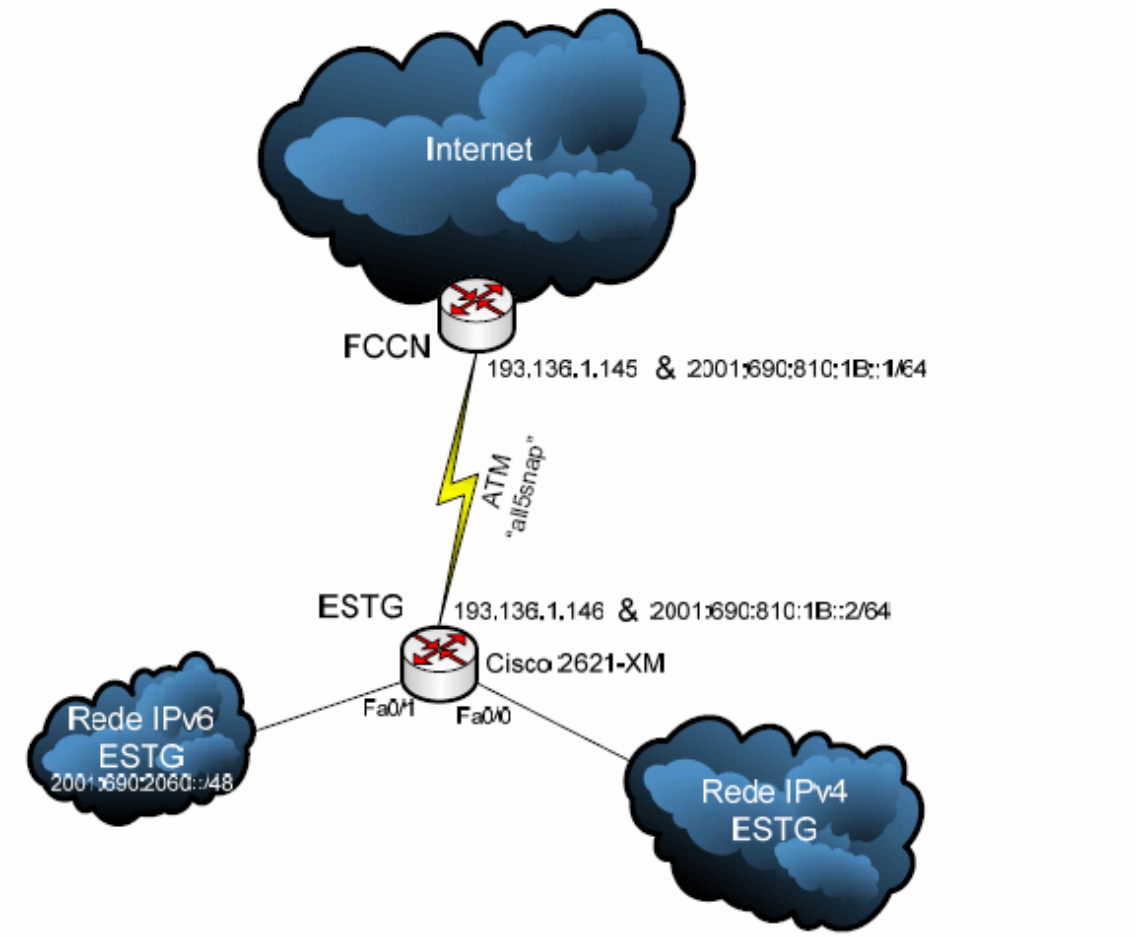

Figura 5.1 – Distribuição da rede IPv6 e IPv4 da ESTG-Leiria com a FCCN

A possibilidade que se colocava era a de ligar a rede IPv6 ESTG (rede IPv6 nativa) ao Relay router do cenário anterior.

A ESTG-Leiria possui um router IPv6-*only* ligado à rede FCCN. Na figura seguinte esse router está representado com o nome Router IPv6.

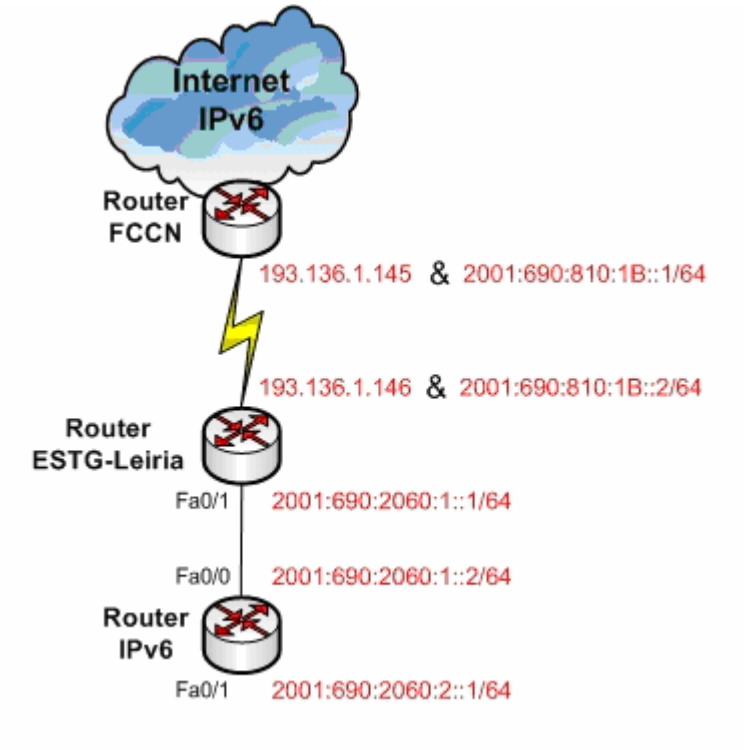

Figura 5.2 – Ligação da ESTG – Leiria à FCCN

O que se pensou fazer foi implementar a ponta do túnel 6to4 (Router B relay router) do cenário anterior no "Router IPv6" de modo a dar conectividade as máquinas da rede ESTG-Leiria ao *backbone* IPv6.

No entanto esta não é uma solução correcta uma vez que ao fazermos isso estaríamos a destruir a rede nativa IPv6, ou seja, para implementar o túnel teríamos de dar um IP versão 4 ao "Router IPv6" deitando o conceito "rede IPv6 nativa" por terra. A uma rede nativa IPv6 devem-se ligar máquinas com IP's (versão 6) globais. Ao fazer o contrário estaríamos a atrasar o processo de transição!

O que faria mais sentido seria ligar ao router 2621-XM (router ESTG Leiria onde ligam as máquinas IPv4 da rede ESTG - Figura 5.1) todos os PCs da escola através de um túnel ISATAP. A partir desse router, uma vez que já existe conectividade IPv6, o tráfego seria encaminhado normalmente até ao router FCCN.

6. NAT

O mecanismo NAT foi e é um mecanismo essencial nas redes existentes por esse mundo fora. Apesar de ser um mecanismo importante e essencial é odiado por muita gente pois é considerado limitador em alguns aspectos, nomeadamente o aspecto de ligações peer-to-peer. Apesar de ser um mecanismo bom para fornecer segurança às redes IPv4 não se pensa continuar a usar este mecanismo com o novo protocolo IPv6. Para o protocolo IPv6 as funções relacionados com segurança irão ficar a cargo do IPSec. Contudo, o NAT ainda existirá durante muitos anos e é importante perceber como se portará com a existência de novos endereços.

Cenário

Resumo

Pretendeu-se simular um cenário em que houvesse uma tradução de endereços entre endereços IPv4 e IPv6. De um lado da rede temos uma rede IPv4 e do outro uma rede IPv6. O objectivo é traduzir um endereço IPv6 para um endereço IPv4.

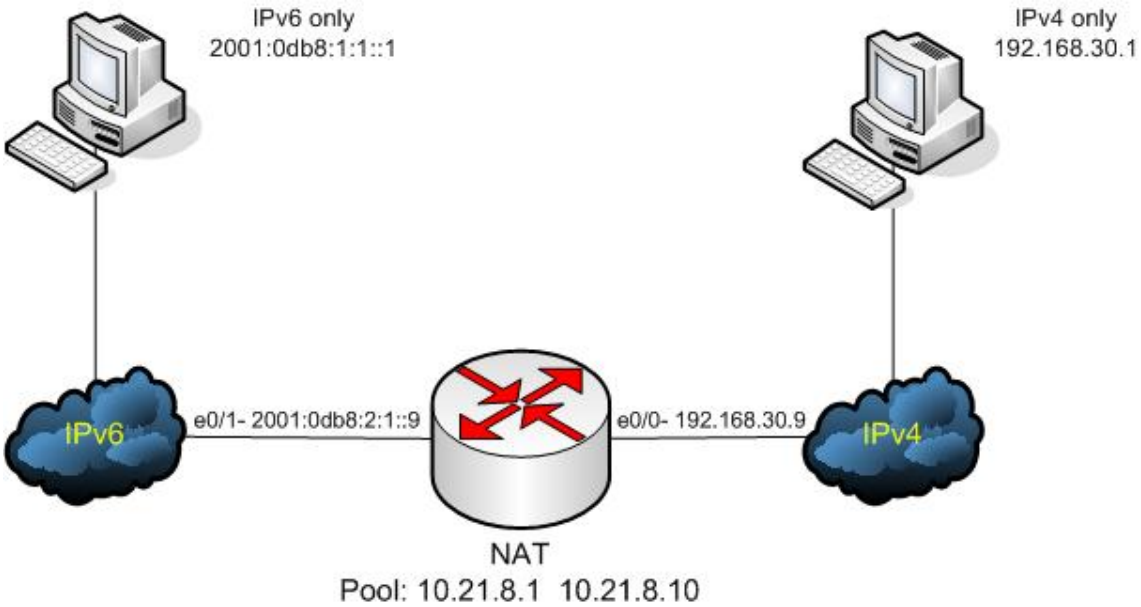

Figura 6.1 – Simulação de um dispositivo NAT

```
ipv6 nat prefix 2001:0db8::/96
```
Prefixo para o qual os endereços irão ser traduzidos.

```
interface ethernet 0/1
        ipv6 address 2001:0db8:2:1::9/64
        ipv6 nat
```

```
ipv6 nat prefix 2001::/96 v4-mapped v4map_acl
```
Para que o tráfego da rede IPv6 seja enviado para a rede IPv4 sem que seja preciso configurar mapeamento de endereço de destino.

```
interface ethernet 0/0
        ip address 192.168.30.9 255.255.255.0
        ipv6 nat

ipv6 nat v6v4 source list pt-list1 pool v4pool
```
Tradução de IPv6 para IPv4

```
ipv6 nat v6v4 pool v4pool 10.21.8.1 10.21.8.10 prefix-length 24

ipv6 nat v4v6 source list 1 pool v6pool
```
Tradução de IPv4 para IPv6

```
ipv6 nat v4v6 pool v6pool 2001:0db8:yyyy::1 2001:0db8:yyyy::2 prefix-length 128

ipv6 access-list v4map_acl permit ipv6 2001::/96 2000::/96
```

Observações:

Não conseguimos realizar com sucesso este cenário. Não conseguimos que o router fizesse a tradução correcta, logo não conseguimos pingar a máquina IPv4 a partir da máquina IPv6. Para alem disso também não conseguimos obter informação suficiente de como pingar uma máquina IPv4 a partir de uma máquina IPv6 only ou vice versa. De qualquer maneira o que tentámos foi pingar a máquina IPv4 através de um IP mapeado, ou seja, o ip 192.168.30.1 seria 2001:0db8::1E09.
1E =30
09=9
Segundo escassas informações encontradas na Internet (que até podem estar incorrectas) A parte 192.168 do endereço é implicitamente definida quando estamos a pingar máquinas IPv4 a partir de IPv6 puro! Apenas é necessário colocar o prefixo IPv6 (para que o pacote ICMPV6 possa sair da máquina) seguido dos últimos 16bits do endereço IPv4. Este foi um dos problemas com que nos deparámos, como pingar uma máquina IPv4 quando apenas se tem a pilha IPv6 instalada e vice-versa! Infelizmente não conseguimos encontrar informação acerca deste problema. Neste cenário apenas conseguimos pingar o Gateway da rede IPv6 através do Host IPv6 only.

7. DNS (A6 & AAAA)

Um *Domain Name System* (DNS) é necessário para uma coexistência fiável dos protocolo IPv4 e IPv6. O DNS é das ferramentas mais importantes da *Internet* actual e ganha uma importância ainda maior com a progressiva adopção do protocolo IPv6. O utilizador comum acede a uma página através de um nome e não através de um IP, mesmo que pensemos em pessoas especializadas em informática que saibam usar os endereços IP para aceder a recursos isso torna-se um pouco mais difícil com os novos endereços IPv6. A actualização da infra-estrutura DNS passa pela implementação nos servidores DNS de suporte para resolução de endereços IPv6 para nomes e vice-versa.

Cenário

Resumo

Usámos como servidor DNS a distribuição de Linux "Fedora Core 5". Esta escolha deve-se ao facto de no projecto "Instalação de uma rede piloto IPv6" ter sido usado esta distribuição e por ser uma distribuição com a qual estávamos mais familiarizados.

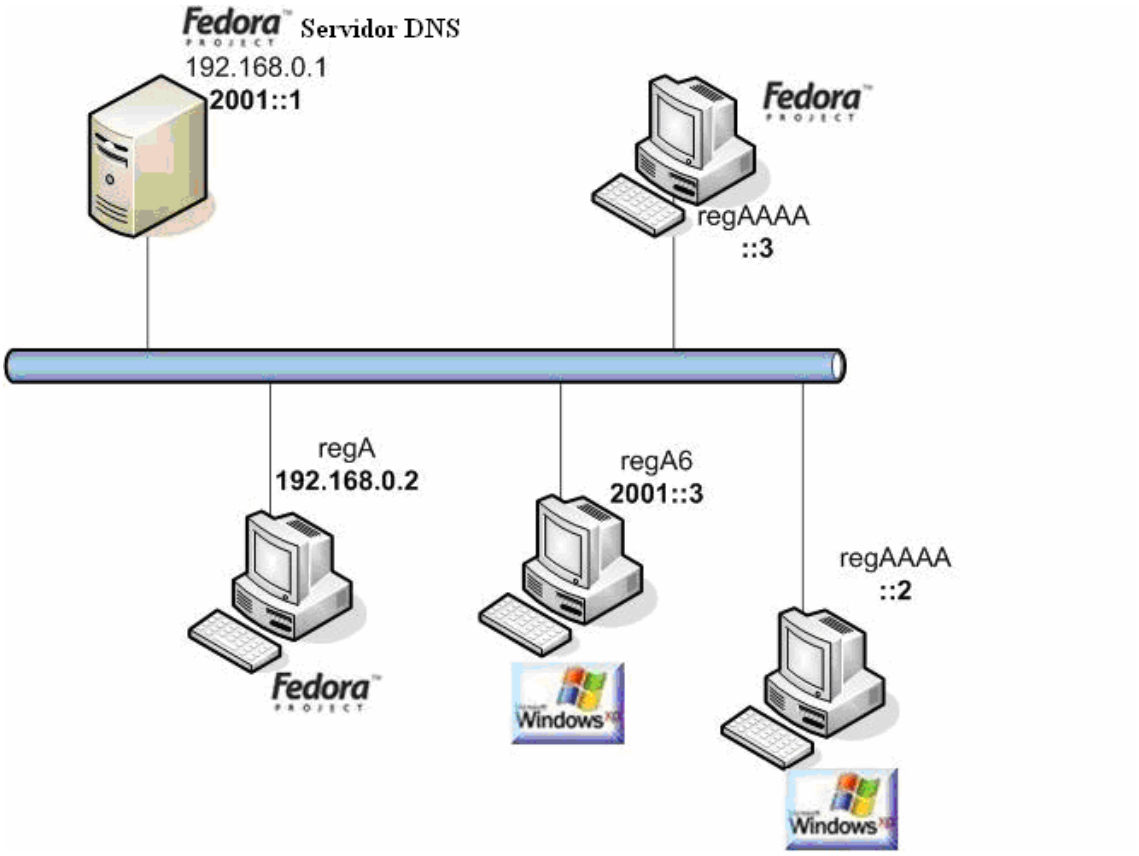

Figura 7.1 – Implementação de um servidor DNS com registos A, AAAA e A6

Nota: o servidor DNS tem dois endereços, um endereço IPv4 e outro IPv6. As máquinas Windows XP como têm pilha dupla vão consultar o servidor pelo

endereço IPv4, uma vez que na sua configuração apenas podemos configurar um endereço IPv4 para o servidor DNS.

Configurações

- **Instalar o Bind**

```
#yum install bind
```

- **Modificar o ficheiro de configuração /etc/resolv.conf**

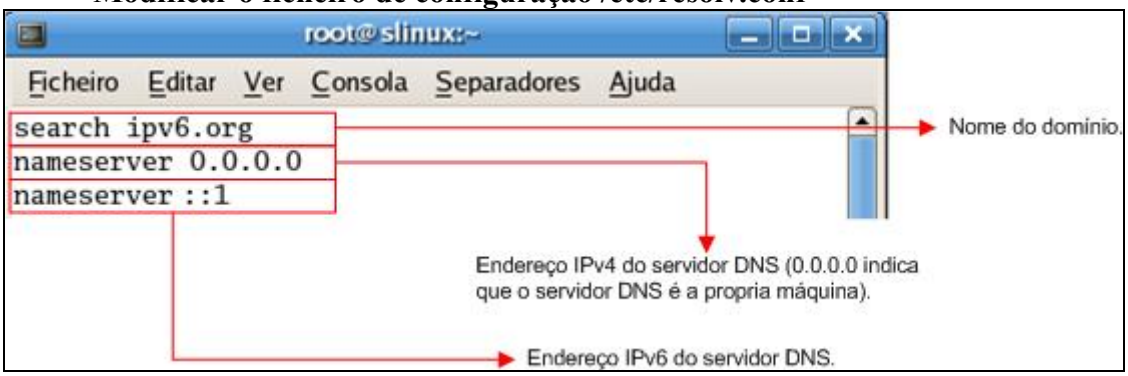

Figura 7.2 – Configuração do ficheiro de configuração do *resolver* DNS.

- **Modificar o ficheiro de configuração /etc/named.conf**

```
options {
        directory "/var/named";
        dump-file "/var/named/data/cache_dump.db";
        statistics-file "/var/named/data/named_stats.txt";
        listen-on-v6 { any; };
        /*
         * If there is a firewall between you and nameservers you want
         * to talk to, you might need to uncomment the query-source
         * directive below.  Previous versions of BIND always asked
         * questions using port 53, but BIND 8.1 uses an unprivileged
         * port by default.
         */
         // query-source address * port 53;
        forwarders { 192.168.234.3; 192.168.234.4; };
};
```

Activar o suporte IPv6 no servidor DNS

Figura 7.3 – Activar o suporte IPv6 no ficheiro de configuração do servidor DNS.

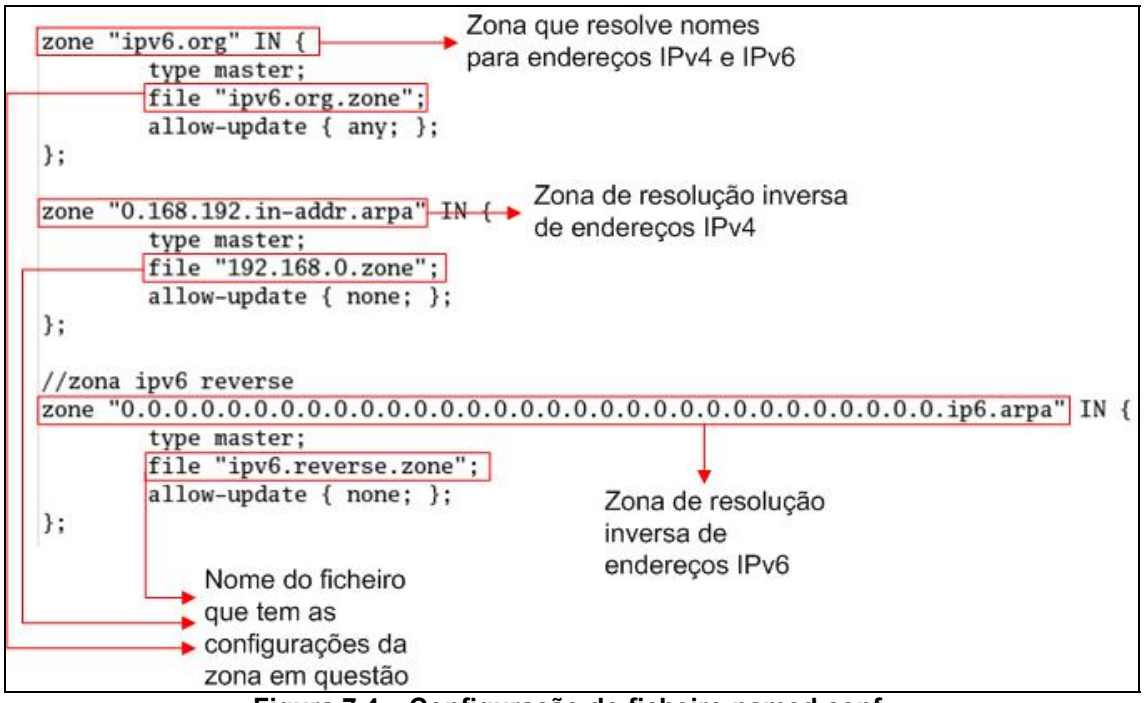

Figura 7.4 – Configuração do ficheiro named.conf.

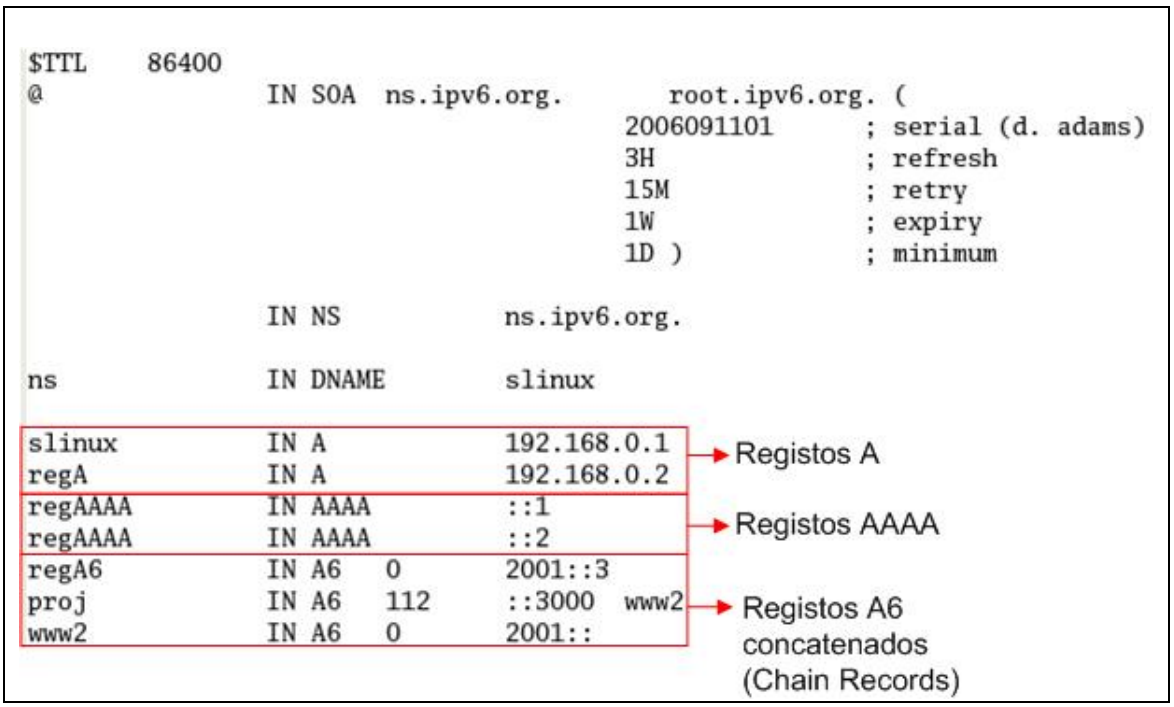

Figura 7.5 – Configuração do ficheiro "ipv6.org.zone"

- **Ficheiro "ipv6.reverse.zone"**

```
$TTL    86400
@       IN      SOA     ns.ipv6.org. root.ipv6.org.  (
                                2006091101 ; Serial
                                28800      ; Refresh
                                14400      ; Retry
                                3600000    ; Expire
                                86400 )    ; Minimum
        IN      NS      ns.ipv6.org.

0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.1      IN      PTR     regAAAA.ipv6.org.
0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.2      IN      PTR     regAAAA.ipv6.org.
0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.3      IN      PTR     regA6.ipv6.org.
```

Figura 7.6 – Configuração do ficheiro de resolução inversa de endereços IPv6.

Nota: Não foi possível colocar os endereços com o novo formato (*bit-string*) IP6.ARPA pois o serviço “named” da erro ao compilar ambos os ficheiros de configuração (“named.conf” e “ipv6.reverse.zone”). Assim concluímos que a *package* Bind 9.3.2 (servidor DNS) tem algumas incompatibilidades com formato *bit-string*.

- **Iniciar o servidor DNS**

#service named start
Ou
#/etc/init.d/named start

Nota: Se o serviço “named” já estiver a correr substitui-se o “start” por “restart”.

- **Activar o encaminhamento IPv6 do servidor**

#sysctl –w net.ipv6.conf.all.forwarding=1

Adicionar ao ficheiro /etc/sysctl.conf a seguinte linha de código para activar o encaminhamento IPv6 nas interfaces logo quando o sistema operativo inicia. Deste modo não será necessário escrever sempre na consola a linha acima mencionada.

```
# Controls IPv6 packet forwarding
net.ipv6.conf.all.forwarding = 1
```

Figura 7.7 – Activar o encaminhamento IPv6.

- **Activar suporte IPv6 na interface**

Adicionar ao ficheiro /etc/sysconfig/network-scripts/ifcfg-ethX, onde ‘X’ é a interface *ethernet* do servidor DNS.

```
IPV6INIT=yes
IPV6ADDR=2001::1/128
```

IPV6INIT=yes → Activa o suporte IPv6 na interface
IPV6ADDR=2001::1/128 → Atribui um endereço IPv6 à interface

Figura 7.8 – Configurações da interface ethernet.

- **Reiniciar o serviço Network**

#service network start
Ou
#/etc/init.d/network start

8. DHCPv6

O protocolo DHCP é uma ferramenta essencial para quem é responsável pela gestão de uma rede. Sem este protocolo cada utilizador teria de ter um endereço estático atribuído na sua máquina. Qualquer alteração de IP não programada poderia ter repercussões imprevisíveis no acesso a rede.
Em IPv6 existe um mecanismo de autoconfiguração (stateless address autoconfiguration), ou seja, é automaticamente atribuído um IP à máquina. Esta característica do protocolo IPv6 poderá fazer muita gente questionar qual a relevância do DHCPv6 no futuro.
No entanto existem casos em que o uso de DHCPv6 poderá ajudar num maior controlo sobre o endereçamento permitindo um maior controlo sobre a rede.

Configuração de um servidor DHCPv6:

Resumo

Usámos a distribuição de Linux "Fedora Core 5" para implementar um servidor de DHCPv6.
A configuração básica de um servidor DHCPv6 é bastante simples e parecida com a configuração de um servidor DHCP para IPv4.

Instalação do DHCPv6

```
#yum install dhcpv6
```

Configuração de um servidor DHCPv6:

- **Ficheiro /etc/dhcp6s.conf. (Servidor)**

```
interface eth0 {
        server-preference 255;
        renew-time 60;
        rebind-time 90;
        prefer-life-time 130;
        valid-life-time 200;
        allow rapid-commit;
        option dns_servers 2001:db8:0:f101::1 sub.domain.example;
        link AAA {
                range 2001:db8:0:f101::1000 to 2001:db8:0:f101::ffff/64;
                prefix 2001:db8:0:f101::/64;
        };
};
```

Nota: no range é especificada a gama de endereços e prefixos a atribuir

- **Ficheiro /etc/dhcp6c.conf. (Cliente)**

```
interface eth0 {
        send rapid-commit;
        request domain-name-servers;
};
```

- **Iniciar o servidor**

```
# service dhcp6s start
```

- **Iniciar o cliente**

```
#  dhcp6c -f eth0
```

- **Debug (Servidor)**

```
# dhcp6s -d -D -f eth0
```

- **Debug (Cliente)**

```
# dhcp6c -d -f eth0
Oct/03/2005 17:18:16 dhcpv6 doesn't support hardware type 776
Oct/03/2005 17:18:16 doesn't support sit0 address family 0
Oct/03/2005 17:18:16 netlink_recv_rtgenmsg error
Oct/03/2005 17:18:16 netlink_recv_rtgenmsg error
Oct/03/2005 17:18:17 status code for this address is: success
Oct/03/2005 17:18:17 status code: success
Oct/03/2005 17:18:17 netlink_recv_rtgenmsg error
Oct/03/2005 17:18:17 netlink_recv_rtgenmsg error
Oct/03/2005 17:18:17 assigned address 2001:db8:0:f101::1002 prefix len
is not in any RAs prefix length using 64 bit instead
Oct/03/2005 17:18:17 renew time 60, rebind time 9
```

- **Ficheiro : /etc/sysconfig/network-scripts/ifcfg-eth0**

```
IPv6INIT=yes
DHCP6C=yes
```

9. Comandos úteis

Windows XP (Service Pack 2)

- **Activar a pilha ipv6**

```
>Ipv6 install

Ou

>netsh interface ipv6 install
```

- **Mostrar a informação de todas as interfaces existentes (IPv4 e IPv6)**

```
>ipconfig
```

- **Mostrar toda a informação de uma interface**

```
>ipv6 if [número da interface]
```

Exemplo:

```
>ipv6 if 2
```

- **Controlar os atributos de uma interface**

```
>ipv6 ifc <numero da interface> [forwards] [advertises]  [-forwards] [-advertises] [mtu <número de bytes>] [site <identificador do site>]
```

Forwards: encaminha os pacotes cujo endereço de destino não é o da interface em questão.

Advertises: faz com que a interface envie Router Advertisements

Mtu: define o mtu da interface. Este mtu deve ser igual ou superior a 1280 bytes.

Site: define o identificador do site da interface

Exemplo:

```
>ipv6 ifc 1 forwards
```

- **Apagar uma interface**

```
>Ipv6 ifd <número da interface>
```

NOTA: as interfaces loopback e tunnel pseudo-interfaces não podem ser eliminadas.

Exemplo:

>ipv6 ifd 1

- **Mostrar o conteúdo do Neighbor Cache**

>Ipv6 nc [<numero da interface> [endereço IPv6]]

Exemplo:

>Ipv6 nc

- **Apagar o conteúdo do Neighbor Cache**

>Ipv6 rcf [<numero da interface> [endereço IPv6]]

Exemplo:

>Ipv6 rcf

- **Mostrar o conteúdo da Route Cache**

>Ipv6 rc [<numero da interface> [endereço IPv6]]

Exemplo:

>Ipv6 rc

- **Apagar o conteúdo da Route Cache**

>Ipv6 ncf [<numero da interface> [endereço IPv6]]

Exemplo:

>Ipv6 rcf

- **Mostrar o conteúdo da Binding Cache**

>Ipv6 bc

- **Adicionar um endereço unicast ou anycast IPv6 numa interface**

>ipv6 adu <número da interface>/<endereço IPv6> [lifetime VL[/PL]] [anycast] [unicast]

Lifetime: tempo de vida do endereço em segundos. Se nada for especificado ou se for especificado o valor *infinite*, este endereço tem validade infinita. Se o tempo de vida for igual a 0, o endereço é eliminado.

VI: tempo de vida válido.

PI: tempo de vida preferível.

Ex:

>ipv6 adu 2/2002:836B:1:5:200:5EFE:10.40.1.29

- **Mostrar o conteúdo da tabela de prefixos**

```
>Ipv6 spt
```

- **Adicionar, remover ou actualizar a tabela de prefixos**

```
>ipv6 spu <prefixo> <número da interface> [lifetime <segundos>]
```

Lifetime: tempo de vida do endereço em segundos. Se nada for especificado ou se for especificado o valor *infinite*, este endereço tem validade infinita. Se o tempo de vida for igual a 0, o endereço é eliminado.

Exemplo:

>ipv6 spu 2002::/ 2

- **Mostrar o conteúdo da tabela de encaminhamento**

```
>ipv6 rt
```

- **Adicionar ou remover uma rota da tabela de encaminhamento**

```
>ipv6 rtu <prefixo> <numero da interface>[/nexthop] [lifetime L] [preference P]
[publish] [age] [spl site-prefix-length]
```

Nexthop: endereço vizinho de uma interface.

Lifetime: tempo de vida do endereço em segundos. Se nada for especificado ou se for especificado o valor *infinite*, este endereço tem validade infinita. Se o tempo de vida for igual a 0, o endereço é eliminado.

Publish: rota que será usada na construção do Router Advertisements.

Age: usado no Router Advertisements que indica que a rota poderá expirar.

Spl: especifica o tamanho do prefixo associado a rota. Este prefixo é usado somente ao enviar Router Advertisements.

Fedora Core 5

- **Verificar se a versão do kernel suporta IPv6 (a partir da versão 2.4 do kernel já é suportado):**

```
# test −f /proc/net/if_inet6 && echo "Running kernel is IPv6 ready"
```

- **Carregar o modulo IPv6:**

```
# modprobe ipv6
```

- **Carregar o módulo automaticamente no arranque (caso a versão do kernel não suporte IPv6):**

```
Adicionar ao ficheiro /etc/modprobe.conf:

#alias net−pf−10 ipv6
```

- **Activar a pilha IPv6**

```
Adicionar ao ficheiro /etc/sysconfig/network:

NETWORKING_IPV6=yes

Adicionar ao ficheiro /etc/sysconfig/network-scripts/ifcfg-eth0:

IPV6INIT=yes
```

- **Ligar/desligar interfaces**

```
# ip link set dev <interface> up

# ip link set dev <interface> down

Ou

# /sbin/ifconfig <interface> up

# /sbin/ifconfig <interface> down
```

Exemplos:

```
# ip link set dev eth0 up

# ip link set dev eth0 down

# /sbin/ifconfig eth0 up

# /sbin/ifconfig eth0 down
```

- **Visualizar as características das interfaces**

```
#ifconfig
```

- **Alterar definições das interfaces**

Editar o ficheiro /etc/radvd.conf

- **Adicionar rotas IPv6**

```
# /sbin/ip −6 route add <prefixo IPv6>/<tamanho do prefixo> dev <interface>
```

Ou

/sbin/route −A inet6 add <prefixo IPv6>/<tamanho do prefixo> dev <interface>

Exemplos:

/sbin/ip −6 route add 2000::/3 dev eth0 metric 1

/sbin/route −A inet6 add 2000::/3 dev eth0

- **Remover rotas IPv6**

/sbin/ip −6 route del < prefixo IPv6>/<tamanho do prefixo> dev <interface>

/sbin/route −A inet6 del < prefixo IPv6>/<tamanho do prefixo> dev <interface>

Exemplos:

/sbin/ip −6 route del 2000::/3 dev eth0

/sbin/route −A inet6 del 2000::/3 dev eth0

- **Adicionar endereços IPv6**

/sbin/ip −6 addr add <endereço IPv6>/<tamanho do prefixo> dev <interface>

Ou

/sbin/ifconfig <interface> inet6 add <endereço IPv6>/<tamanho do prefixo>

Exemplos:

/sbin/ip −6 addr add 3ffe:ffff:0:f101::1/64 dev eth0

/sbin/ifconfig eth0 inet6 add 3ffe:ffff:0:f101::1/64

- **Eliminar endereços IPv6**

/sbin/ip −6 addr del <endereço IPv6>/<tamanho do prefixo> dev <interface>

Ou

/sbin/ifconfig <interface> inet6 del <endereço IPv6>/<tamanho do prefixo>

Exemplos:

/sbin/ip −6 addr del 3ffe:ffff:0:f101::1/64 dev eth0

/sbin/ifconfig eth0 inet6 del 3ffe:ffff:0:f101::1/64

- **Mostrar rotas IPv6**

/sbin/ip −6 route show [dev <interface>]

Ou

/sbin/route −A inet6

Exemplos:

/sbin/ip −6 route show dev eth0

/sbin/route −A inet6 |grep −w "eth0"

- **Adicionar um gateway IPv6**

/sbin/ip −6 route add <prefixo IPv6>/<tamanho do prefixo> via <ipv6address> [dev <device>]

Ou

/sbin/route −A inet6 add <prefixo IPv6>/<tamanho do prefixo> gw <endereço IPv6> [dev <interface>]

Exemplos:

/sbin/ip −6 route add 2000::/3 via 3ffe:ffff:0:f101::1

/sbin/route −A inet6 add 2000::/3 gw 3ffe:ffff:0:f101::1

- **Eliminar um gateway IPv6**

/sbin/ip −6 route del <prefixo IPv6>/<tamanho do prefixo> via <endereço IPv6> [dev <interface>]

Ou

/sbin/route −A inet6 del <prefixo IPv6>/<tamanho do prefixo> [dev <interface>]

Exemplos:

/sbin/ip −6 route del 2000::/3 via 3ffe:ffff:0:f101::1

/sbin/route −A inet6 del 2000::/3 gw 3ffe:ffff:0:f101::1

- **Mostrar a vizinhança (Neighbor Cache)**

ip −6 neigh show [dev <interface>]

Exemplo:

ip −6 neigh show

- **Adicionar uma entrada na tabela de vizinhança**

```
# ip −6 neigh add <endereço IPv6> lladdr <endereço MAC> dev <interface>
```

Exemplo:

ip −6 neigh add fec0::1 lladdr 02:01:02:03:04:05 dev eth0

- **Remover uma entrada na tabela de vizinhança**

```
# ip −6 neigh del <endereço IPv6> lladdr <endereço MAC> dev <interface>
```

Exemplo:

ip −6 neigh del fec0::1 lladdr 02:01:02:03:04:05 dev eth0

- **Mostrar os tuneis existentes**

```
# /sbin/ip −6 tunnel show [<interface>]

Ou

# /sbin/route −A inet6
```

Exemplos:

/sbin/ip −6 tunnel show

/sbin/route −A inet6 | grep "\Wsit0\W*$"

- **Criação de um Tunel**

```
# /sbin/ip tunnel add <interface> mode sit ttl <ttl por defeito> remote <endereço IPv4 do tunnel vizinho> local <endereço IPv4 local>
```

Exemplo:

/sbin/ip tunnel add sit1 mode sit ttl <ttldefault> remote <ipv4addressofforeigntunnel1> local <ipv4addresslocal>

- **Remoção de um tunnel**

```
# /sbin/ip tunnel del <interface>
```

Exemplo:

/sbin/ip −6 route del <prefixo para rota> dev sit1

/sbin/ip link set sit1 down

/sbin/ip tunnel del sit1

10. Anexos

Túneis 6to4

Cenário 2

Router A

```
Current configuration : 1259 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname Router
!
boot-start-marker
boot-end-marker
!
resource policy
!
no aaa new-model
memory-size iomem 10
no network-clock-participate slot 1
no network-clock-participate wic 0
!
ip cef
!

interface Tunnel0
 no ip address
 no ip redirects
 ipv6 address 2002:C80F:F01::1/128
 tunnel source FastEthernet0/0
 tunnel mode ipv6ip 6to4
!
interface FastEthernet0/0
 ip address 200.15.15.1 255.255.255.0
 duplex auto
 speed auto
!
interface Serial0/0
 no ip address
 shutdown
 no fair-queue
 no dce-terminal-timing-enable
!
interface FastEthernet0/1
 ip address 200.15.30.1 255.255.255.0
duplex auto
 speed auto
!
interface Serial0/1
 no ip address
 shutdown
 no dce-terminal-timing-enable
!
interface Serial0/2
```

```
 no ip address
 shutdown
 no dce-terminal-timing-enable
!
interface Serial0/3
 no ip address
 shutdown
 no dce-terminal-timing-enable
!
router eigrp 100
 network 200.15.15.0
 network 200.15.30.0
 auto-summary
!
ip http server
no ip http secure-server
!
ipv6 route 2002::/16 Tunnel0
!
control-plane
```

Router B

```
Current configuration : 1259 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname Router
!
boot-start-marker
boot-end-marker
!
resource policy
!
no aaa new-model
memory-size iomem 10
no network-clock-participate slot 1
no network-clock-participate wic 0
!
ip cef
!
interface Tunnel0
 no ip address
 no ip redirects
 ipv6 address 2002:C80F:F02::1/128
 tunnel source FastEthernet0/0
 tunnel mode ipv6ip 6to4
!
interface FastEthernet0/0
 ip address 200.15.15.2 255.255.255.0
 duplex auto
 speed auto
!
interface Serial0/0
 no ip address
 shutdown
 no fair-queue
 no dce-terminal-timing-enable
```

```
!
interface FastEthernet0/1
 ip address 200.15.20.2 255.255.255.0
duplex auto
 speed auto
!
interface Serial0/1
 no ip address
 shutdown
 no dce-terminal-timing-enable
!
interface Serial0/2
 no ip address
 shutdown
 no dce-terminal-timing-enable
!
interface Serial0/3
 no ip address
 shutdown
 no dce-terminal-timing-enable
!
router eigrp 100
 network 200.15.15.0
 network 200.15.20.0
 auto-summary
!
ip http server
no ip http secure-server
!
ipv6 route 2002::/16 Tunnel0
!
control-plane
!
line con 0
line aux 0
line vty 0 4
 login
!
!
End
```

Cenário 3

Router A

```
Current configuration : 1276 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname Router
!
boot-start-marker
boot-end-marker
!
resource policy
!
```

```
no aaa new-model
memory-size iomem 10
no network-clock-participate slot 1
no network-clock-participate wic 0
!
ip cef
!
interface Tunnel0
 no ip address
 no ip redirects
 ipv6 address 2002:C80F:F01::1/128
 tunnel source FastEthernet0/0
 tunnel mode ipv6ip 6to4
!
interface FastEthernet0/0
 ip address 200.15.15.1 255.255.255.0
 duplex auto
 speed auto
!
interface Serial0/0
 no ip address
 shutdown
 no fair-queue
 no dce-terminal-timing-enable
!
interface FastEthernet0/1
 ip address 200.15.30.1 255.255.255.0
 duplex auto
 speed auto
!
interface Serial0/1
 no ip address
 shutdown
 no dce-terminal-timing-enable
!
interface Serial0/2
 no ip address
 shutdown
 no dce-terminal-timing-enable
!
interface Serial0/3
 no ip address
 shutdown
 no dce-terminal-timing-enable
!
ip route 200.15.20.0 255.255.255.0 200.15.15.2
!
ip http server
no ip http secure-server
!
ipv6 route 2001::/16 2002:C80F:F02::1
ipv6 route 2002::/16 Tunnel0
!
control-plane
!
line con 0
line aux 0
line vty 0 4
 login
!
```

```
!
End
```

Router B (relay router)

```
Current configuration : 1269 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname Router
!
boot-start-marker
boot-end-marker
!
resource policy
!
no aaa new-model
memory-size iomem 10
no network-clock-participate slot 1
no network-clock-participate wic 0
!
ip cef
ipv6 unicast-routing
!

interface Tunnel0
 no ip address
 no ip redirects
 ipv6 address 2002:C80F:F02::1/128
 tunnel source FastEthernet0/0
 tunnel mode ipv6ip 6to4
!
interface Loopback0
 no ip address
!
interface FastEthernet0/0
 ip address 200.15.15.2 255.255.255.0
 duplex auto
 speed auto
!
interface Serial0/0
 no ip address
 shutdown
 no fair-queue
no dce-terminal-timing-enable
!
interface BRI0/0
 no ip address
 encapsulation hdlc
 shutdown
!
interface FastEthernet0/1
 ip address 200.15.20.1 255.255.255.0
 duplex auto
 speed auto
 ipv6 address 2001:2::3/16
!
interface Serial0/1
```

```
 no ip address
 shutdown
 no dce-terminal-timing-enable
!
ip route 200.15.30.0 255.255.255.0 200.15.15.1
!
ip http server
no ip http secure-server

ipv6 route 2002::/16 Tunnel0
!
control-plane
!
voice-port 1/0/0
!
voice-port 1/0/1
!
line con 0
line aux 0
line vty 0 4
login
!
end
```

6to4 + ISATAP

Cenário 1

Router A 6to4/ISATAP

```
Current configuration : 1479 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname Router
!
boot-start-marker
boot-end-marker
!
resource policy
!
no aaa new-model
memory-size iomem 10
no network-clock-participate slot 1
no network-clock-participate wic 0
!
ip cef
ipv6 unicast-routing
!
interface Tunnel0
 no ip address
 no ip redirects
 ipv6 address 2002:C80F:F01::1/64
 no ipv6 redirects
 tunnel source FastEthernet0/0
 tunnel mode ipv6ip 6to4
!
```

```
interface Tunnel1
 no ip address
 no ip redirects
 ipv6 address 2002:C832:3201::/64 eui-64
 no ipv6 nd suppress-ra
 tunnel source FastEthernet0/1
 tunnel mode ipv6ip isatap
!
interface FastEthernet0/0
 ip address 200.15.15.1 255.255.255.0
 duplex auto
 no shutdown
!
interface Serial0/0
 no ip address
 shutdown
 no fair-queue
 no dce-terminal-timing-enable
!
interface BRI0/0
 no ip address
 encapsulation hdlc
 shutdown
!
interface FastEthernet0/1
 ip address 192.168.20.1 255.255.255.0
 duplex auto
 speed auto
 no shutdown
!
interface Serial0/1
 no ip address
 shutdown
 no dce-terminal-timing-enable
!
 network 192.168.20.0
 network 200.15.15.0
!
ip http server
no ip http secure-server
!
ipv6 route 2001::/16 2002:C80F:F02::1
ipv6 route 2002:C80F:F01::/48 Tunnel1
ipv6 route ::/0 Tunnel0
!
control-plane
!
voice-port 1/0/0
!
voice-port 1/0/1
!
line con 0
line aux 0
line vty 0 4
 login
!
end
```

Router B (relay router)

```
Current configuration : 1264 bytes
```

```
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname Router
!
boot-start-marker
boot-end-marker
!
resource policy
!
no aaa new-model
memory-size iomem 10
no network-clock-participate slot 1
no network-clock-participate wic 0
!
ip cef
ipv6 unicast-routing
!
interface Tunnel0
 no ip address
 no ip redirects
 ipv6 address 2002:C80F:F02::1/128
 tunnel source FastEthernet0/0
 tunnel mode ipv6ip 6to4
!
interface Loopback0
 no ip address
!
interface FastEthernet0/0
 ip address 200.15.15.2 255.255.255.0
 duplex auto
 speed auto
!
interface Serial0/0
 no ip address
 shutdown
 no fair-queue
 no dce-terminal-timing-enable
!
interface BRI0/0
 no ip address
 encapsulation hdlc
 shutdown
!
interface FastEthernet0/1
 ip address 200.15.20.1 255.255.255.0
 duplex auto
 speed auto
 ipv6 address 2001:2::3/16
!
interface Serial0/1
 no ip address
 shutdown
 no dce-terminal-timing-enable
!
ip route 200.15.30.0 255.255.255.0 200.15.15.1
!
```

```
ip http server
no ip http secure-server
!
ipv6 route ::/0 Tunnel0
!
control-plane
!
voice-port 1/0/0
!
voice-port 1/0/1
!
line con 0
line aux 0
line vty 0 4
login
!
end
```