



# Stretnutie 8: Internet Protocol v. 6



BSCI/ROUTE Module 8

# Prečo IPv6?

- Rôzne dôvody
  - Protokol IPv4 je už 30 rokov starý
    - Niektoré jeho vlastnosti sa časom ukázali ako nevyužívané (možno dokonca nebezpečné)
    - Iné vlastnosti potrebné pre súčasné siete mu zasa chýbajú
    - Za 30 rokov sa veľa zmení a dá sa veľa dobrým veciam naučiť, prípadne odpozerat'
  - Najväčším strašiakom je už dlhší čas malý adresový priestor IPv4 – cca 4 mld. adries

This report generated at 03-Apr-2012 07:59 UTC.

IANA Unallocated Address Pool  
Exhaustion:

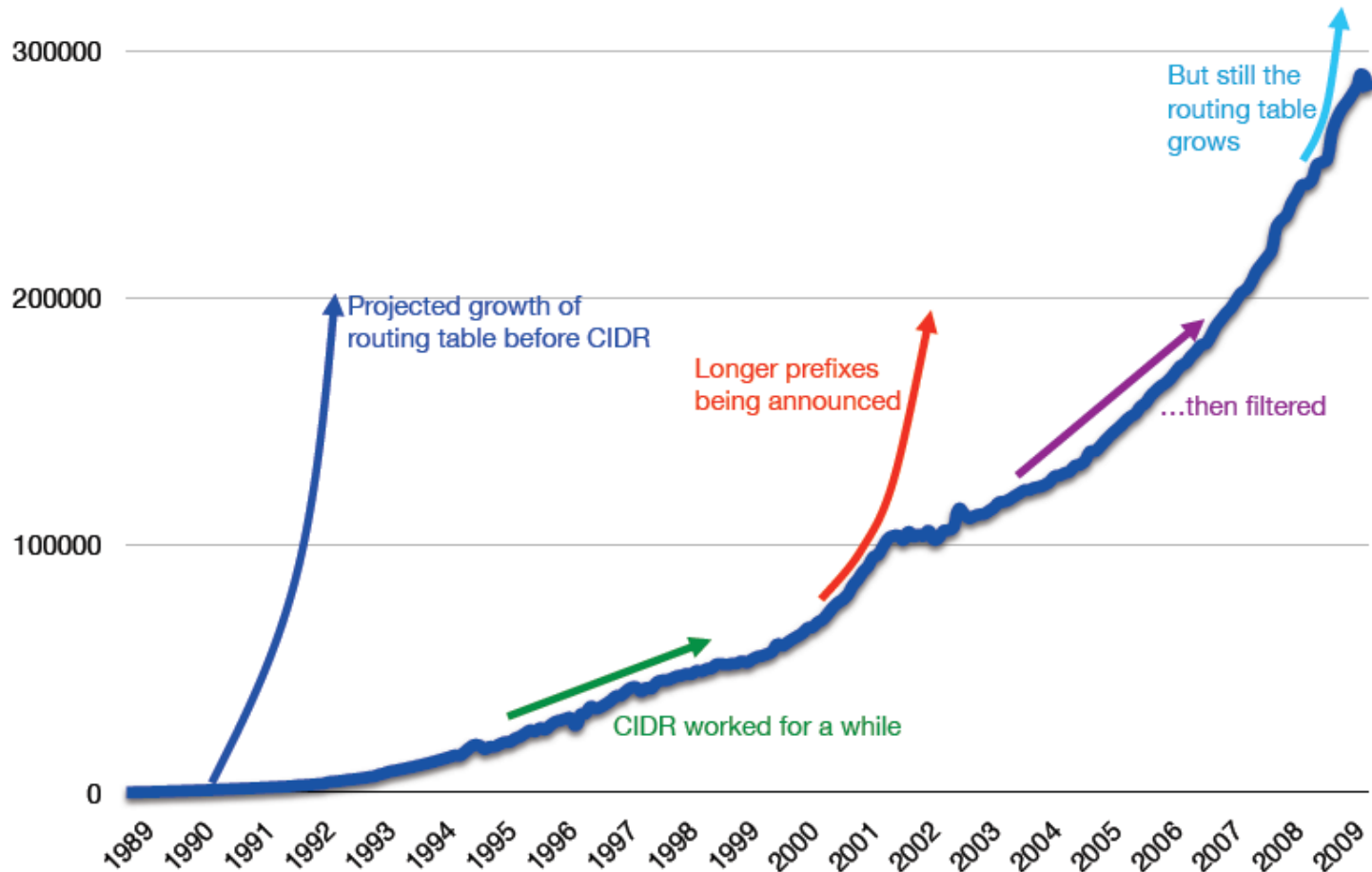
**03-Feb-2011**

Projected RIR Address Pool Exhaustion  
Dates:

RIR	Projected Exhaustion Date	Remaining Addresses in RIR Pool (/8s)
APNIC:	<b>19-Apr-2011</b>	1.1647
RIPENCC:	<b>11-Aug-2012</b>	2.5334
ARIN:	<b>27-Jul-2013</b>	5.5057
LACNIC:	<b>28-Jan-2014</b>	3.7881
AFRINIC:	<b>29-Oct-2014</b>	4.3299

# Komplikovaný IPv4 routing

## Growth of the routing table



# Vlastnosti IPv6

## Väčší adresový priestor

- Globálna dosiahnuteľnosť
- Agregácia
- Multihoming
- Autokonfigurácia
- Komunikácia bez NAT

## Jednoduchšia hlavička

- Efektívne smerovanie
- Bez broadcastov
- Bez kontrolných súm
- Rozširujúce hlavičky
- Flow labels (návestia tokov)

## Mobilita a bezpečnosť

- Mobilita pokrytá RFC dokumentom
- Povinná podpora IPSec

## Migrácia

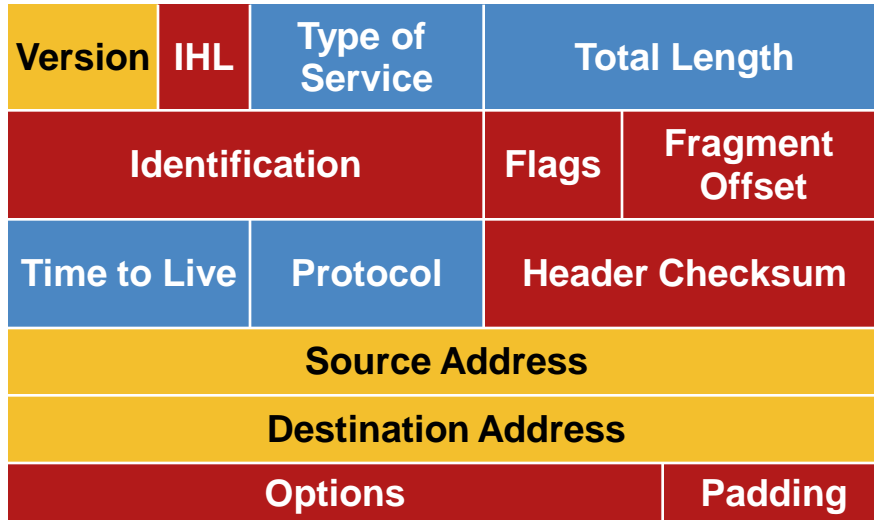
- Dual stack
- Tunelovanie
- Preklad

# Formát IPv6 paketov

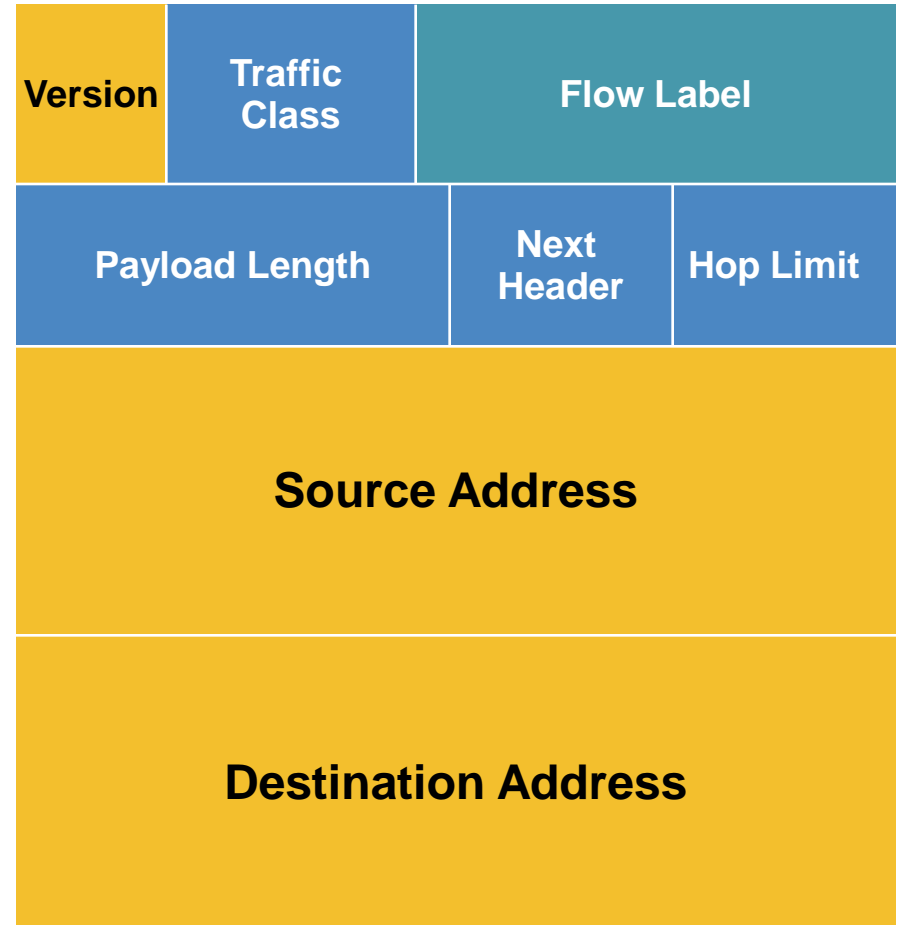


# Porovnanie hlavičiek IPv4 a IPv6

## IPv4 Header



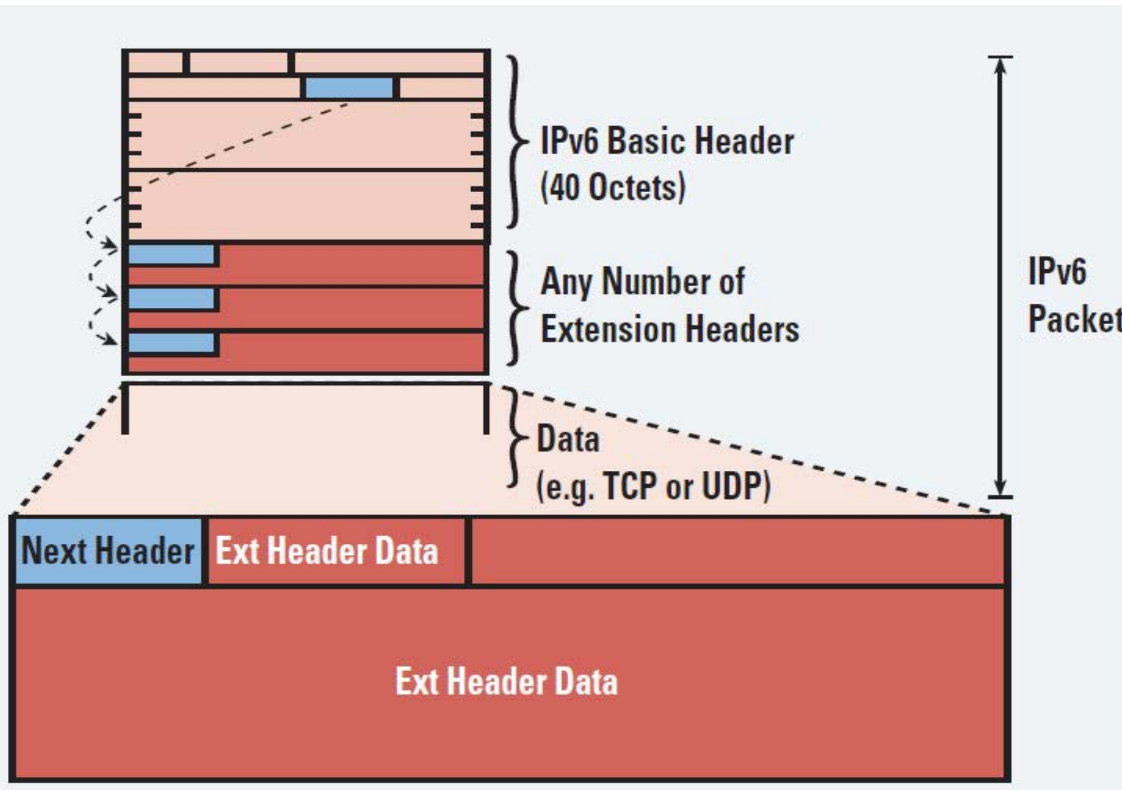
## IPv6 Header



### Legenda

- Polia rovnakého významu
- Polia vypustené z IPv6 hlavičky
- Polia so zmeneným názvom a polohou
- Nové polia v IPv6 hlavičke

# Rozširujúce hlavičky v IPv6



- IPv6 má pre množstvo funkcií rozširujúce hlavičky
- IPv6 efektívnejšie implementuje voliteľné časti hlavičky
  - IPv4 Options musel analyzovať každý smerovač po ceste
- Počet hlavičiek nie je fixný
  - Ich poradie však áno
  - Sú spracovávané v poradí

# Poradie rozšírených hlavičiek

Process Order	Extension Header	Next-header value (protocol #)	Processed
1	Hop-by-hop options header	0	Every router
2	Destination options header	60	Only routers in routing header
3	Routing header	43	List of routers to cross
4	Fragment header	44	Destination
5	Authentication header (AH) and ESP header	ESP = 50 AH = 51	Destination
6	Upper-layer header: TCP UDP	TCP = 6 UDP = 17	Destination



# Záležitosti týkajúce sa MTU

- Pre IPv6 je stanovené minimálne MTU na 1280B (staršie IPv4 malo minimálne MTU 68B)
  - Na rozhraniach, kde je  $MTU < 1280B$ , sa musí použiť špecifický mechanizmus linkového protokolu pre fragmentáciu a opätovné poskladanie
  - IPv6 smerovače už **nerealizujú** fragmentáciu paketu
  - Pokiaľ je to nevyhnutné, fragmentáciu vykonáva priamo odosielateľ
- Od implementácií IPv6 sa očakáva, že sú schopné realizovať tzv. path MTU discovery, ak chcú posielat' pakety väčšie než 1280B
  - Využíva na to ICMPv6 (správa "ICMP packet too big" )
- Minimalistické implementácie IPv6 môžu path MTU discovery vynechať
  - ak všetky pakety udržia na veľkosti  $\leq 1280B$
- IPv6 podporuje pakety veľké až  $2^{32}$  B (tzv. jumbogramy)

# Adresovanie v IPv6



# Väčší adresový priestor

IPv4 = 32 bits

11111111.11111111.11111111.11111111

IPv6 = 128 bits

11111111.11111111.11111111.11111111 11111111.11111111.11111111.11111111 11111111.11111111.11111111.11111111 11111111.11111111.11111111.11111111

## IPv4

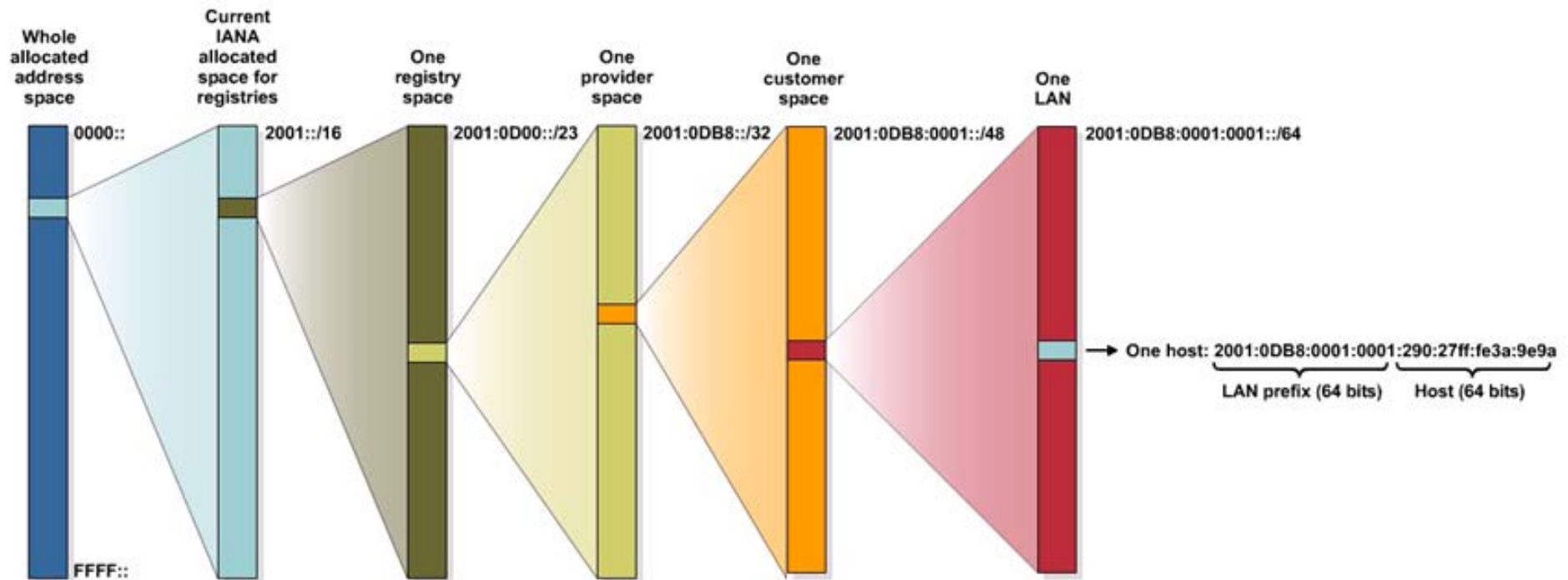
- 32 bitov
  - ~ 4,200,000,000 adresovateľných uzlov

## IPv6

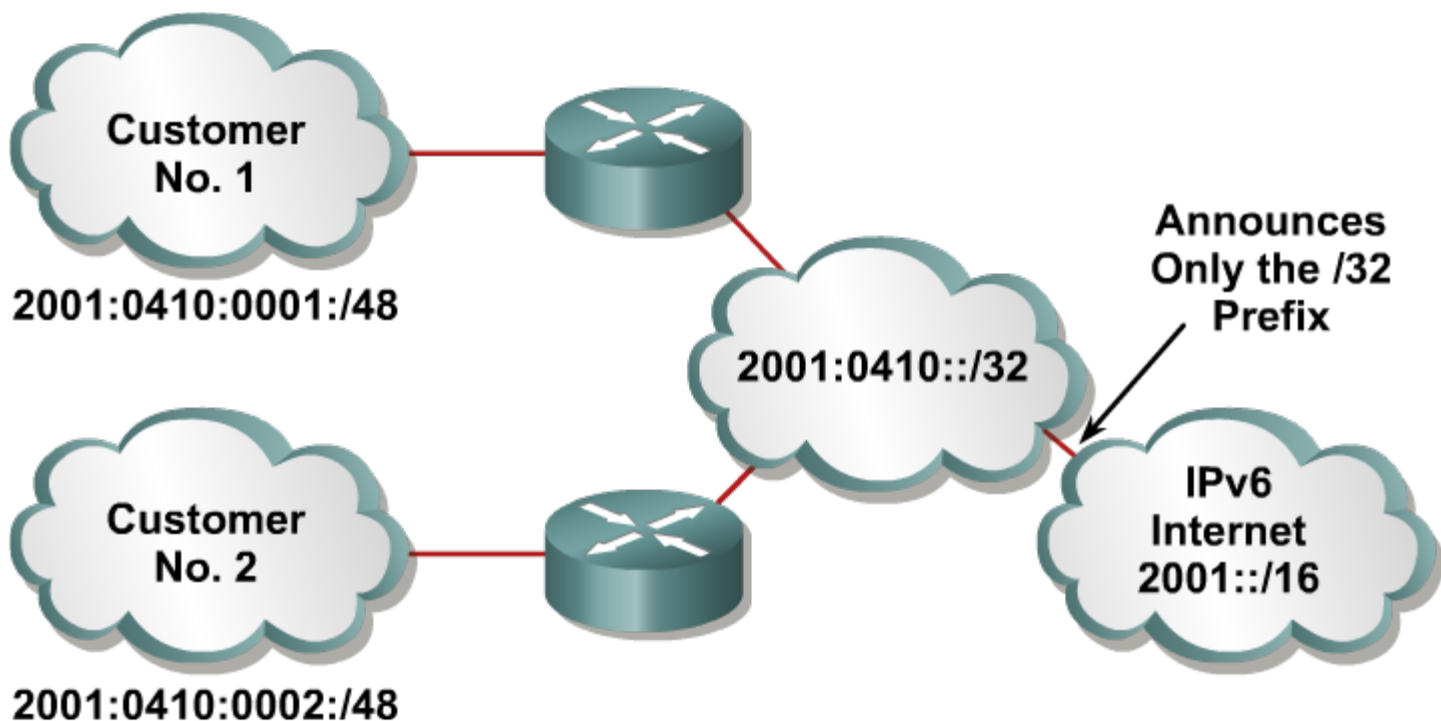
- 128 bitov (16 bajtov): štvornásobná dĺžka oproti IPv4
  - ~  $3.4 * 10^{38}$  adresovateľných uzlov
  - ~ 340,282,366,920,938,463,374,607,432,768,211,456
  - ~  $5 * 10^{28}$  adries na osobu

# Pridelovanie IPv6 adres

- Zatiaľ bolo IANA na RIR alokovaná len malá časť IPv6 globálneho priestoru (asi 12.5%)
  - Rozsah 2001::/16



# Veľký adresový priestor s efektívnou agregáciou



- Do globálnych smerovacích tabuliek sa ohlasujú agregované prefixy
  - Do istej miery toto pripomína návrat ku classful smerovaniu, hoci sa to nedá takto v pravom zmysle slova nazvať
- Efektívne a škálovateľné smerovanie

# Zápis IPv6 adres

**x:x:x:x:x:x:x:x**

- Kde máme **osem** 16-bitových hexadecimálnych polí
  - nazývané “coloned hex”
- Úvodné nuly v hexadecimálnom poli sú nepovinné
  - 2031:0:130F:0:0:9C0:876A:130B
- Za sebou idúce polia 0 sa dajú skrátit' zápisom ::, avšak len jedenkrát v adrese
- Prvé nuly zľava sa môžu v danom hexa poli vynechať

```
2031:0000:130F:0000:0000:09C0:876A:130B
```

```
2031:0:130f::9c0:876a:130b
```

```
FF01:0:0:0:0:0:0:1 >>> FF01::1
```

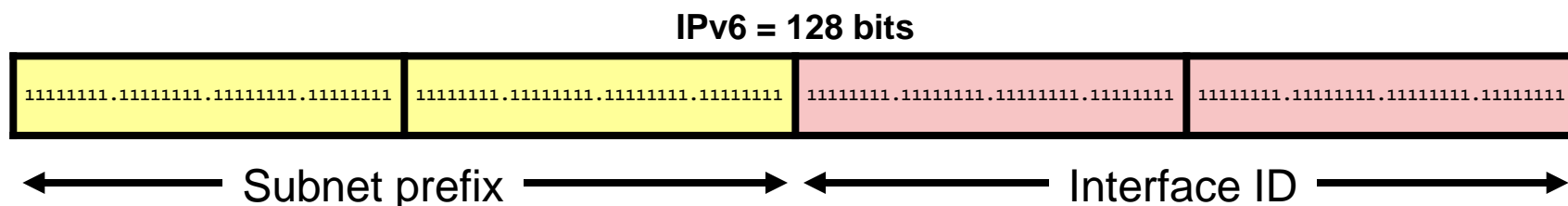
```
0:0:0:0:0:0:0:1 >>> ::1
```

```
0:0:0:0:0:0:0:0 >>> ::
```

- Repräsentácia
  - [http://\[2001:0::43\]:80/index.html](http://[2001:0::43]:80/index.html)

# IPv6 adresa

- Každá IPv6 adresa pozostáva z dvoch polí (ako ipv4)
  - *Subnet prefix*
    - Reprezentujúci sieť ku ktorej je IPv6 host pripojený svojim rozhraním
    - Môže byť rôznej dĺžky
      - Zvyčajne je však dlhý 64bit (/64)
  - *Interface ID*
    - Identifikátor rozhrania
    - Zvyčajne dlhý 64bit (/64)



# Subnet Prefix

- IPv6 používa CIDR zápis
  - *ipv6-address/prefix-length*

Príklad:

**fec0:0:0:1::1234/64**

je preto reálne

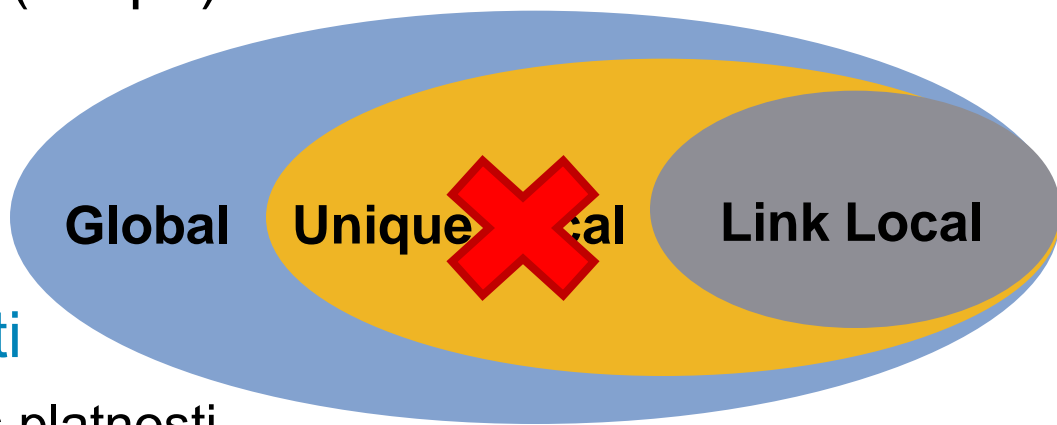
**fec0:0000:0000:0001:0000:0000:0000:1234/64**

- Kde prvých 64-bitov (**fec0:0000:0000:0001**) je adresný prefix
- Posledných 64-bitov (**0000:0000:0000:1234**) je Interface ID.



# Adresový model v IPv6

- Adresy sa priradzujú rozhraniam
- Rozhranie v IPv6 má spravidla niekoľko adries
  - V IPv4 má rozhranie typicky len jednu adresu
- Adresy majú svoj rozsah (scope)
  - Link Local
  - Global unicast
  - Site-local (zastaralý)
- Adresy majú čas platnosti
  - Platnosť a preferovaný čas platnosti
  - Týka sa adries získaných bezstavovou autokonfiguráciou
- Adresy majú svoj typ



# Typy adres

## ■ Unicast

- Adresa patrí jednému rozhraniu
- Existuje mnoho podtypov (napr. globálne a IPv4-mapované)

## ■ Multicast

- Pre one-to-many adresovanie
- Efektívnejšie využíva prostriedky siete
- Používa širší adresový rozsah

## ■ Anycast

- One-to-nearest (alokované z unicastového priestoru)
- Viaceré zariadenia zdieľajú tú istú adresu
- Všetky takéto zariadenia by mali poskytovať rovnaké služby
- Zdrojové zariadenia odosielajú pakety na anycast adresu
- Smerovače rozhodnú o najbližšom uzle s danou adresou
- Vhodné pre load balancing a poskytovanie obsahu (content delivery)

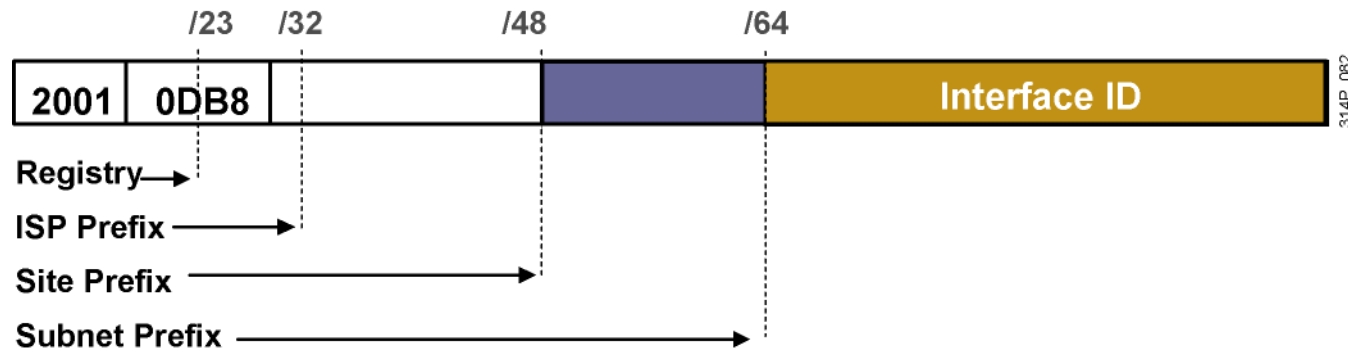
# Adresovanie v IPv6

- Adresy sa priradzujú rozhraniam
- Aktuálna IPv6 adresová architektúra je popísaná v RFC 4291
  - `::/128` Nešpecifikovaná adresa
  - `::1/128` Loopback
  - `FF00::/8` Multicast
  - `FE80::/10` Link-Local Unicast
  - `FEC0::/10` Site local Unicast, zrušené v RFC 3879
  - `FC00::/7` Unique Local Unicast, RFC 4193
  - `::A.B.C.D` IPv4-kompatibilné adresy (neodporúčané)
  - `::FFFF:A.B.C.D` IPv4-mapované adresy
  - Všetko ostatné Global Unicast

# IPv6 adresy na rozhraniach

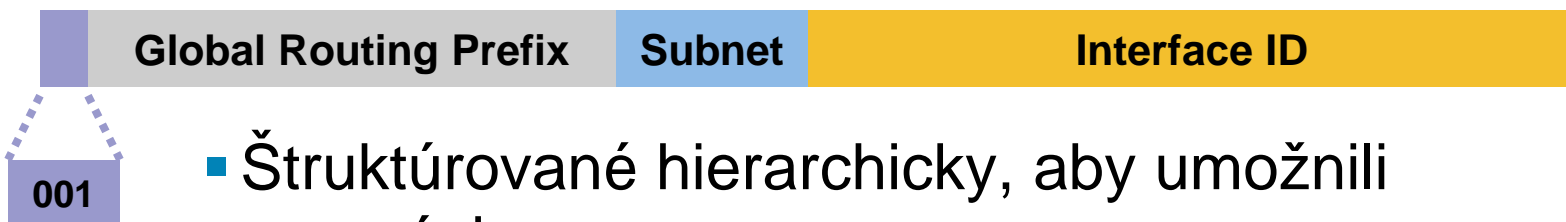
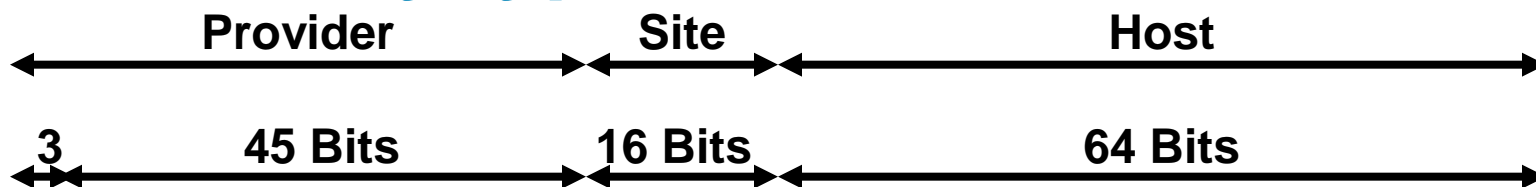
- Na rozhraní môže byť ľubovoľný počet adries každého typu
  - Každé IPv6 rozhranie **musí** mať priradenú prinajmenšom *link-local adresu*
  - Každé IPv6 rozhranie **môže** mať priradené viaceré unikátne *lokálne* či *globálne* adresy
  - Každé IPv6 rozhranie **môže** mať priradené *anycastovú* adresu
  - Adresy typu Global Unicast a Anycast majú rovnaký formát
    - Obsahujú tzv. globálny smerovací prefix,
      - Cieľ => agregovať adresy až po providera
    - Anycastová adresa je jednoducho adresa, ktorá je spoločne pridelená viacerým rozhraniam
      - spravidla na rôznych zariadeniach

# IPv6 adresy Global Unicast a Anycast

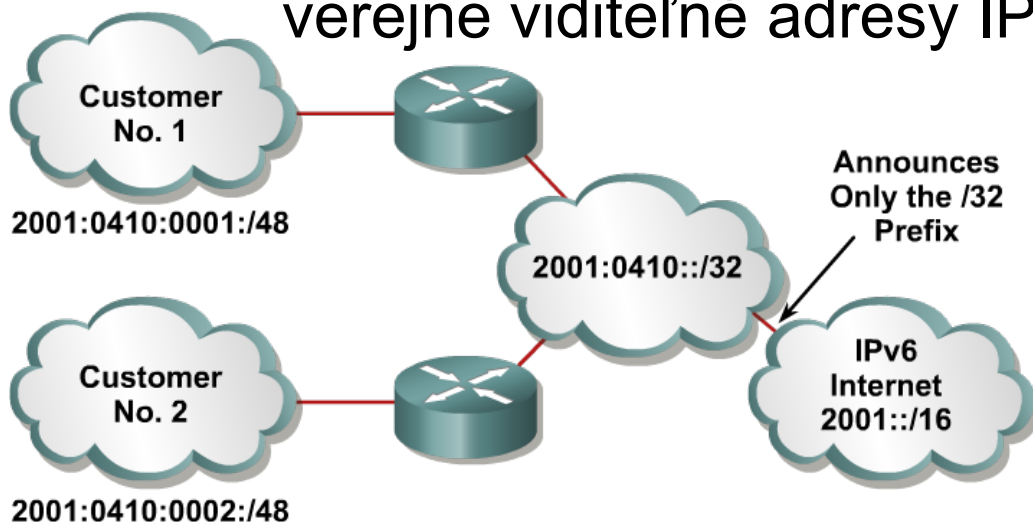


- Adresa typu Global Unicast a Anycast má tri časti
  - Global Routing Prefix (/48)
  - Subnet ID (/16)
  - Interface ID (/64)
- Štruktúra GRP a SID nie je fixná
  - Požaduje sa, aby Interface ID malo 64 bitov, ak IPv6 adresa začína bitmi inými než 000/3
  - V súčasnosti sa pridelujú Global Unicast adresy s prefixom /48 z poolu 2001::/16
    - Subnet ID má teda 16 bitov

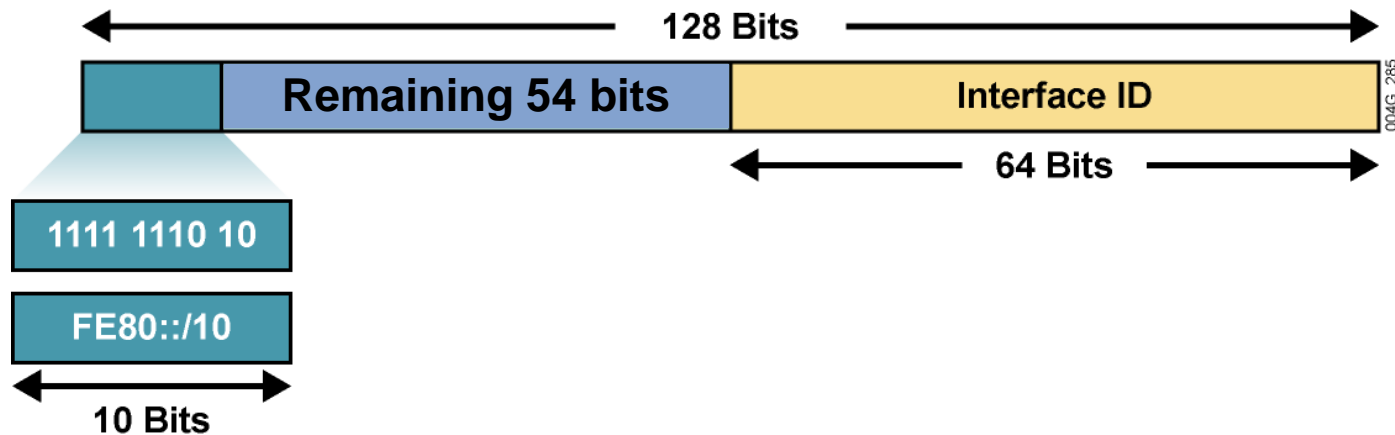
# IPv6 adresy typu Global Unicast



- Štruktúrované hierarchicky, aby umožnili agregáciu
- Tento formát adres sa používa pre unikátne a verejne viditeľné adresy IPv6 staníc



# IPv6 adresy typu Link-Local Unicast



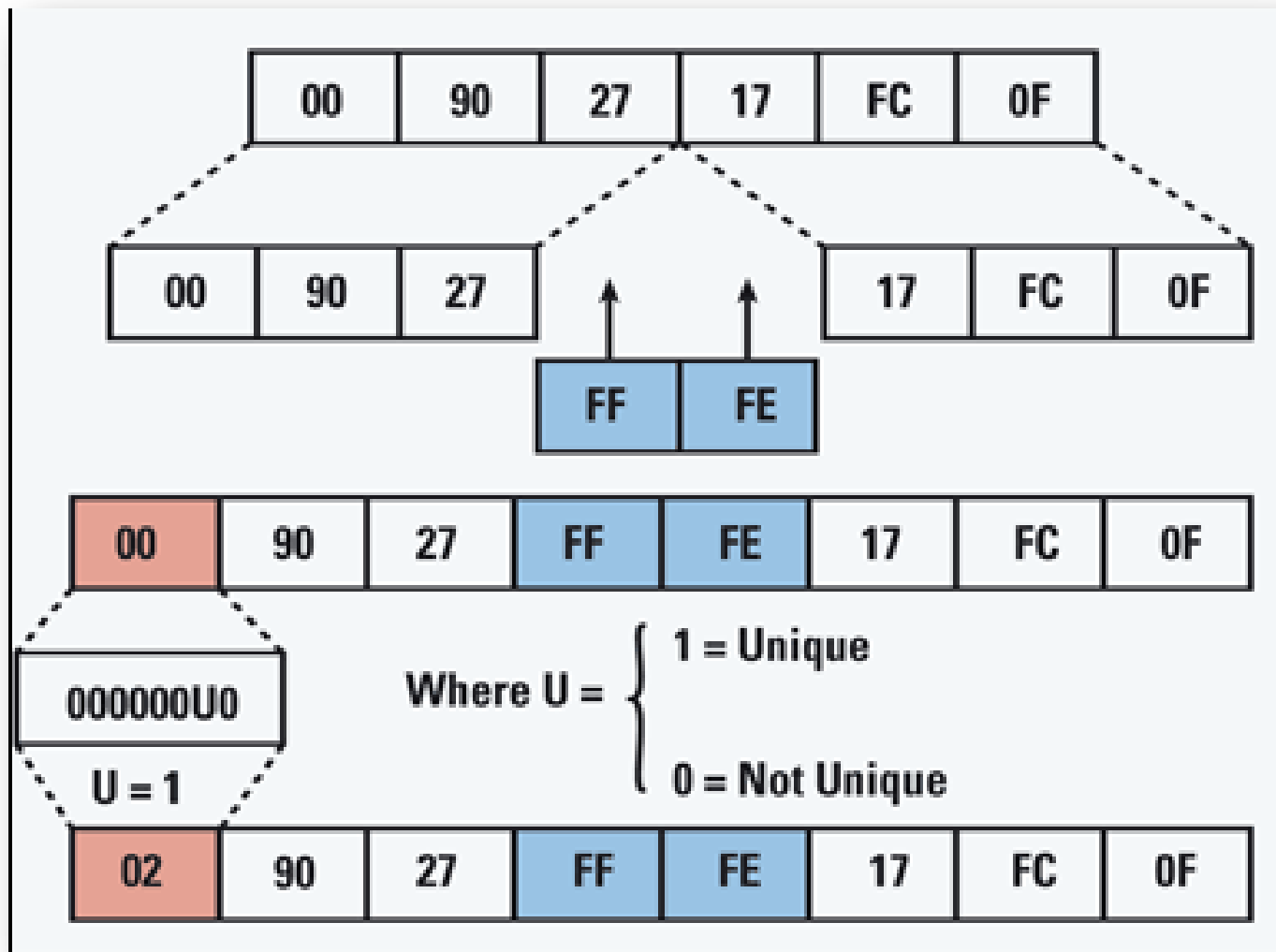
- Link-local adresa pozostáva z prefixu **FE80::/10**, ľubovoľných 54 bitov (nuly) a Interface ID
  - Napr. vo formáte EUI-64 alebo staticky nastavenom
- Umožňuje komunikáciu na linke bez potreby globálnej adresy
- Link-local adresy sú **povinné** pre komunikáciu dvoch susedných zariadení
  - Auto konfigurácia, router discovery, neighbor discovery
  - Takisto sa využívajú v smerovacích protokoloch ako next-hop adresy
- Link-local adresa je vytvorená automaticky
  - Hneď ako je aktivovaná podpora IPv6
- Sú jedinečné a platné len v rámci broadcastovej domény
  - Nie sú smerovateľné
- Pri ich použití sa musí špecifikovať aj rozhranie
  - Napr. ping na susedovú link-local

# IPv6 adresy Global a Link Local Unicast - IPv6 Interface ID

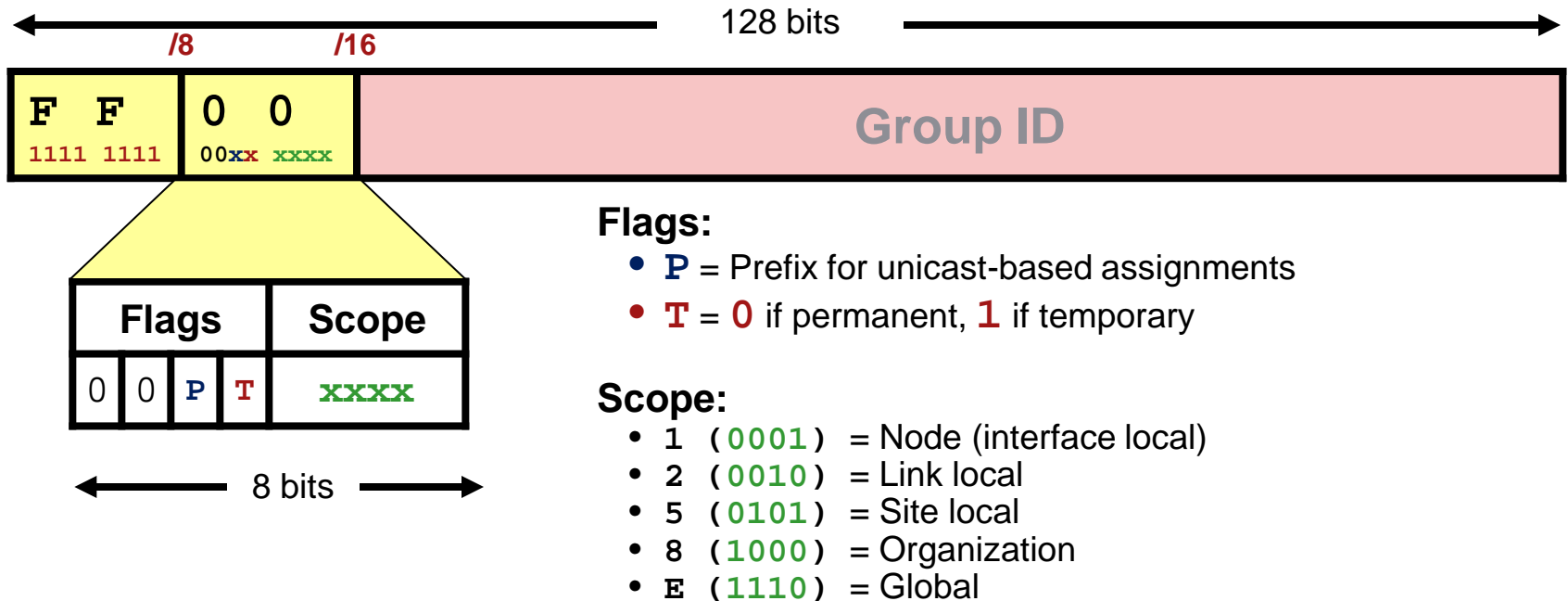
- Pre bezstavovú konfiguráciu IPv6 adresy sa využíva tzv. *modified extended universal identifier* – modified EUI-64
  - Modified EUI-64 má 64 bitov a vkladá sa do časti Interface ID
- Modified EUI-64 sa získa z MAC adresy tým, že:
  - MAC adresu rozdelíme na dve polovice (3B/3B)
  - Medzi tieto polovice vložíme konštantný dvojbajt **FF:FE**
  - Invertujeme bit Universal/Local (druhý najnižší bit 1. bajtu MAC)
- Napríklad:
  - 04:4B:80:6F:7C:B1 → **06**:4B:80:**FF:FE**:6F:7C:B1
- Niektoré operačné systémy si okrem tejto adresy automaticky vygenerujú aj ďalšiu, v ktorej je Interface ID náhodné
  - Cieľom je chrániť súkromie – MAC adresa až príliš jednoznačne identifikuje konkrétny počítač



# Konverzia MAC adresy na EUI-64



# IPv6 multicastové adresy



- Definované prefixom **FF00::/8**
  - 112 bitový identifikátor mcast skupiny
- Multicasty sa v IPv6 využívajú veľmi často
- Broadcast sa v IPv6 nepoužíva
- Prvý oktet FF
- Druhý oktet obsahuje
  - Prefix, platnosť a rozšírenie
  - Adresy **FF00::** to **FF0F::** sú rezervované
    - FF02::/16 – všeobecne známe, link local

# Príklady trvalých multicastových IPv6 adries

Reserved Multicast Address	Description
<b>FF02::1</b>	<ul style="list-style-type: none"><li>• All nodes on a link (link-local scope).</li></ul>
<b>FF02::2</b>	<ul style="list-style-type: none"><li>• All routers on a link.</li></ul>
<b>FF02::9</b>	<ul style="list-style-type: none"><li>• All routing information protocol (RIP) routers on a link.</li></ul>
<b>FF02::1:FFxx:xxxx</b>	<ul style="list-style-type: none"><li>• All <b>solicited-node multicast addresses</b> used for host autoconfiguration and neighbor discovery (similar to ARP in IPv4).</li><li>• The xx:xxxx is the far right 24 bits of the corresponding unicast or anycast address of the node.</li></ul>
<b>FF05::101</b>	<ul style="list-style-type: none"><li>• All Network Time Protocol (NTP) servers.</li></ul>

# Multicastové adresy Solicited-Node

## IPv6 Address



- Solicited-node multicast adresa pozostáva z prefixu [FF02::1:FF/104](#)
  - Adresa má rozsah link-local
  - Má ju každý IPv6 host
- Nasadenie
  - Neighbor discovery (ND)
    - Zistenie link adresy suseda, zistenie default route, zistenie smerovača na linke
  - Bezstavová autokonfigurácia
- Typické použitie je v ICMPv6, ktoré nahrádza ARP
  - Spodných 24 bitov IPv6 adresy je 24 bitov z IPV6 adresy hľadaného suseda
  - ICMPv6 je vo vnútri IPv6 paketu, takže paket musí mať adresu príjemcu, v tomto prípade práve Solicited-Node

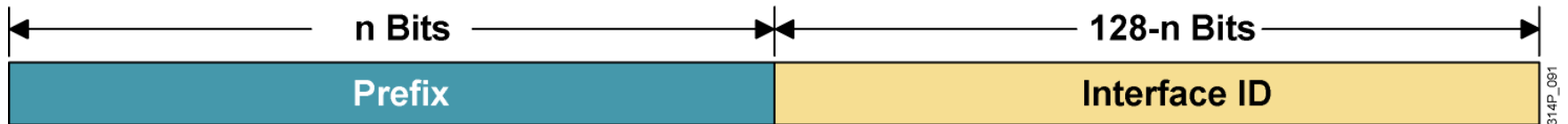
# IPv6 adresy na rozhraní

```
R1#sh ipv6 int e0
Ethernet0 is up, line protocol is up
IPv6 is enabled, link-local address is FE80::21B:D5FF:FE5B:A408
Global unicast address(es):
  2001:8:85A3:4289:21B:D5FF:FE5B:A408, subnet is
2001:8:85A3:4289::/64 [EUI]
Joined group address(es):
  FF02::1
  FF02::2
  FF02::1:FF5B:A408
MTU is 1514 bytes
ICMP error messages limited to one every 100 milliseconds
ICMP redirects are enabled
ND DAD is not supported
ND reachable time is 30000 milliseconds
Hosts use stateless autoconfig for addresses.
```

**Adresa Solicited-Node Multicast**



# IPv6 Anycast adresa



- IPv6 anycast adresa je adresa typu global unicast, ktorá je priradená viac než jednému rozhraniu
- Komunikácia one – to – nearest



## Neighbor discovery a ICMPv6



# Kompetencie protokolu ICMPv6

- V IPv6 má protokol ICMP podstatne dôležitejšiu úlohu než v IPv4
  - Podporuje všetky dôležité funkcie svojho predchodcu
    - Destination Unreachable, Packet Too Big, Time Exceeded, Parameter Problem, Echo/Echo Reply, Redirect
  - Pridáva **nové** funkcie
    - Router Solicitation, Advertisement (do istej miery nahrádza DHCP)
    - Neighbor Solicitation, Advertisement (nahrádza ARP)
- Pridané funkcie ICMP slúžia na
  - Vyhľadanie smerovačov
  - Propagácia default route
  - Automatickú konfiguráciu IPv6 adresy a parametrov IPv6 rozhrania
  - Preklad IPv6 na MAC adresu
  - Detekcia duplikátnej IPv6 adresy v sieti (DAD)

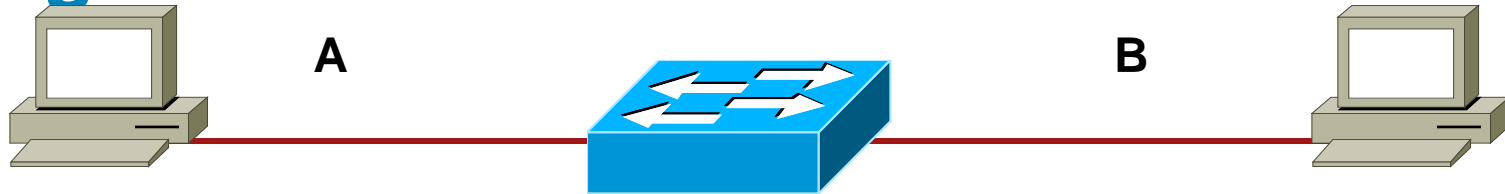


# Neighbor Discovery ICMPv6 – typy paketov

- Neighbor Discovery používa 4 typy ICMPv6 správ:
  - Neighbor Solicitation a Neighbor Advertisement správy
  - Router Solicitation a Router Advertisement správy

ICMPv6 Message	Type	Description
<b>Neighbor Solicitation (NS)</b>	135	<ul style="list-style-type: none"><li>• Posielané hostom na zistenie L2 adresy suseda (ako predtým ARP)</li><li>• Použitá na overovanie dostupnosti suseda</li><li>• Použitá na overovanie duplicity adres (Duplicate Address Detection (DAD)).</li></ul>
<b>Neighbor Advertisement (NA)</b>	136	<ul style="list-style-type: none"><li>• Generovaná ako odpoveď na NS správu.</li><li>• IP host môže poslať aj nevyžiadajúcu odpoveď ako oznámenie o zmene jeho L2 adresy.</li></ul>
<b>Router Advertisement (RA)</b>	134	<ul style="list-style-type: none"><li>• RA obsahuje prefixy ktoré sa používajú na určenie linky, konfiguráciu adres, hop limit, MTU a podobne</li><li>• RA sú posielané periodicky alebo ako odpoveď na RS správu.</li><li>• Generuje smerovač so zapnutým IPv6 unicast routing</li></ul>
<b>Router Solicitation (RS)</b>	133	<ul style="list-style-type: none"><li>• Použitá pri bootovaní hosta na vyžiadanie RA správy od smerovača.</li><li>• Negeneruje smerovač so zapnutým IPv6 unicast routing</li></ul>

# Neighbor Solicitation

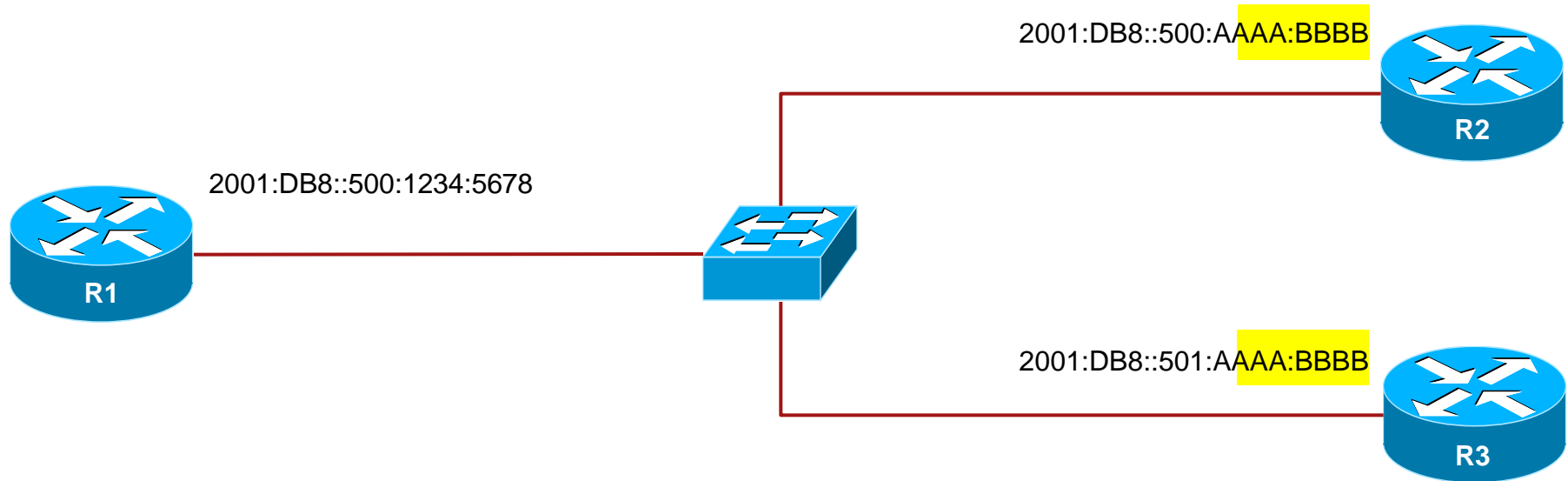


**ICMP type:** 135 (NS)  
**Source:** IPv6 adresa hosta A  
**Destination:** Solicited-node multicast B (**FF02::1:FFxx.xxxx**)  
**Data:** Link-layer adresa hosta A  
**Query:** Aká je tvoja link layer adresa?

**ICMP type:** 136 (NA)  
**Source:** IPv6 adresa hosta B  
**Destination:** IPv6 adresa hosta A  
**Data:** Link-layer adresa hosta B

- Výhoda použitia multicastu v ND oproti broadcastu v ARP je zrejmä
  - Správa nie je spracovávaná všetkými (bcast)
  - Ale len tými uzlami, čo majú danú mcast adresu
- Mapovanie IPv6 adresy na ethernet multicast
  - Vezmi posledných 32 bitov Ipv6 adresy a pridaj 33-33
  - FF02::1:**FF68:12CB** => 33-33-FF-68-12-CB

# Neighbor Solicitation



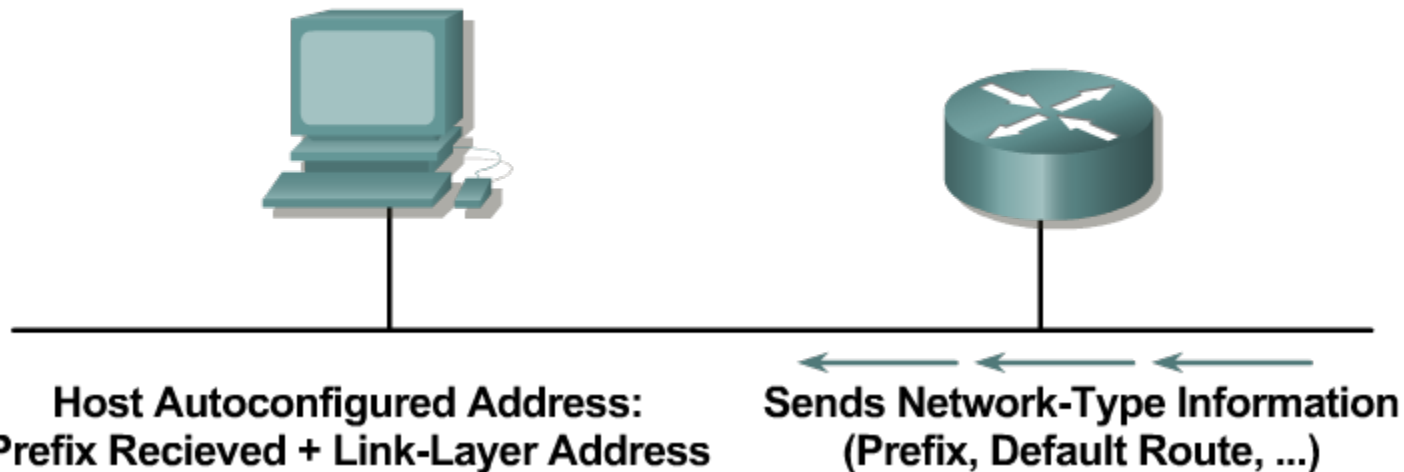
- V prípade, že na sieti je viac hostov s rovnakou solicited-node multicast adresou
  - Príjem všetci s danou adresou
  - Odpoveď len ten, ktorého IPv6 adresa sedí voči IPv6 adrese nesenej v tele NS správy

# DAD – Duplicate address detection

- Používaná IPv6 hostami po tom ako im bola pridelená IPv6 adresa
- Za účelom zistenia jedinečnosti danej IPv6 adresy
  - Používa ND proces
  - Host sa pýta pomocou NS na svoju IPv6 adresu
  - Nemal by dostať zo siete žiadnu odpoveď

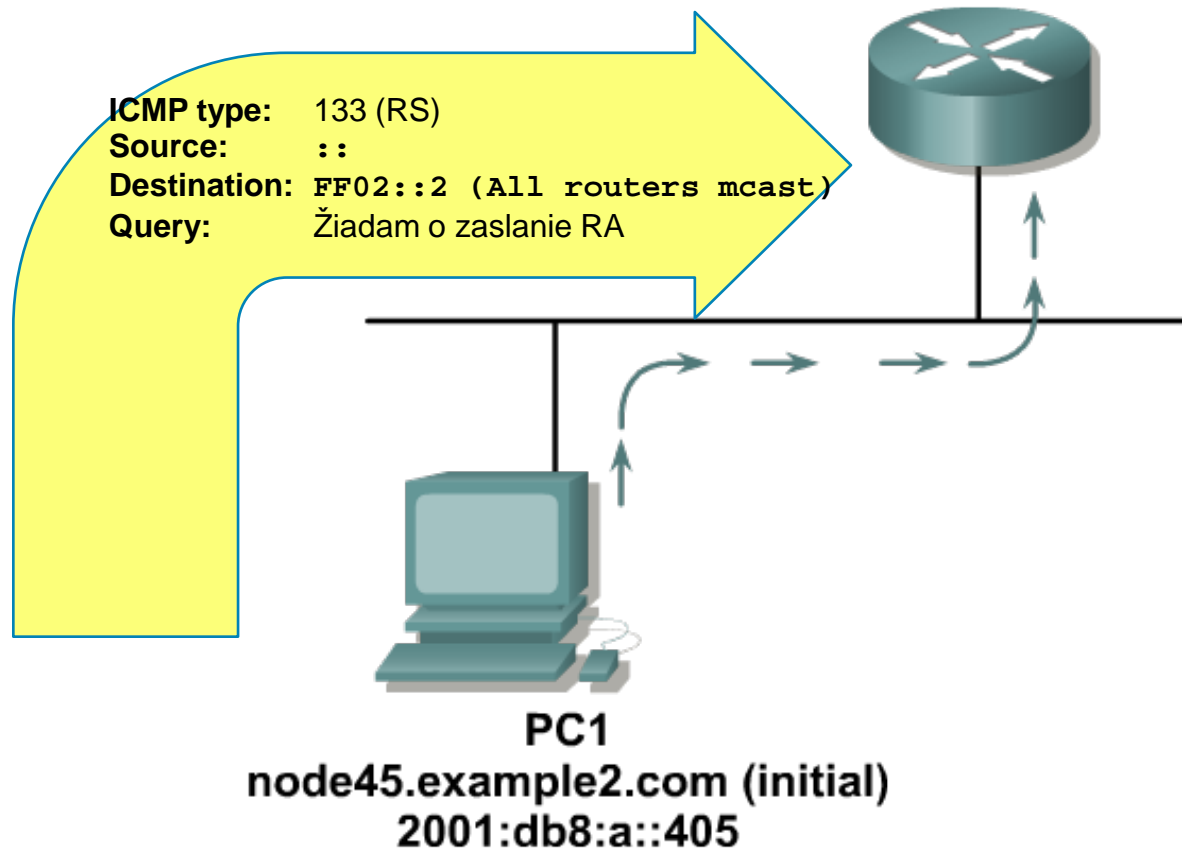
# Bezstavová konfigurácia (Stateless Autoconfiguration)

Interface Identifier ::2004:0FD1:9CAA:1002



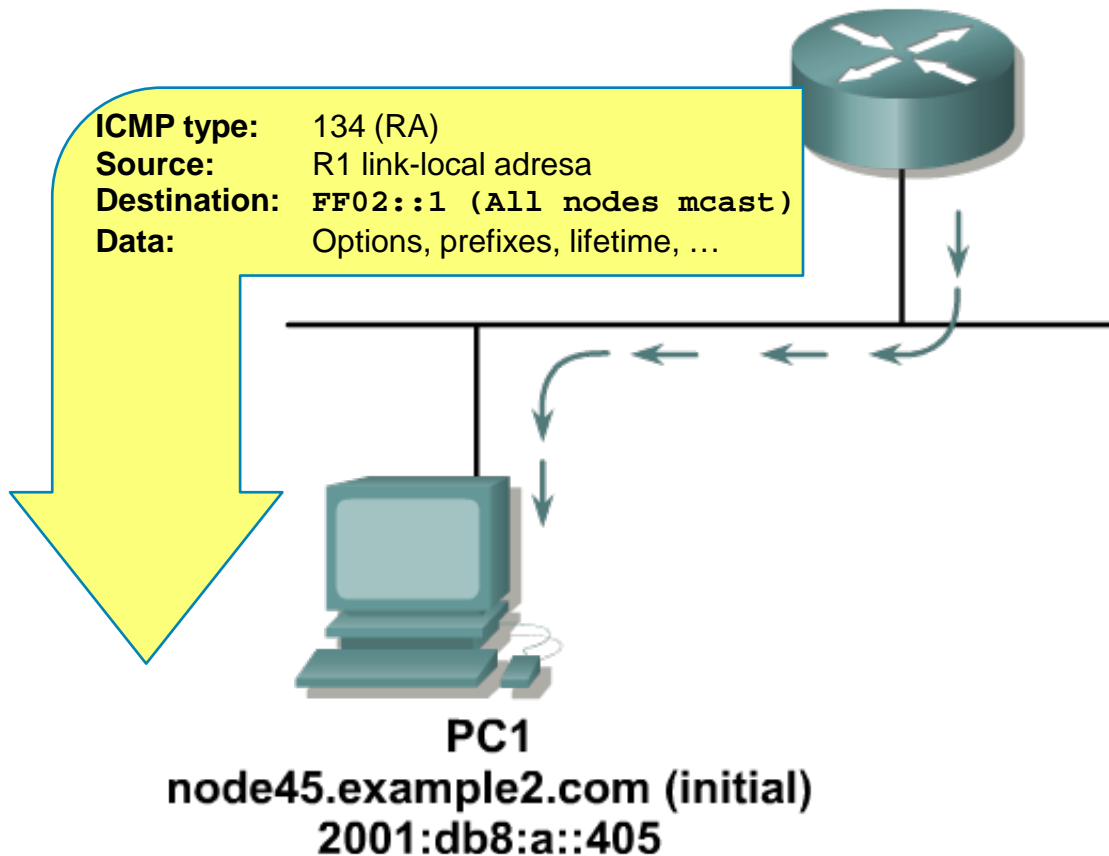
- Každá IPv6 stanica si dokáže sama pridelit' IPv6 adresu
  - tak, že pripojí svoj 64-bitový Interface ID k prefixu siete, ktorý prijala od routera v RA správe
- Smerovač za týmto účelom posiela rozhraním informácie všetkým uzlom na sieti
  - tzv. router advertisement správy, RA
  - RA je posielané periodicky, ale uzol si zaslania môže vynútiť generovaním správy RS

# Postup bezstavovej konfigurácie adresy



- Fáza 1: PC odošle správu „router solicitation (RS)“ a vyžiada si sieťový prefix pre bezstavovú konfiguráciu

# Postup bezstavovej konfigurácie adresy



- Fáza 2: Router odpovedá správou Router Advertisement, v ktorej uvedie okrem iných údajov aj prefix lokálnej siete

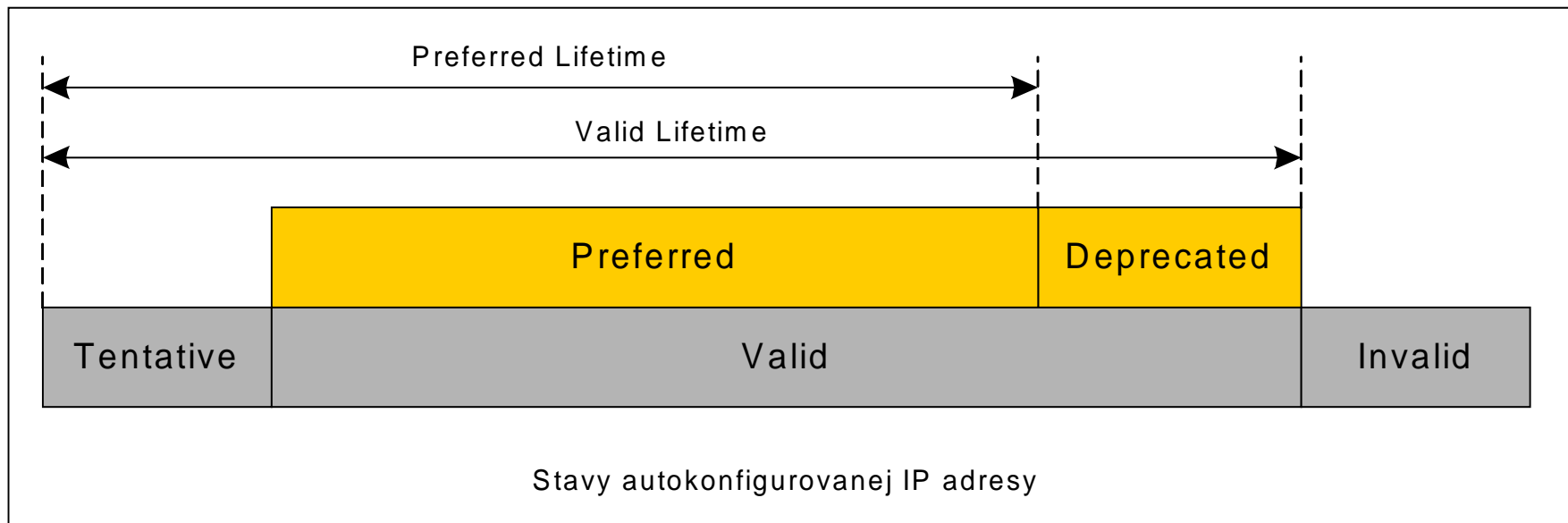
# Autokonfigurované adresy

- Stav automaticky nastavenej adresy:
  - **Tentative** (neoverená, pokusná)
    - V procese preverovania unikátnosti (Duplicate Address Detection)
    - Unicast komunikácia je zakázaná
    - Multicast komunikácia – len správy Neighbor Advertisement
  - **Valid** (platná)
    - Unikátnosť adresy bola potvrdená
    - Adresu je možné používať
    - Stav Valid obsahuje v sebe ďalšie 2 stavy
      - **Preferred** (normálny stav) – adresa je platná
      - **Deprecated** (neschválená) – adresa je platná, ale je zbavená schopnosti nadväzovať nové spojenia
        - existujúca komunikácia môže prebiehať ďalej
  - **Invalid** (neplatná)
    - Do tohto stavu sa adresa dostane po uplynutí časovača Valid Lifetime
    - Adresa v tomto stave nie je použiteľná



# Stavy automatickej IP adresy

- Autokonfigurovaná adresa prechádza týmito stavmi cyklicky
- Trvanie stavov získa zo správy Router Advertisement
- Autokonfigurované adresy obvykle patria na koncové stanice, smerovače ich spravidla nevyužívajú



# Konfigurácia IPv6 na smerovači



# Konfigurácia IPv6 na smerovači

- Konfiguráciu IPv6 smerovania na smerovači je potrebné najprv aktivovať príkazom

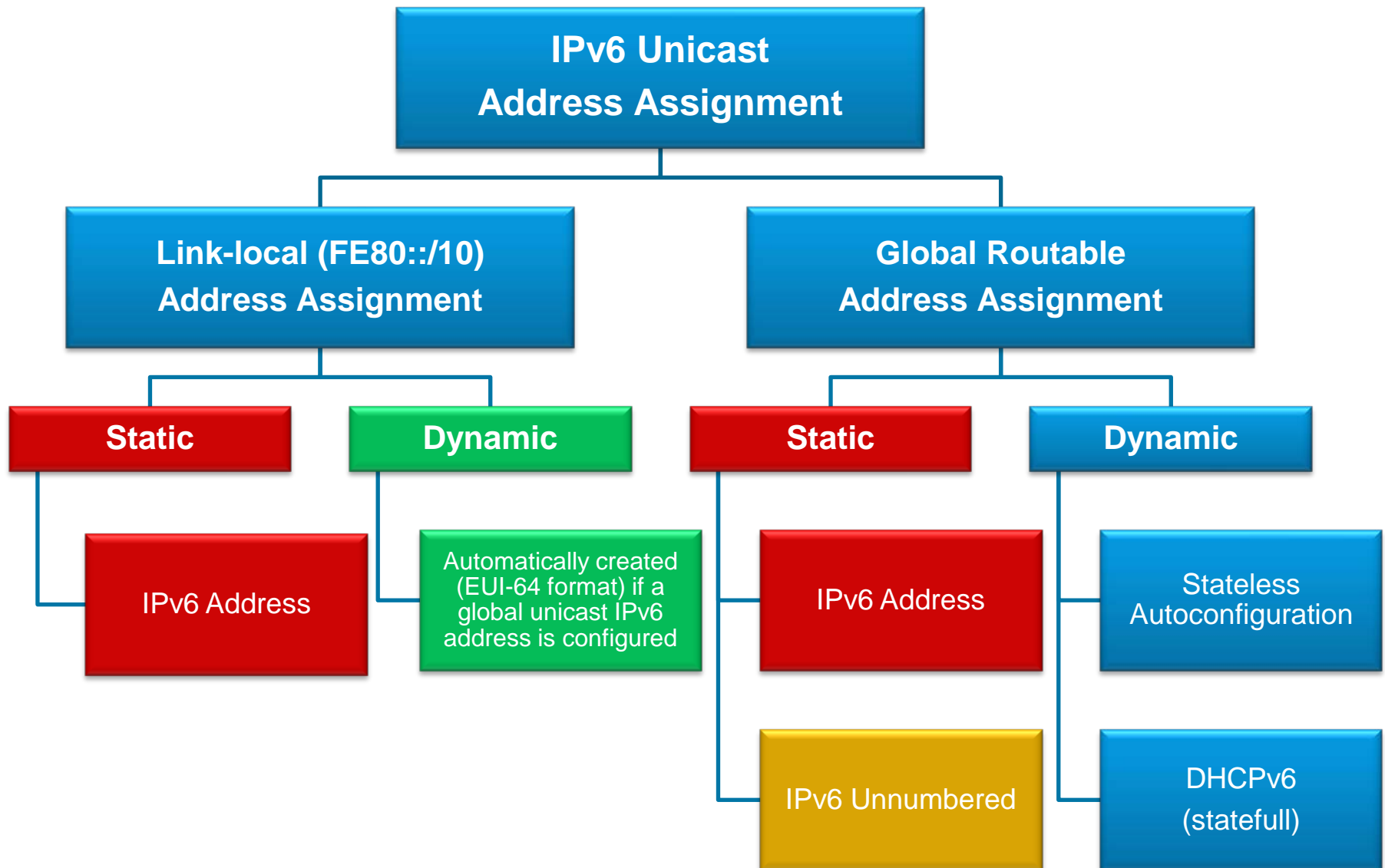
```
Router(config)#
```

```
ipv6 unicast-routing
```

- IPv6 sa konfiguruje veľmi podobne ako IPv4
  - Vo všetkých známych príkazoch sa píše „ipv6“ namiesto „ip“
  - Platí pre konfiguráciu adries či statických ciest
  - Aj overovanie

```
Router(config)# ipv6 route 2000::/3 2001:4118:300:122::1  
Router(config)# interface fa0/0  
Router(config-if)# ipv6 address 2001:4118:300:123::1/64
```

# Možnosti konfigurácie unicast IPv6 adres



# Statické pridelenie IPv6 adresy

Static

- Statické pridelenie IPv6 adresy a povolenie IPv6 na rozhraní

```
Router(config-if)#
```

```
ipv6 address address/prefix-length [link-local]
```

- Príklad

- Statické pridelenie *globálnej IPv6 adresy*

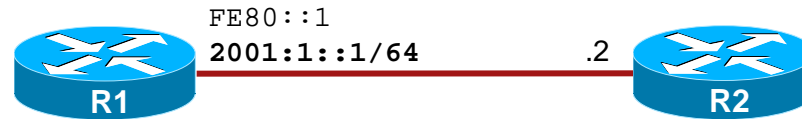
```
Router(config-if)# ipv6 address 2001:1::1/64
```

- Statické pridelenie *link-local adresy*

```
Router(config-if)# ipv6 address FE80::1 link-local
```

- Nie je treba pridávať masku
  - Zvyčajne je generovaná automaticky (keď host získa IPv6 adresu
    - EUI-64 alebo static global
  - Vhodné napr. ak automaticky generovaný formát sa zle pamätá

# Overenie nastavenia IPv6 adresy



```
R1# show ipv6 interface fa0/0
FastEthernet0/0 is up, line protocol is up
IPv6 is enabled, link-local address is FE80::1 [TEN]
Global unicast address(es):
  2001:1::1, subnet is 2001:1::/64 [TEN]
Joined group address(es):
  FF02::1
  FF02::2
  FF02::1:FF00:1
MTU is 1500 bytes
ICMP error messages limited to one every 100 milliseconds
ICMP redirects are enabled
ND DAD is enabled, number of DAD attempts: 1
ND reachable time is 30000 milliseconds
ND advertised reachable time is 0 milliseconds
ND advertised retransmit interval is 0 milliseconds
ND router advertisements are sent every 200 seconds
ND router advertisements live for 1800 seconds
Hosts use stateless autoconfig for addresses.
R1#
```

# Dynamické pridelenie IPv6 adresy EUI-64

- Dynamické pridelenie IPv6 adresy s generovaním Interface ID

```
Router(config-if)#
```

```
ipv6 address address/prefix-length eui-64
```

- Príklad

- Dynamické pridelenie *globálnej IPv6 adresy EUI-64*
  - Generovanie Interface ID z L2 adresy

```
Router(config-if)# ipv6 address 2001::/64 eui-64
```

```
R2(config-if)# do show interface fa0/0 | include addr  
Hardware is Gt96k FE, address is 0019.5592.b212 (bia 0019.5592.b212)  
R2(config-if)# do show ipv6 interface fa0/0 | include Global  
Global unicast address(es): 2001::219:55FF:FE92:B212  
R2(config-if)#
```

# Konfigurácia IPv6 Unnumbered

- Povolí IPv6 na rozhraní bez explicitne uvedenej adresy

```
Router(config-if)#
```

```
ipv6 unnumbered INTERFACE-TYPE INTERFACE-NUMBER
```

- Použije na danom rozhraní IPv6 adresu zo špecifikovaného rozhrania
  - Ako source IPv6 adresu
  - Rozhrania z ktorého si požičiavam musí byť „up“

```
R1(config)# interface loopback 10
R1(config-if)# ipv6 address 2001:1::10/64
R1(config-if)# exit
R1(config)#
R1(config)# interface s0/0/0
R1(config-if)# ipv6 unnumbered loopback 10
R1(config-if)# no shut
R1(config-if)# do show ipv6 interface s0/0/0
Serial0/0/0 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::222:55FF:FE18:7DE8
  No Virtual link-local address(es):
  Interface is unnumbered. Using address of Loopback10
  No global unicast address is configured
  Joined group address(es):
    FF02::1
    FF02::2
... <output omitted> ...
```



# Spôsoby konfigurácie IPv6 adresy

- Aktivácia IPv6 na rozhraní bez explicitnej adresy:

```
Router(config-if)#
```

```
ipv6 enable
```

- Tento príkaz aktivuje podporu IPv6 na rozhraní
  - Rozhranie dostane link-local adresu, no nebude mať nijakú ďalšiu adresu
  - Môže byť vhodné pre rýdzo tranzitné linky (podobne ako IP Unnumbered), pretože smerovacie protokoly využívajú výlučne link-local adresy ako next-hop adresy

# Bezstavová autokonfigurácia

- Nastaví auto konfiguráciu:

```
Router(config-if)#
```

```
ipv6 address autoconfig [default]
```

- Adresa je pridelená na základe prefixu prijatého v RA
- `default`: inštalácia default route cez tento smerovač

# Nastavenie ND Neighbor časovača

```
Router(config-if)#
```

```
ipv6 nd reachable-time MILLISECONDS
```

- Špecifikácia času v [ms], po uplynutí ktorého je vzdialený IPv6 sused považovaný za nedostupného
  - Default je
    - 0ms pre RA
    - 30000ms pre ND aktivity
  - **Poznámka:** Krátky čas zvyšuje množstvo vymieňaných správ

# Statické pridanie IPv6 suseda

- Príkaz pridá IPv6 suseda do discovery cache

```
Router(config)#
```

```
ipv6 neighbor IPV6-ADDRESS INTERFACE-TYPE INTERFACE-  
NUMBER HARDWARE-ADDRESS
```

- Mapovanie IPv6 na HW adresu

# IPv6 konektivita cez NBMA siete

- Pri NBMA, ako napr. FR, je treba spraviť mapovanie (príkaz **frame-relay map**)
  - pre link-local IPv6 adresy
    - Používané smerovacími protokolmi
  - aj pre global unicast IPv6 adresy

# Základné overenie IPv6

Command	Description
<pre>show ipv6 interface [brief] [type number] [prefix]</pre>	<p>Displays the status of interfaces configured for IPv6.</p> <ul style="list-style-type: none"><li>• The <b>brief</b> keyword displays a brief summary.</li><li>• The <b>prefix</b> keyword displays the IPv6 neighbor discovery prefixes that are configured on a specified interface.</li></ul>
<pre>show ipv6 routers [interface-type interface-number] [conflicts]</pre>	<p>Displays IPv6 router advertisement information received from on-link routers (those locally reachable on the link).</p> <ul style="list-style-type: none"><li>• The <b>conflicts</b> keyword displays information about routers advertising parameters that differ from the advertisement parameters configured for the specified interface on which the advertisements are received.</li></ul>
<pre>show ipv6 neighbors [interface-type interface-number   ipv6- address   ipv6-hostname   statistics]</pre>	<p>Displays IPv6 neighbor discovery cache information for the specified neighbors.</p> <ul style="list-style-type: none"><li>• The optional <b>statistics</b> parameter displays neighbor discovery cache statistics.</li></ul>

# Diagnostika IPv6 ND

```
Router# show ipv6 neighbors
```

IPv6 Address	Age	Link-layer Addr	State	Interface
2001:1::2	4	0019.55df.ad22	STALE	Fa0/0
FE80::219:55FF:FEDF:AD22	4	0019.55df.ad22	STALE	Fa0/0

```
Router# ping 2001:1::2
```

IPv6 Address	Age	Link-layer Addr	State	Interface
2001:1::2	4	0019.55df.ad22	REACH	Fa0/0
FE80::219:55FF:FEDF:AD22	4	0019.55df.ad22	STALE	Fa0/0

- REACH

- So susedom sme komunikovali v rámci času `nd reachable-time`

- STALE

- So susedom sme už určitú dobu nekomunikovali
  - Čas `nd reachable-time` uplynul

# Diagnostika IPv6

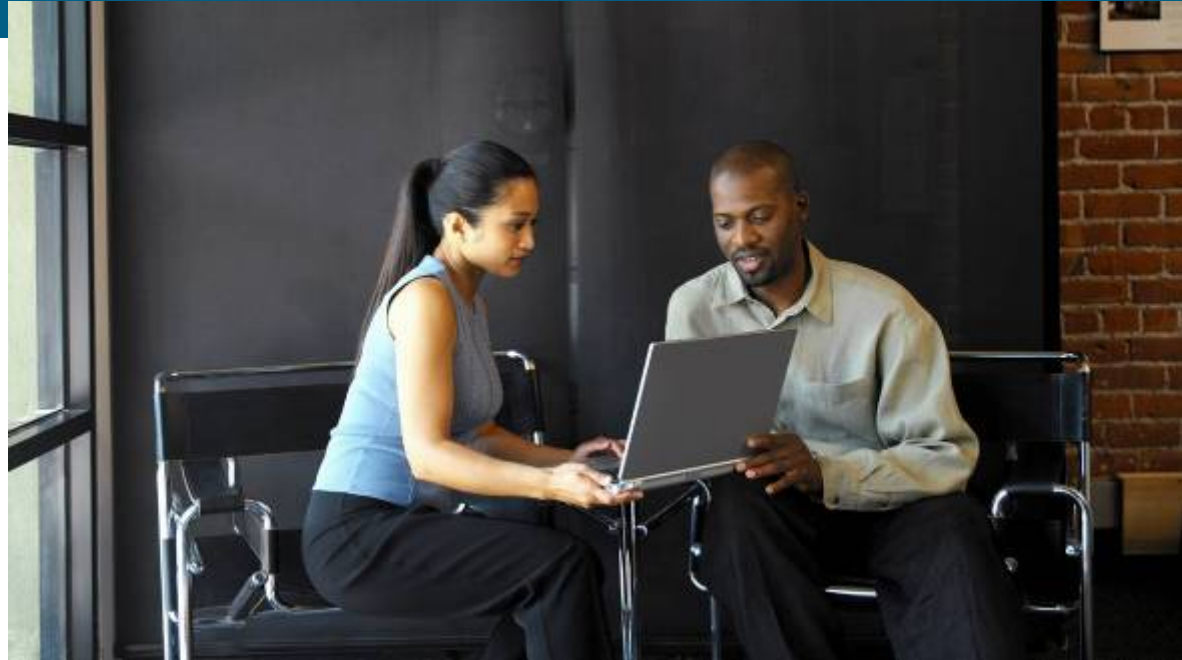
Command	Description
<code>debug ipv6 nd</code>	<p>Displays messages associated with ICMPv6 neighbor discovery.</p> <ul style="list-style-type: none"><li>• ICMPv6 neighbor discovery is the IPv6 replacement for the IPv4 ARP.</li></ul>
<code>debug ipv6 packet</code> <code>[<i>access-list access-list-name</i>] [detail]</code>	<p>Displays information associated with IPv6 packet processing.</p> <ul style="list-style-type: none"><li>• When an IPv6 access list is specified, only packets permitted by the ACL are displayed.</li><li>• The <code>detail</code> keyword displays more information.</li></ul>



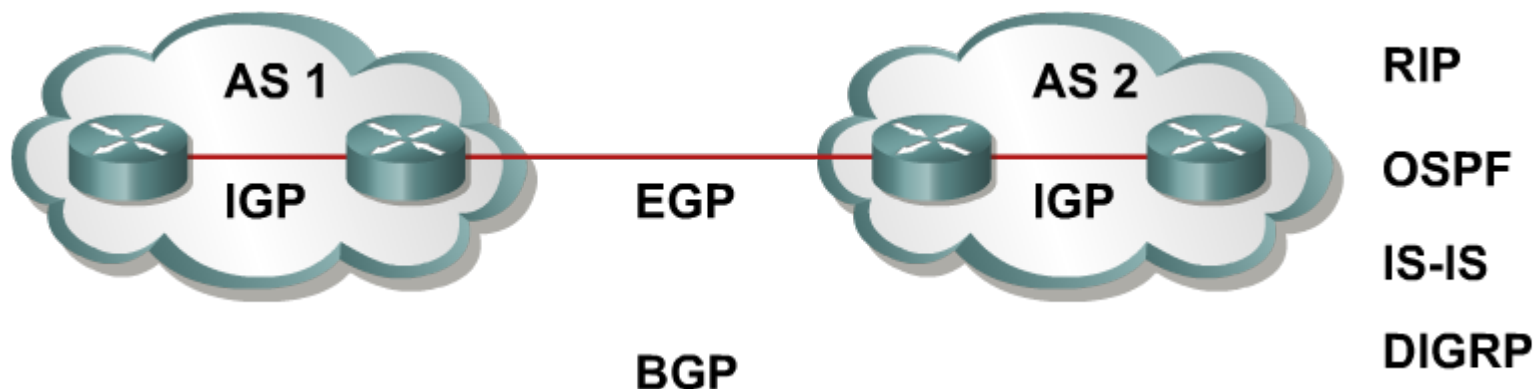
# Diagnostika IPv6 ND a DAD

```
R2# debug ipv6 nd
ICMP Neighbor Discovery events debugging is on
R2#config t
Enter configuration commands, one per line. End with CNTL/Z.
R2(config)#interface fa0/0
R2(config-if)#ipv6 address 2001:1:1001::1/64
R2(config-if)#
*Aug 13 13:58:29.400: ICMPv6-ND: Adding prefix 2001:1:1001::1/64 to
FastEthernet0/0
*Aug 13 13:58:29.400: ICMPv6-ND: Sending NS for 2001:1:1001::1 on
FastEthernet0/0
*Aug 13 13:58:30.400: ICMPv6-ND: DAD: 2001:1:1001::1 is unique.
*Aug 13 13:58:30.400: ICMPv6-ND: Sending NA for 2001:1:1001::1 on
FastEthernet0/0
*Aug 13 13:58:30.400: ICMPv6-ND: Address 2001:1:1001::1/64 is up on
FastEthernet0/0
R2(config-if)#exit
R2(config)# ipv6 unicast-routing
R2(config)#
*Aug 13 14:00:07.428: ICMPv6-ND: Sending RA to FF02::1 on FastEthernet0/0
*Aug 13 14:00:07.428: ICMPv6-ND: MTU = 1500
*Aug 13 14:00:07.428: ICMPv6-ND: prefix = 2001:1::/64 onlink autoconfig
2592000/604800 (valid/preferred)
*Aug 13 14:00:08.428: ICMPv6-ND: Received NA for
2001:1:1001:0:219:56FF:FE2C:9F60 on FastEthernet0/0 from
2001:1:1001:0:219:56FF:FE2C:9F60
```

# Smerovanie IPv6 prevádzky



# Smerovacie protokoly s podporou IPv6



- Možnosti pre smerovanie v IPv6
  - Statické smerovanie
  - RIPng (RFC 2080)
  - OSPFv3 (RFC 2740)
  - IS-IS pre IPv6
  - MP-BGP4 (RFC 2545/2858)
  - EIGRP pre IPv6
- Všetky IGP protokoly sa aktivujú **na rozhraniach**,
  - nie príkazmi `network`
- Okrem IS-IS majú všetky protokoly svoj konfiguračný odstavec s názvom **ipv6 router PROTOKOL [parametre]**
- Pre každý smerovací protokol musí byť aktivované smerovanie
  - **ipv6 unicast-routing**

# Statické smerovanie

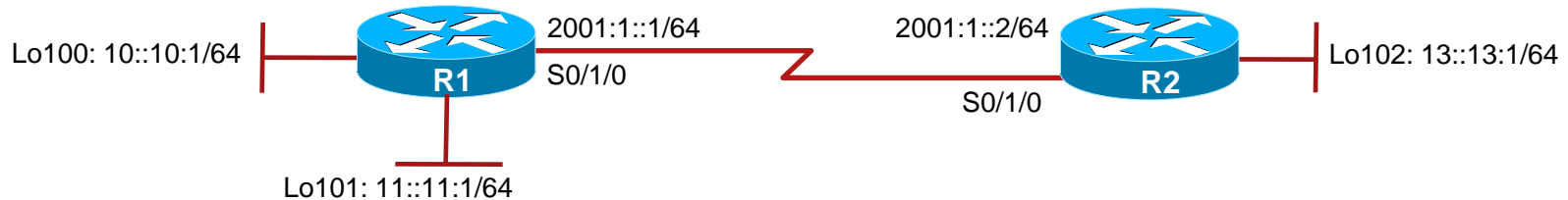
- Tak isto ako v IPv4

```
Router(config)#
```

```
ipv6 route ipv6-prefix/prefix-length {ipv6-address |  
interface-type interface-number [ipv6-address]}  
[administrative-distance] [administrative-multicast-  
distance | unicast | multicast] [next-hop-address] [tag  
tag]
```

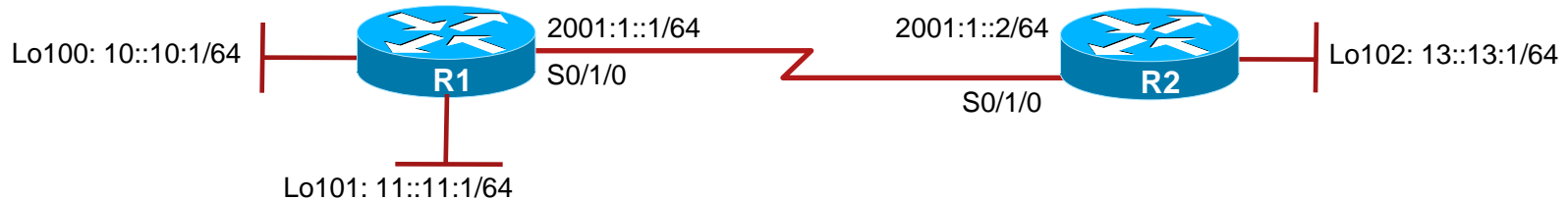
- Pre použitie outgoing interface a next hop IP adresu platia tie isté pravidla
  - Priamo pripojená statická cesta (out interface)
    - Vhodné pre P-t-P
  - Rekurzívna statická cesta
    - Používa NEXT HOP IP a je nutný rekurzívny lookup
  - Plne špecifikovaná statická cesta
    - Odporúčaná pre BMA `outgoing_int + next_hop_IPv6_addr`

# Príklad Static Route



```
R1(config)# ipv6 unicast-routing
R1(config)# ipv6 route 13::/64 s0/1/0
R1(config)# exit
R1# show ipv6 route static
IPv6 Routing Table - 9 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
S      13::/64 [1/0]
       via ::, Serial0/1/0
R1#
```

# Príklad default static route



```
R2(config)# ipv6 unicast-routing
R2(config)# ipv6 route ::/0 s0/1/0
R2(config)# exit
R2# show ipv6 route static
IPv6 Routing Table - 9 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
S      ::/0 [1/0]
       via ::, Serial0/1/0
R2#
```



# RIPng



# RIPng (RFC 2080)

- Rovnaké vlastnosti ako v IPv4:
  - Distance-vector princíp, maximálne 15 hopov, split-horizon a poison reverse
  - Založený na RIPv2
- Aktualizované vlastnosti pre IPv6:
  - Prenos prefixov a next-hop adries vo formáte IPv6
  - Aktualizácie posiela pomocou UDP, port 521, na multicastovú adresu FF02::9, tzv. *all-rip-routers* skupinu
  - Adresa odosielateľa je IPv6 link-local adresa

```
Router(config-if)#
```

```
ipv6 rip NAME enable
```

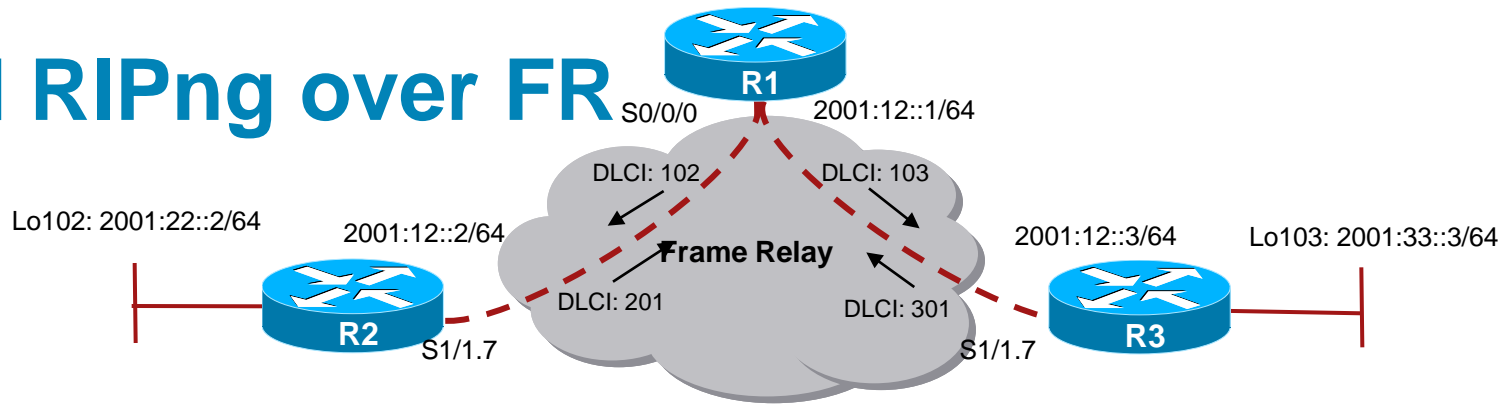
- Ak proces s daným menom neexistuje je vytvorený



# Overenie a diagnostika RIPng

Command	Description
<code>show ipv6 protocols</code> <code>[summary]</code>	Displays the parameters and current state of the active IPv6 routing protocol processes. The <code>summary</code> keyword specifies that only the configured routing protocol process names are displayed.
<code>debug ipv6 rip</code> <code>[interface-type</code> <code>interface-number]</code>	Displays IPv6 RIPng routing transaction debug messages. The <code>interface-type interface-number</code> option can be used to display interface specific debug messages.

# Príklad RIPng over FR



```
interface Serial1/1.7
  multipoint
  ipv6 address 2001:12::2/64
  frame-relay map ipv6 2001:12::1
    201 broadcast
  frame-relay map ipv6
    FE80::219:56FF:FE2C:9F60 201
    broadcast
  ipv6 unicast-routing
  ipv6 rip RIPTag enable

interface lo102
  ipv6 rip RIPTag enable
```

```
interface Serial0/0/0
  encapsulation frame-relay IETF
  ipv6 address 2001:12::1/64
  frame-relay lmi-type cisco
  frame-relay map ipv6 2001:12::2
    102 brodcast
  frame-relay map ipv6 2001:12::3
    103 brodcas
  frame-relay map ipv6
    FE80::2B0:64FF:FE33:FB60 102
    broadcast
  frame-relay map ipv6
    FE80::250:73FF:FE3D:6A20 103
    broadcast
  ipv6 unicast-routing
  ipv6 rip RIPTag enable
  no split-horizon
```

```
interface Serial1/1.7
  multipoint
  ipv6 address 2001:12::3/64
  frame-relay map ipv6 2001:12::1
    301 broadcast
  frame-relay map ipv6
    FE80::219:56FF:FE2C:9F60 301
    broadcast
  ipv6 unicast-routing
  ipv6 rip RIPTag enable

interface lo103
  ipv6 rip RIPTag enable
```



# EIGRP



# EIGRP pre IPv6

- EIGRP pre IPv6 je stále distance-vector routing protokol.
  - Konfigurácia a činnosť je podobná ako pre IPv4
  - Slovíčko `ip` nahrádza `ipv6`
- Čo ostáva v EIGRP pre IPv6 ako bolo pre IPv4:
  - Používa sa rovnaké protokolové číslo (88)
  - Použitie topo tabuľky a generovanie Query/Updates
  - Použitie DUAL na kalkuláciu successor routes
- Rozdiely EIGRP IPv6 oproti IPv4:
  - Používajú sa IPv6 prefixy.
  - Konfigurácia je na rozhraní
    - Príkaz `network` neexistuje
  - Pre zakladanie susedských vzťahov sa používa Link-local adresa
    - Môžu byť susedia aj keď nie je pridelená globálna adresa
  - Pre IPv6 sa nerobí auto sumarizácia.

# EIGRP pre IPv6

- EIGRP pre IPv6 je dostupné v IOS 12.4T
- Pre svoj beh vyžaduje 4-bajtové RouterID
  - Ak nie je možné RouterID zistiť, je ho nutné nakonfigurovať ručne
- Smerovací protokol je štandardne „vypnutý“, treba ho preto následne „zapnúť“

```
Router(config)# interface fa0/0
Router(config-if)# ipv6 eigrp 64512
Router(config-if)# exit
Router(config)# ipv6 router eigrp 64512
Router(config-rtr)# router-id 158.193.138.255
! Alternatívne podľa IOS
Router(config-rtr)# eigrp router-id 158.193.138.255
Router(config-rtr)# no shutdown
```

# Konfigurácia Stub Router

- Nastav smerovač ako stub router.

```
Router(config-rtr)#
```

```
eigrp stub [receive-only | connected | static | summary  
| redistributed]
```

- **Note:**

- Od Cisco IOS Release 15.0(1)M a 12.2(33)SRE, príkaz eigrp stub nahrádza príkaz stub.

# Sumarizácia IPv6 ciest

- Konfigurácia agregovanej sumarizovanej adresy na rozhraní

```
Router(config-if)#
```

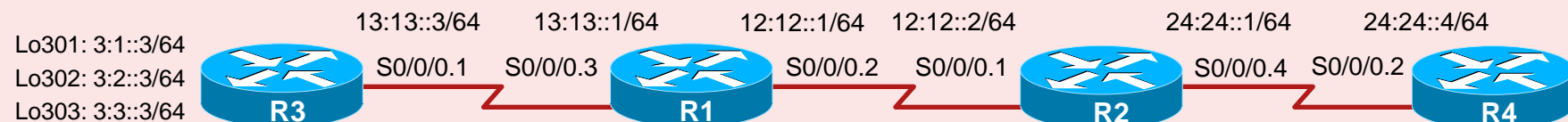
```
ipv6 summary-address eigrp AS-NUMBER IPV6-ADDRESS  
[ADMIN-DISTANCE]
```

- Činnosť ako pre IPv4.

Parameter	Description
<i>as-number</i>	Specifies the EIGRP AS number for which routes are to be summarized.
<i>ipv6-address</i>	The IPv6 address of the summary route.
<i>admin-distance</i>	(Optional) Specifies the administrative distance, a value from 0 through 255. The default value is 90.

# Príklad EIGRP

## IPv6 EIGRP AS 100



R3

```
ipv6 unicast-routing
interface Serial0/0/0.1 point-
to-point
ipv6 eigrp 100
interface loopback 301
ipv6 eigrp 100
interface loopback 302
ipv6 eigrp 100
interface loopback 303
ipv6 eigrp 100

ipv6 router eigrp 100
eigrp router-id 3.3.3.3
no shutdown

interface serial 0/0/0.1
ipv6 summary-address eigrp 100
3::/16
```

R1

```
ipv6 unicast-routing
interface Serial0/0/0.2 point-
to-point
ipv6 e
interface Serial0/0/0.3 point-
to-point
ipv6 eigrp 100

ipv6 router eigrp 100
eigrp router-id 1.1.1.1
no shutdown
```

R2

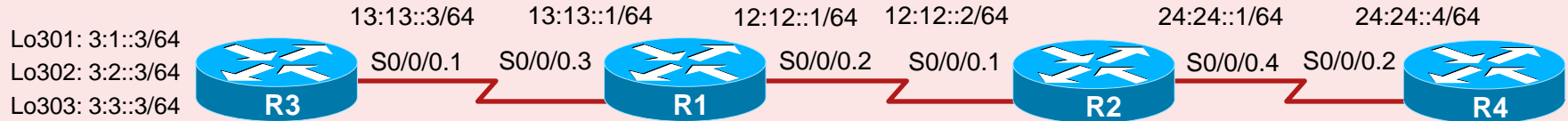
```
ipv6 unicast-routing
interface Serial0/0/0.1 point-
to-point
ipv6 eigrp 100
interface Serial0/0/0.4 point-
to-point
ipv6 eigrp 100

ipv6 router eigrp 100
eigrp router-id 2.2.2.2
no shutdown
```



# Overenie

## IPv6 EIGRP AS 100



```
R4# show ipv6 route eigrp
```

```
<output omitted>
```

```
D   3::/16 [90/3321856]  
    via FE80::2, Serial0/0/0.2  
D  12::/64 [90/2681856]  
    via FE80::2, Serial0/0/0.2  
D  13::/64 [90/3193856]  
    via FE80::2, Serial0/0/0.2
```

```
R4#
```

# OSPFv3



# OSPFv3 - Podobnosti s OSPFv2

- OSPFv3 je verzia OSPF pre IPv6 (RFC 2740):
  - Vychádza z OSPFv2 + rozšírenia
  - Distribuuje IPv6 prefixy
  - Beží priamo nad IPv6
- OSPFv3 & v2 je možné v sieti prevádzkovať súčasne
  - Sú však úplne nezávislé a každá si spravuje vlastnú databázu
- OSPFv3 používa rovnaké základné typy paketov:
  - Hello
  - Database description (DBD)
  - Link state request (LSR)
  - Link state update (LSU)
  - Link state acknowledgement (LSAck)

# OSPFv3 - Podobnosti s OSPFv2

- Mechanizmus zisťovania susedov a tvorby neighborhood/adjacency sú identické
- Podpora pre RFC-definované NBMA a point-to-multipoint režimy je identická, rovnako ako Cisco špecifická podpora pre point-to-point a broadcast
- Princíp pre LSA flooding and aging je zhodný
- Podpora multi area (NSSA) je zhodná

# Rozdiely medzi OSPFv2 a OSPFv3

- OSPFv3 má rovnaké typy paketov, niektoré však zmenili formát

Packet Type	Description
1	Hello
2	Database Description
3	Link State Request
4	Link State Update
5	Link State Acknowledgement

- Všetky OSPFv3 pakety majú 16B hlavičku namiesto 24B hlavičky v OSPFv2

Version	Type	Packet Length
Router ID		
Area ID		
Checksum	Autype	
Authentication		
Authentication		

Version	Type	Packet Length
Router ID		
Area ID		
Checksum	Instance ID	0

314P\_110

# Rozdiely medzi OSPFv2 a OSPFv3

- OSPFv3 pracuje nad linkou, nie nad subnetom
  - V IPv6 sú rozhrania pripojené k linkám, nie k subnetom
    - Indikuje sa tým možnosť komunikovať cez L2
- Nad jednou linkou môže byť mnoho IPv6 „subnetov“ – prefixov
- Pojmy „network“ a „subnet“ sa nahrádzajú pojmom “link”
- Dva OSPFv3 smerovače môžu komunikovať, aj ak na linke nemajú spoločný subnet – prefix
  - Používajú link-local adresy, nie global unicast

# Rozdiely medzi OSPFv2 a OSPFv3

- Nad jednou linkou môže bežať viacero OSPFv3 inštancií
- Je možné, aby jedna linka patrila do viacerých nezávislých oblastí
  - napríklad pri spojení dvoch autonómnych systémov
- V paketoch OSPFv3 je definované nové pole s názvom *Instance ID*, ktoré identifikuje inštanciu OSPFv3, ktorá daný paket vygenerovala
  - Aby dva OSPFv3 smerovače komunikovali, musia sa zhodnúť na číse inštancie.
  - By default je číslo inštancie 0 a pre každú ďalšiu inštanciu sa inkrementuje.

# Rozdiely medzi OSPFv2 a OSPFv3

- Na identifikáciu suseda sa používa unicast link-local adresa
- Použité multicastové adresy
  - FF02::5 – Všetky OSPFv3 routery na segmente (link-local scope)
  - FF02::6 – Všetky DR/BDR routery na segmente (link-local scope)
- Zmena v spôsobe používania adres
  - IPv6 adresy sa nenachádzajú v hlavičkách OSPF paketov (súčasť payload)
  - Router LSA (LSA1) a network LSA (LSA2) neobsahujú IPv6 adresy
    - Tie sú v nových LSA8 (link) a LSA9 (intra-area prefixes)
  - Router ID, area ID a link-state ID zostávajú 32-bitové
  - DR a BDR sú identifikované podľa ich Router ID, nie podľa ich IPv6 adresy
- Bezpečnosť
  - OSPFv3 využíva IPSec hlavičky AH a ESP pre autentifikáciu
  - Samotné OSPFv3 neobsahuje nijaký mechanizmus pre autentifikáciu



# Prehľad LSA

LSA Name	LS Type code	Flooding scope	LSA Function code
Router LSA	0x2001	Area scope	1
Network LSA	0x2002	Area scope	2
Inter-Area-Prefix-LSA	0x2003	Area scope	3
Inter-Area-Router-LSA	0x2004	Area scope	4
AS-External-LSA	0x4005	AS scope	5
Group-membership-LSA	0x2006	Area scope	6
Type-7-LSA	0x2007	Area scope	7
Link-LSA	0x0008	Link-local scope	8
Intra-Area-Prefix-LSA	0x2009	Area scope	9

Premenované

Nové

# Link LSA – LSA 8

- A link LSA per link
- Link local scope flooding on the link with which they are associated
- Provide router link local addresses
- List all IPv6 prefixes attached to the link
- Assert a collection of option bit for the Router-LSA

# Intra Area Prefix (LSA9)

- Zoznam prefixov
- Odvolávka na Link ID v LSA1/2

# Inter-Area Prefix LSA – LSA 3

- Describes the destination outside the area but still in the AS
- Summary is created for one area, which is flooded out in all other areas
- Originated by an ABR
- Only intra-area routes are advertised into the backbone
- Link State ID simply serves to distinguish inter-area-prefix-LSAs originated by the same router
- Link-local addresses must never be advertised in inter-area- prefix-LSAs

# Konfigurácia OSPFv3



# Konfigurácia OSPFv3

- OSPFv3 príkazy sú v mnohom podobné tým v OSPFv2
  - Zámena kľúčového slova **ip** za **ipv6**
- OSPFv3 sa zásadne konfiguruje na rozhraniach
  - Príkaz **network** už neplatí a využíva sa spôsob konfigurácie priamo na rozhraní
    - Podobne ako v novších IOS pre OSPFv2, resp. IS-IS
    - Namiesto slova „ip“ sa používa „ipv6“
- Konfiguračná sekcia pre OSPFv3 (arey, autentifikácia, Router ID...)
  - Samostatný kontext: **ipv6 router ospf process-id**
  - Väčšina známych príkazov z OSPFv2 sa používa rovnako, resp. primerane vzhľadom na zmenu formátu adresy
  - napr. sumarizácia príkazmi **area range**, **summary-prefix**, nastavenie typov arey, redistribúcia, atď...
    - Summary route má Cost najvyšší zo sumarizovaných
      - oproti Cisco OSPFv2 – tam je najmenší

# Povolenie a spustenie OSPFv3 procesu

## ■ Voliteľný príkaz

```
Router(config)#
```

```
ipv6 router ospf process-id
```

- *process-id* jednoznačne identifikuje OSPFv3 proces lokálnej smerovaču

```
R1(config)# ipv6 router ospf 10
```

```
R1(config-rtr)#?
```

area	OSPF area parameters
auto-cost	Calculate OSPF interface cost according to bandwidth
default	Set a command to its defaults
default-information	Distribution of default information
default-metric	Set metric of redistributed routes
discard-route	Enable or disable discard-route installation
distance	Administrative distance
distribute-list	Filter networks in routing updates
ignore	Do not complain about specific event
log-adjacency-changes	Log changes in adjacency state
maximum-paths	Forward packets over multiple paths
passive-interface	Suppress routing updates on an interface
process-min-time	Percentage of quantum to be used before releasing CPU
redistribute	Redistribute IPv6 prefixes from another routing protocol
router-id	router-id for this OSPF process
summary-prefix	Configure IPv6 summary prefix
timers	Adjust routing timers

# Router ID

- OSPF 3 však pre svoj beh vyžaduje 4-bajtové RouterID
  - Formát IPv4 adresy
  - RID musí byť jedinečný
  - Proces výberu RID ako v OSPFv2
    - RID, najvyššia IPv4 loop adresa, najvyššie aktívne IPv4 rozhranie
  - Ak nie je možné RouterID zistiť, je ho nutné nakonfigurovať ručne

```
Router(config)# ipv6 router ospf 1  
Router(config-rtr)# router-id 158.193.138.255
```



# Povolenie OSPFv3 na rozhraní

- Spustenie OSPFv3 procesu na rozhraní

```
Router(config-if)#
```

```
ipv6 ospf process-id area area-id [instance instance-id]
```

Parameter	Description
<i>process-id</i>	Internal identifier for the OSPF process that is locally assigned and can be any positive integer.
<i>area-id</i>	Specifies the area that is to be associated with the OSPF interface.
<i>instance-id</i>	(Optional) Used to control selection of other routers as neighboring routers. Router becomes neighbors only with routers that have the same instance ID.

# Ladenie OSPFv3

- OSPFv3 cost

```
Router(config)# ipv6 ospf cost interface-cost
```

- OSPFv3 priority

```
Router(config)# ipv6 ospf priority number-value
```

- OSPFv3 stub/totally area

```
Router(config-rtr)# area area-id stub [no-summary]
```

- OSPFv3 sumarizácia

```
Router(config-rtr)# area area-id range ipv6-prefix /prefix-length [advertise | not-advertise] [cost cost]
```

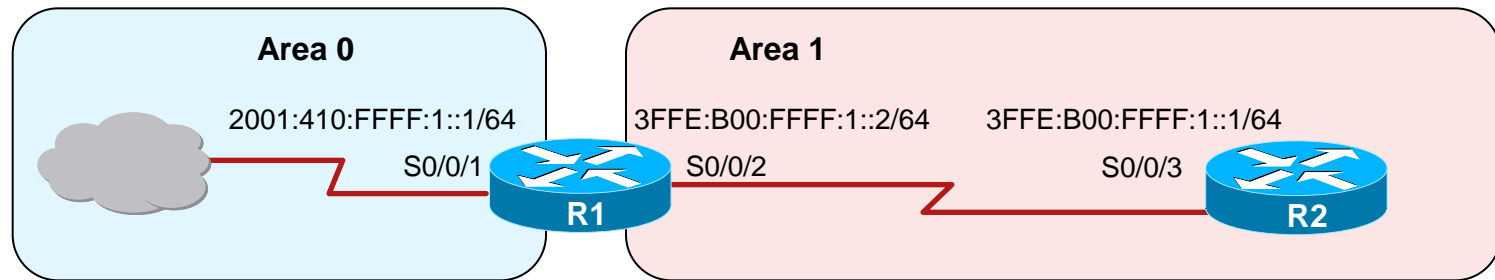
- Premazanie OSPFv3 procesu

```
Router# clear ipv6 ospf [process-id] {process | force-spf | redistribution | counters [neighbor [neighbor-interface | neighbor-id]]}
```

# Overenie a diagnostika OSPFv3

Command	Description
<pre>show ipv6 ospf [process-id] [area-id] neighbor [interface- type interface-number] [neighbor-id] [detail]</pre>	Displays OSPFv3 neighbor information.
<pre>show ipv6 ospf [process-id] [area-id] interface [type number] [brief]</pre>	Displays OSPFv3 interface information.
<pre>show ipv6 ospf [process-id] [area-id]</pre>	Displays general information about the IPv6 OSPF processes.

# Príklad konfigurácie OSPFv3



```
R1(config)# ipv6 router ospf 100
R1(config-rtr)# router-id 10.1.1.3
R1(config-rtr)# area 0 range 2001:410::/32
R1(config-rtr)# exit
R1(config)# interface Serial0/0/1
R1(config-if)# ipv6 address 2001:410:FFFF:1::1/64
R1(config-if)# ipv6 ospf 100 area 0
R1(config-if)# exit
R1(config)# interface Serial0/0/2
R1(config-if)# ipv6 address 3FFE:B00:FFFF:1::2/64
R1(config-if)# ipv6 ospf 100 area 1
R1(config-if)#
```

```
R2(config)# ipv6 router ospf 100
R2(config-rtr)# router-id 10.1.1.4
R2(config-rtr)# exit
R2(config)# interface Serial0/0/3
R2(config-if)# ipv6 address 3FFE:B00:FFFF:1::1/64
R2(config-if)# ipv6 ospf 100 area 1
R2(config-if)#
```



# Multiprotocol BGP

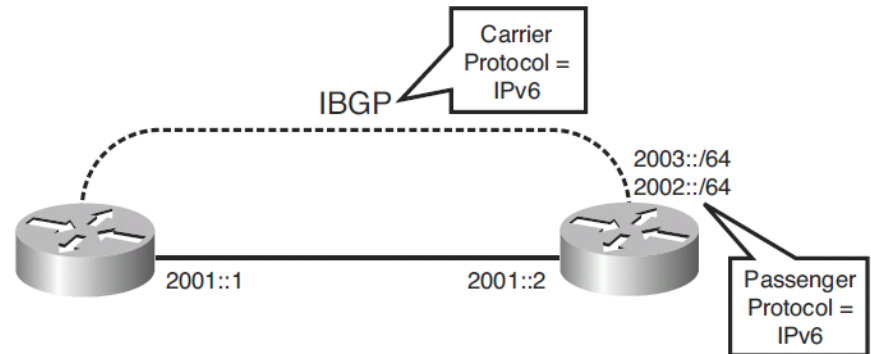
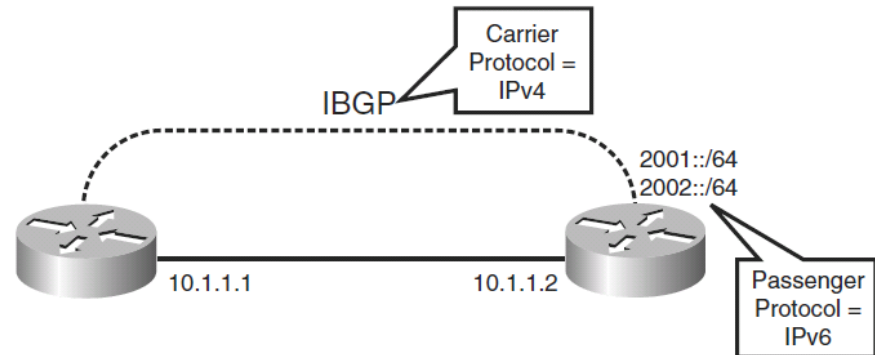
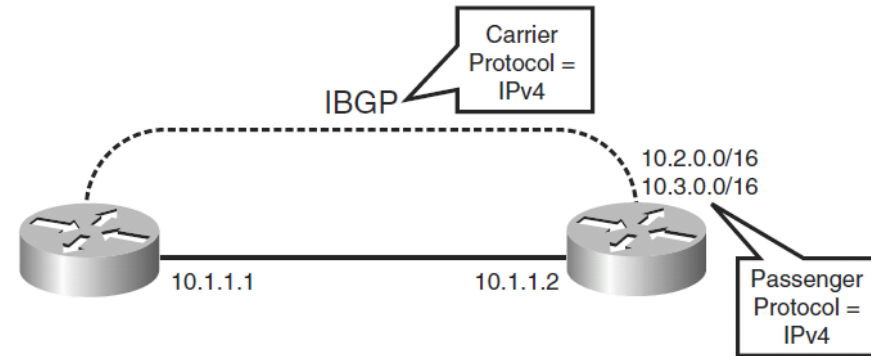


# Multiprotocol Border Gateway Protocol (MP-BGP) (RFC 2858)

- Multiprotokolové rozšírenia pre BGPv4:
  - Podpora rôznych sieťových protokolov
    - MPLS, IPv6 (RFC2545)
  - Identifikátory rodín sieťových protokolov
- MBGP odlišuje
  - Carrier protokol
    - transport
  - Passenger protokol
    - payload
- Špecifické rozšírenia pre IPv6:
  - Identifikátor pre rodinu IPv6 protokolov
  - NEXT\_HOP obsahuje globálnu IPv6 adresu a prípadne aj link-local adresu (len ak sused je priamo pripojený)
  - NEXT\_HOP a NLRI (Network Layer Reachability Information) sú vyjadrené ako IPv6 adresy v atribútoch MP-BGP

# Multiprotocol BGP (MP-BGP)

- BGP konf 1
  - Carrier: IPv4
  - Passenger: IPv4
- BGP konf 2
  - Carrier: IPv4
  - Passenger: IPv6
- BGP konf 3
  - Carrier: IPv6
  - Passenger: IPv6



# Konfigurácia MBGP

- Konfigurácia MBGP pre IPv6 je veľmi podobná konfigurácii BGP pre IPv4
  - Vyžaduje však definovanie carrier a passenger protokolu
- Pre IPv6 je tiež vyžadované Router ID
  - Vo formáte IPv4 adresy

```
Router(config)# router bgp 65000  
Router(config-router)# bgp router-id 158.193.152.108
```



# MBGP Neighbors – Carrier proto

- Definuj BGP peer-a

```
Router(config-router)#
```

```
neighbor {ipv6-address / peer-group-name} remote-as  
autonomous-system-number
```

- Za účelom aktivácie BGP susedskej relácie
  - Internal IBGP s internými susedmi
  - External BGP (EBGP) s externými susedmi
- Definuje *carrier* protocol

```
Router(config-router)# neighbor 2001::1 remote-as 65000  
Router(config-router)# neighbor 2002::1 remote-as 65001
```

iBGP

eBGP

# Address Family – passenger proto

- Definuje passenger protokol, tu IPv6, ktorého prefixy sa budú vymieňať

```
Router(config-router)#
```

```
address-family ipv6 [unicast | multicast | vpnv6]
```

- A umožňuje vstup do podrežimu.

Parameter	Description
<code>unicast</code>	(Optional) Specifies IPv6 unicast address prefixes.
<code>multicast</code>	(Optional) Specifies IPv6 multicast address prefixes.
<code>vpnv6</code>	(Optional) Specifies VPN version 6 address prefixes.

# Identifikácia MBGP suseda pre danú adresnú rodinu

- Definuje BGP peer-a

Router(config-router)#    **or**    Router(config-router-af)#

***neighbor ipv6-address activate***

- Spúšťa výmenu informácií s peerom
- *ipv6-address* je IPv6 adresa suseda

# Príklad MBGP

```
R1(config)# router bgp 1
R1(config-router)# bgp router-id 1.1.1.1
R1(config-router)# neighbor 2001:100:2:4::1 remote-as 100
R1(config-router)#
R1(config-router)# address-family ipv6
R1(config-router-af)# neighbor 2001:100:2:4::1 activate
R1(config-router-af)# redistribute connected
R1(config-router-af)# end
R1#
```

- Carrier protocol je IPv6, lebo konfigurácia obsahuje:
  - Konfiguráciu 32-bit router ID (musí byť len v Ipv6 sieťach)
  - Konfiguráciu BGP peering suseda s použitím IPv6 adresy.
- Passenger protocol je tiež IPv6, lebo konfigurácia obsahuje:
  - Zadefinovanú adresnú rodinu
  - Identifikáciu MBGP suseda s jeho IPv6 adresou



# IPv6 PBR



# IPv6 Policy-Based Routing (PBR)

- Príkaz

```
route-map map-tag [permit | deny] [sequence-number]
```

Je rovnaký ako pre IPv4

- IPv6 PBR tiež používa kritéria a akcie:
  - **match** identifikácia prevádzky pre PBR
  - **set** nastavenie akcie.

# Príkazy match vhodné pre IPv6 PBR

Command	Description
<code>match ipv6 address</code>	Matches any routes that have a destination network number IPv6 address that is permitted by a standard or extended ACL
<code>match ipv6 next-hop</code>	Matches any routes that have a next-hop router IPv6 address that is passed by one of the ACLs specified
<code>match ipv6 route-source</code>	Matches routes that have been advertised by routers and access servers at the IPv6 address that is specified by the ACLs
<code>match community</code>	Matches a BGP community
<code>match interface</code>	Matches any routes that have the next hop out of one of the interfaces specified
<code>match length</code>	Matches based on the layer 3 length of a packet
<code>match metric</code>	Matches routes with the metric specified
<code>match route-type</code>	Matches routes of the specified type
<code>match tag</code>	Matches tag of a route

# Príkazy set vhodné pre IPv6 PBR

Command	Description
<code>set ipv6 default next-hop</code>	Indicates an IPv6 default next hop to which matching packets will be forwarded
<code>set ipv6 next-hop</code>	Indicates where to output IPv6 packets that pass a <code>match</code> clause of a route map for policy routing
<code>set ipv6 precedence</code>	Set the precedence value in the IPv6 packet header
<code>set as-path</code>	Modifies an AS path for BGP routes
<code>set automatic-tag</code>	Computes automatically the tag value
<code>set community</code>	Sets the BGP communities attribute
<code>set default interface</code>	Indicates where to output packets that pass a match clause of a route map for policy routing and have no explicit route to the destination
<code>set interface</code>	Indicates where to output packets that pass a match clause of a route map for policy routing
<code>set local-preference</code>	Specifies a BGP local preference value
<code>set metric</code>	Sets the metric value for a routing protocol
<code>set metric-type</code>	Sets the metric type for the destination routing protocol
<code>set tag</code>	Sets tag value for destination routing protocol
<code>set weight</code>	Specifies the BGP weight value



# Aplikácia PBR route mapy

- Na rozhranie

```
Router(config-if)#
```

```
ipv6 policy route-map route-map-name
```

- Aj pre lokálne generovanú prevádzku

```
Router(config)#
```

```
ipv6 local policy route-map route-map-name
```

# IPv6 ACL

- Vstup do IPv6 ACL konfiguračného režimu

```
Router(config)#
```

```
ipv6 access-list access-list-name
```

- IPv4 ACL a IPv6 ACL nemôžu mať rovnaké meno
- IPv6 ACL nepodporuje číslované zoznamy
- Ďalej nasledujú **permit** a **deny** podmienky na vyhodnocovanie :

```
permit protocol {source-ipv6-prefix/prefix-length | any  
| host source-ipv6-address} {destination-ipv6-  
prefix/prefix-length | any | host destination-ipv6-  
address}
```

# IPv6 Ping so zmenou Default Source rozhrania

- Príkaz špecifikuje, adresa ktorého rozhrania sa použije pre daný ping

```
Router# ping ipv6 ipv6-address source interface-name
```

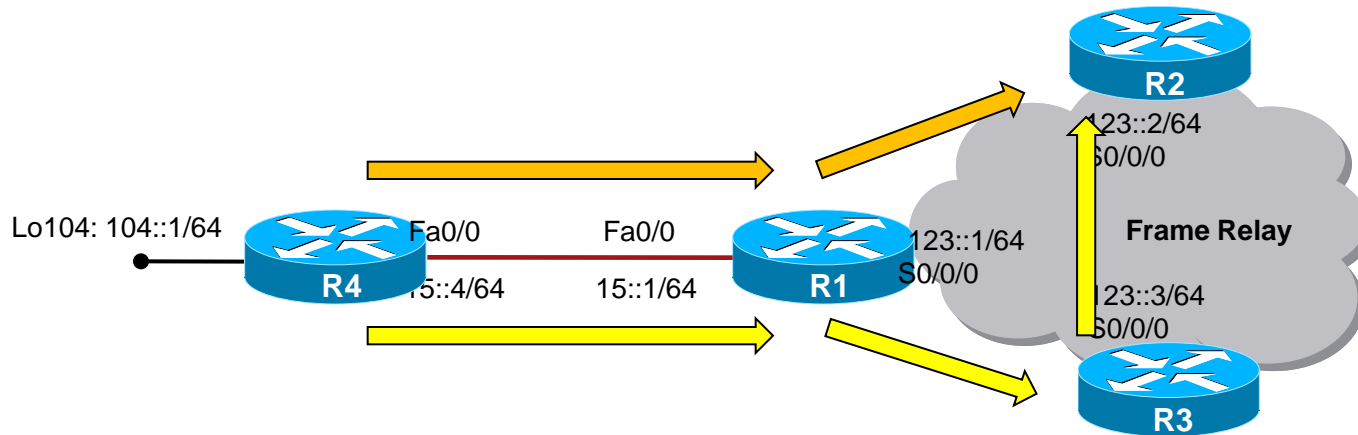
```
Router# ping ipv6-address source interface-name
```

- Default správanie je použiť ako source IP adresu výstupného rozhrania pre ICMP správy

# Overenie a diagnostika IPv6 PBR

Command	Description
<code>show ipv6 access-list [access-list-name]</code>	Displays the contents of all or a specified IPv6 ACL. The <i>access-list-name</i> parameter specifies the name of the access list.
<code>show route-map [map-name]</code>	Displays configured IPv4 and IPv6 route maps The <i>map-name</i> is an optional name of a specific route map.
<code>debug ipv6 policy [access-list-name]</code>	Displays IPv6 policy routing packet activity. The <i>access-list-name</i> parameter specifies the name of the access list.

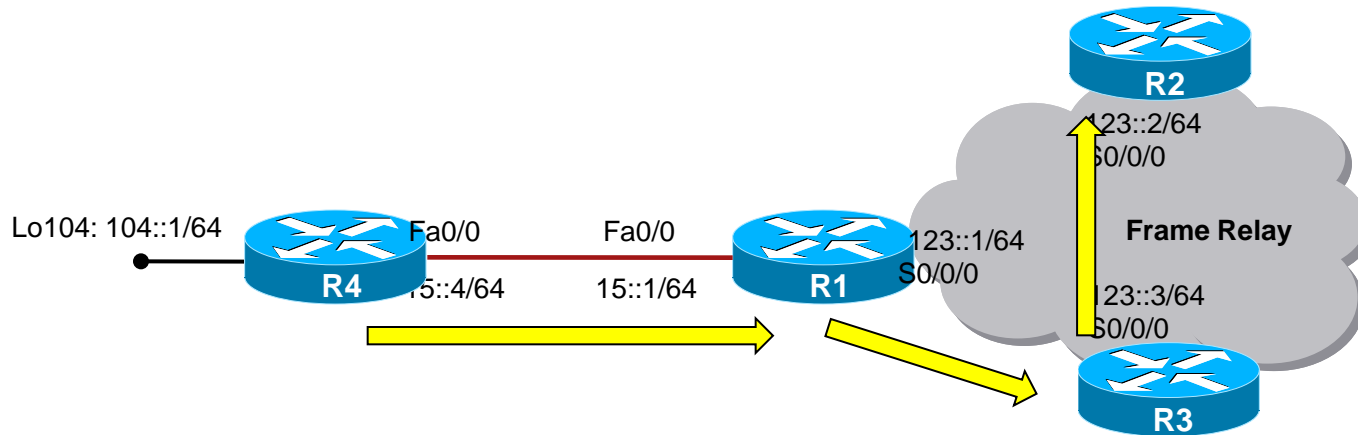
# Príklad IPv6 PBR



```
R1(config)# route-map PBR-SOURCE-ADDRESS permit 10
R1(config-route-map)# match ipv6 address SOURCE-104
R1(config-route-map)# set ipv6 next-hop 123::3
R1(config-route-map)# exit
R1(config)# ipv6 access-list SOURCE-104
R1(config-ipv6-acl)# permit ipv6 104::/64 any
R1(config-ipv6-acl)# exit
R1(config)# interface fa0/0
R1(config-if)# ipv6 policy route-map PBR-SOURCE-ADDRESS
R1(config-if)#
```

- Sieť používa RIPng
- Premávka ide default oranžová (R1 =>R2)
- Mapa mení smerovanie pre source net 104::/64 na R1=> R3 (žltá)

# Príklad IPv6 PBR - overenie

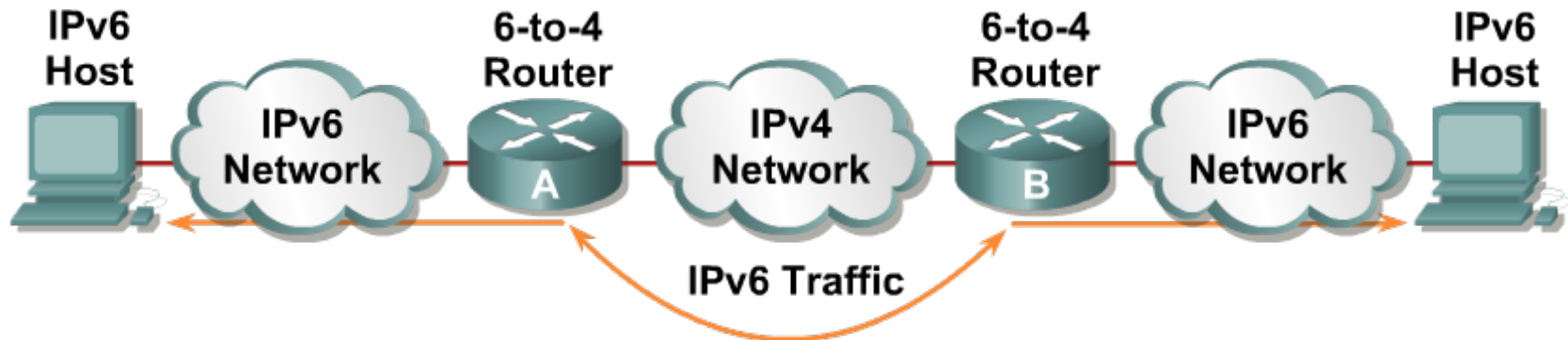


```
R1# show route-map
route-map PBR-SOURCE-ADDRESS, permit, sequence 10
  Match clauses:
    ipv6 address SOURCE-104
  Set clauses:
    ipv6 next-hop 123::3
  Policy routing matches: 5 packets, 500 bytes
R1#
R1# show ipv6 access-list
IPv6 access list SOURCE-104
  permit ipv6 104::/64 any (5 matches) sequence 10
R1#
```

# Migrácia medzi IPv4 a IPv6



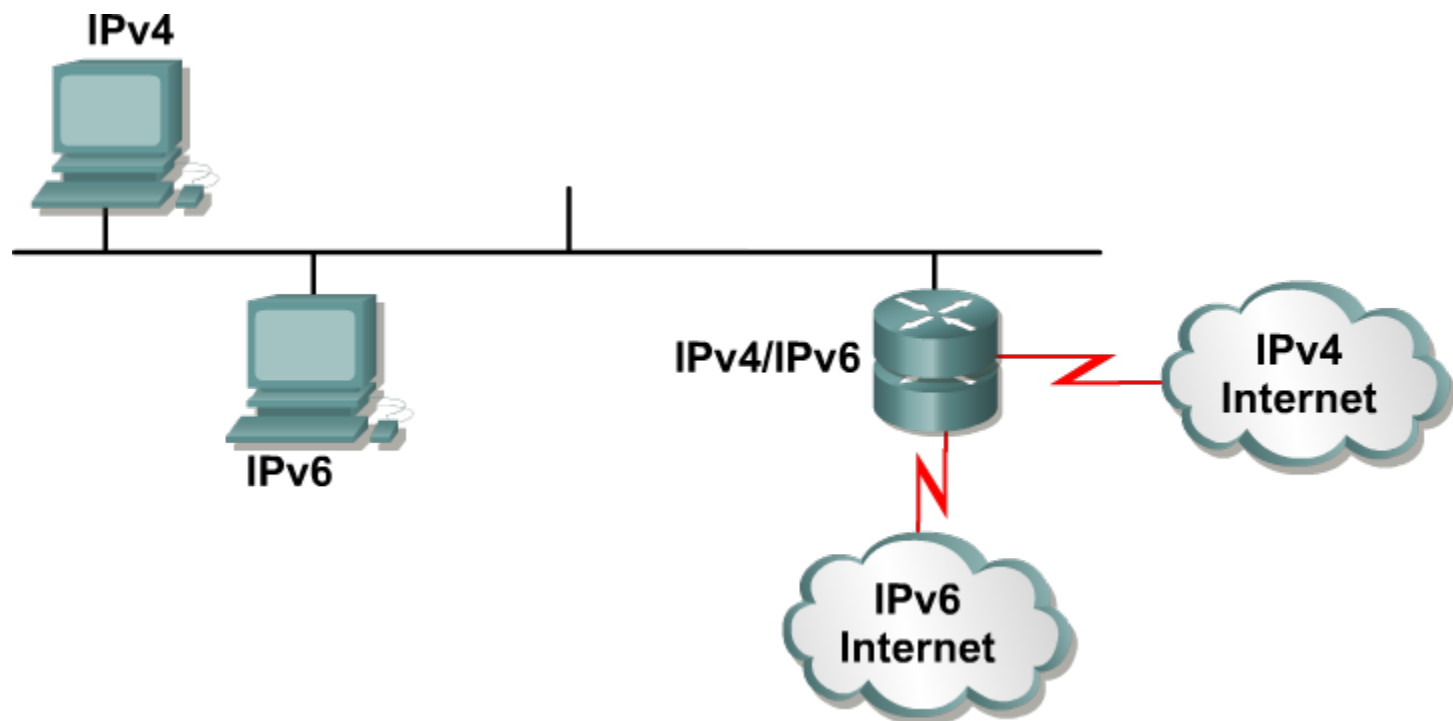
# Migrácia medzi IPv4 a IPv6



- Pre migráciu z IPv4 na IPv6 je definovaných niekoľko rôznych mechanizmov, a nie je nevyhnutné urobiť skokový prechod
- Migračné mechanizmy
  - Dual-stack
  - Tunneling
    - Statické tunely, GRE, 6over4 tunely (zriedka používané, RFC 2529), 6to4 tunely (RFC 3056)
    - ISATAP tunely (RFC 4214)
    - Teredo tunely (RFC 4380)
  - Preklady (translations)
    - NAT-PT (Protocol Translation)

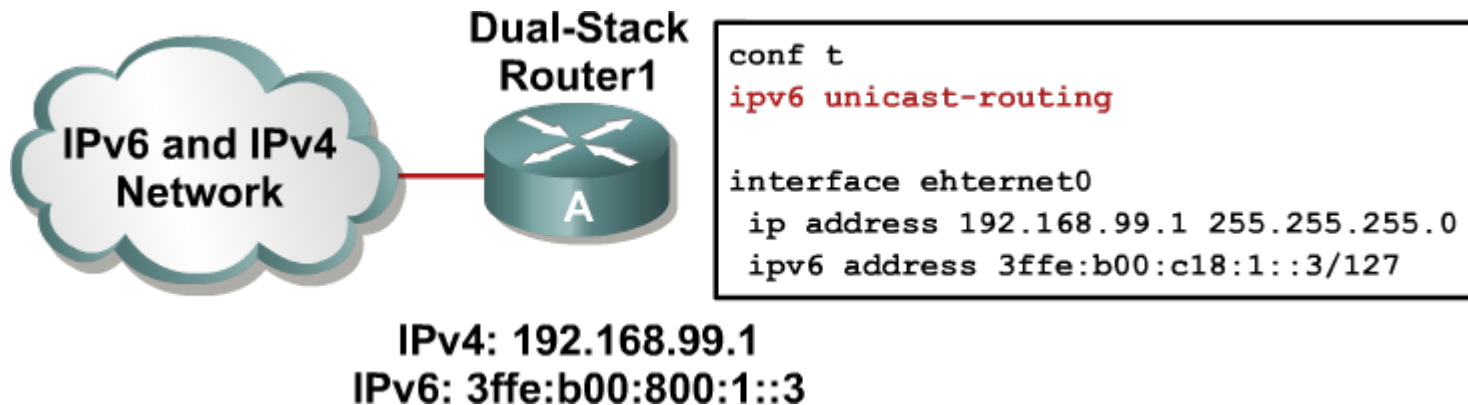


# Dual Stack



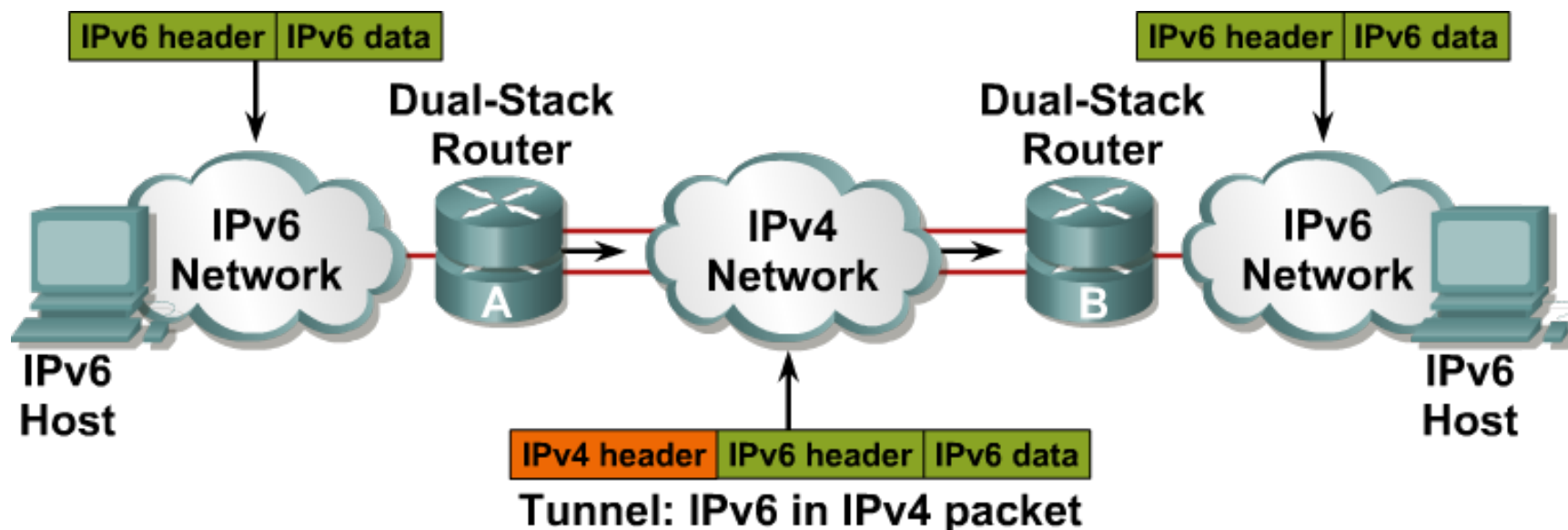
- Dual stack je integračná metóda, kde každá stanica aj router implementuje aj IPv4, aj IPv6
  - Na tom istom or viacerých rozhraniach
- Protokoly sú na sebe úplne nezávislé
- Nevýhoda v zvýšených nárokoch na zdroje
  - Napr. duplicita smerovacích a topo tabuliek.

# Dual stack spôsob konfigurácie



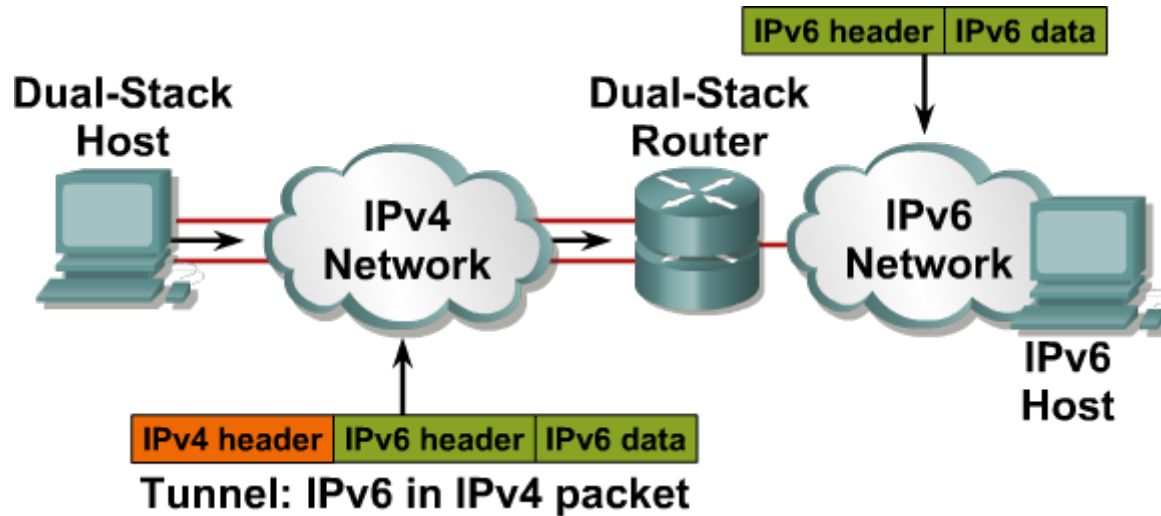
- Stačí nakonfigurovať na rozhraní IPv4 aj IPv6 adresu
- Smerovač sa rozhodne na základe cieľovej adresy
  - IPV6 je preferované

# Tunelovanie IPv6 v IPv4



- Tunelovanie je integračná metóda, kde sa IPv6 paket zapúzdri do paketu iného protokolu, napr. IPv4 (číslo protokolu 41)
  - Overhead tohto tunela je 20B (hlavička IPv4 bez voliteľných častí)
  - IPv6 je v roli passenger protokolu a IPv4 je transport (carrier)
  - Tento spôsob vyžaduje dual-stacked routery (na okrajoch)

# Tunelovanie IPv6 v IPv4

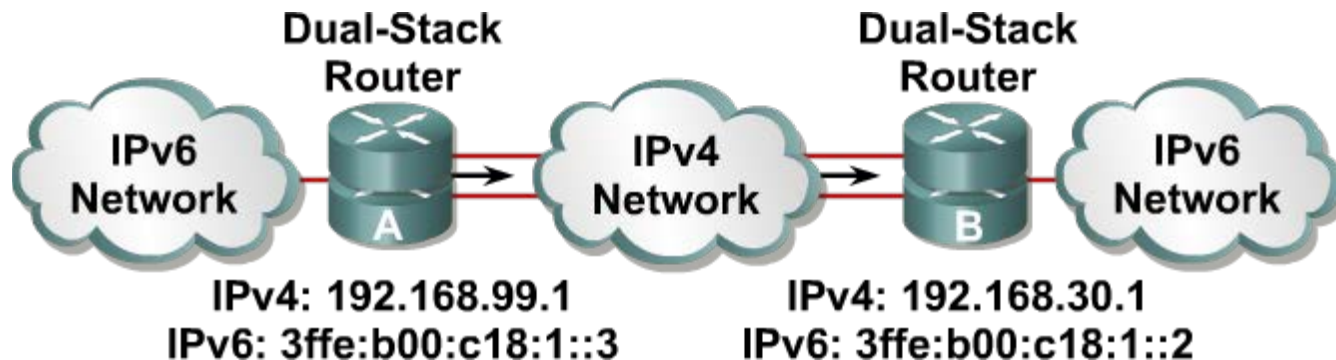


- Zapúzdrenie môže realizovať nielen router, ale aj stanica, ak jej operačný systém ovláda príslušný spôsob tunelovania

# Typy tunelov

- Tunely môžu byť vytvorené:
  - Staticky
    - Statické (manual) IPv6 tunely (p-t-p tunel)
      - Podpora mcast - routing
    - GRE IPv6 tunely (p-t-p tunel)
      - Podpora mcast - routing
  - Automaticky:
    - IPv4-Compatible IPv6 Tunely (potláčané)
    - 6to4 tunely (p-t-m tunel)
      - Podpora len unicast (bgp, statika)
    - ISATAP tunely (Intra-Site Automatic Tunnel Addressing Protocol) (p-t-p tunel)
      - Podpora len unicast (bgp, statika)

# Konfigurácia statických tunelov



- Konfigurácia statických tunelov vyžaduje:
  - Dual-stack endpointy
  - IPv4 a IPv6 adresy konfigurované na oboch koncoch tunela
    - IPv4 adresy tunela sú z iných subnetov, IPv6 z toho istého
  - Platný IPv4 routing medzi zdrojom a cieľom tunela
- Výhoda v jednoduchosti, vhodné pre malé siete
- Nevýhoda v praxi pre komplikované topológie

# Konfigurácia statických tunelov

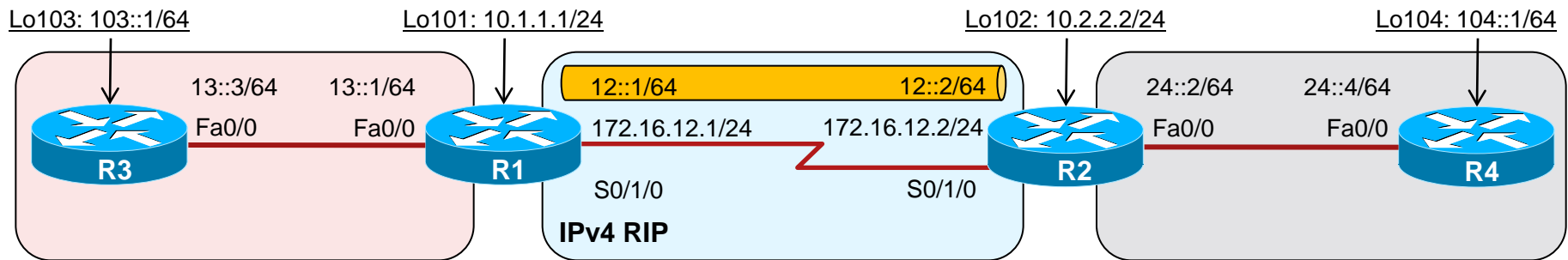
- Ako pri GRE tuneloch (kap7)

```
Router(config)#
```

```
interface tunnel number
```

- Príkaz vytvorí virtuálne rozhranie tunela
- Keď je rozhranie vytvorené, vykoná sa konfigurácia jeho parametrov ako
  - IPv6 adresa
  - Tunnel source (IPv4)
  - Tunnel destination (IPv4)
  - Tunnel mode (typ tunela)
    - ipv6ip

# Príklad na statické IPv6 tunely

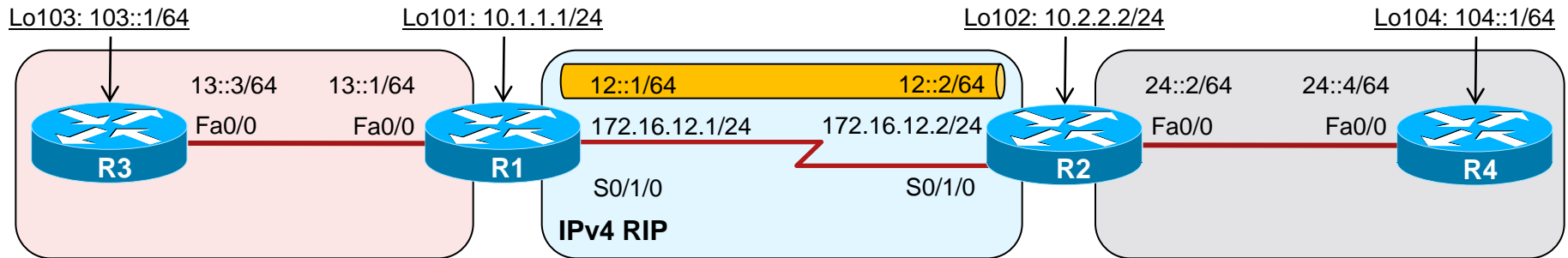


```
R1(config)# interface tunnel 12
R1(config-if)#
*Aug 16 09:34:46.643: %LINEPROTO-5-UPDOWN: Line
protocol on Interface Tunnell2,
changed state to down
R1(config-if)# no ip address
R1(config-if)# ipv6 address 12::1/64
R1(config-if)# tunnel source loopback 101
R1(config-if)# tunnel destination 10.2.2.2
R1(config-if)#
*Aug 16 09:36:52.051: %LINEPROTO-5-UPDOWN: Line
protocol on Interface Tunnell2,
changed state to up
R1(config-if)# tunnel mode ipv6ip
R1(config-if)# exit
R1(config)# ipv6 unicast-routing
R1(config)# interface tunnel 12
R1(config-if)# ipv6 rip RIPv6 enable
R1(config-if)# interface fa0/0
R1(config-if)# ipv6 rip RIPv6 enable
R1(config-if)#
```

```
R2(config)# interface tunnel 12
R2(config-if)#
*Aug 16 09:38:47.532: %LINEPROTO-5-UPDOWN: Line
protocol on Interface Tunnell2,
changed state to down
R2(config-if)# no ip address
R2(config-if)# ipv6 address 12::2/64
R2(config-if)# tunnel source loopback 102
R2(config-if)# tunnel destination 10.1.1.1
R2(config-if)#
*Aug 16 09:39:24.056: %LINEPROTO-5-UPDOWN: Line
protocol on Interface Tunnell2,
changed state to up
R2(config-if)# tunnel mode ipv6ip
R2(config-if)# exit
R2(config)# ipv6 unicast-routing
R2(config)# interface tunnel 12
R2(config-if)# ipv6 rip RIPv6 enable
R2(config-if)# interface fa0/0
R2(config-if)# ipv6 rip RIPv6 enable
R2(config-if)#
```



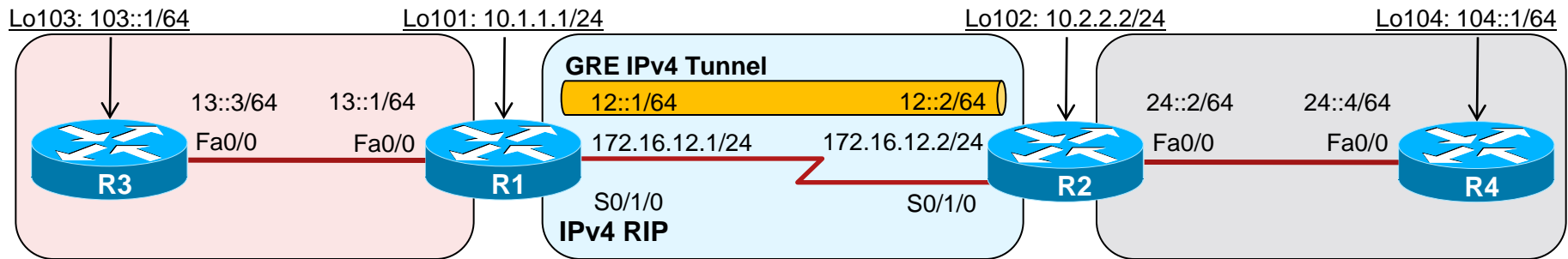
# Overenie



```
R1# show interface tunnel 12
Tunnel12 is up, line protocol is up
  Hardware is Tunnel
  MTU 1514 bytes, BW 9 Kbit/sec, DLY 500000 usec,
    reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation TUNNEL, loopback not set
  Keepalive not set
  Tunnel source 10.1.1.1 (Loopback101), destination 10.2.2.2
  Tunnel protocol/transport IPv6/IP
  Tunnel TTL 255
  Fast tunneling enabled

<output omitted>
```

# Príklad na GRE tunely



```
R1(config)# interface tunnel 12
R1(config-if)#
*Aug 16 09:34:46.643: %LINEPROTO-5-UPDOWN: Line
protocol on Interface Tunnel12,
changed state to down
R1(config-if)# no ip address
R1(config-if)# ipv6 address 12::1/64
R1(config-if)# tunnel source loopback 101
R1(config-if)# tunnel destination 10.2.2.2
R1(config-if)#
*Aug 16 09:36:52.051: %LINEPROTO-5-UPDOWN: Line
protocol on Interface Tunnel12,
changed state to up
R1(config-if)# tunnel mode ipv6ip
R1(config-if)# tunnel mode gre ! Nie je treba
R1(config-if)# exit
R1(config)# ipv6 unicast-routing
R1(config)# interface tunnel 12
R1(config-if)# ipv6 rip R1PoTU enable
R1(config-if)# interface fa0/0
R1(config-if)# ipv6 rip R1PoTU enable
R1(config-if)#
```

```
R2(config)# interface tunnel 12
R2(config-if)#
*Aug 16 09:38:47.532: %LINEPROTO-5-UPDOWN: Line
protocol on Interface Tunnel12,
changed state to down
R2(config-if)# no ip address
R2(config-if)# ipv6 address 12::2/64
R2(config-if)# tunnel source loopback 102
R2(config-if)# tunnel destination 10.1.1.1
R2(config-if)#
*Aug 16 09:39:24.056: %LINEPROTO-5-UPDOWN: Line
protocol on Interface Tunnel12,
changed state to up
R2(config-if)# tunnel mode ipv6ip
R2(config-if)# tunnel mode gre ! Nie je treba
R2(config-if)# exit
R2(config)# ipv6 unicast-routing
R2(config)# interface tunnel 12
R2(config-if)# ipv6 rip R2PoTU enable
R2(config-if)# interface fa0/0
R2(config-if)# ipv6 rip R2PoTU enable
R2(config-if)#
```

# 6to4 tunely (RFC 3056)

- 6to4 tunely môžu mať mnoho koncových bodov
  - na rozdiel od trvalých statických tunelov) tunely
- IPv6 prefixy jednotlivých IPv6 ostrovov oddelených IPv4 internetom sú navrhnuté tak, aby v sebe obsahovali priamo IPv4 adresu tunelujúceho routera, ktorý je na okraji tohto ostrova
  - IPv6 adresy pri použití 6to4 tunelov využívajú striktne prefix **2002::/16**
  - Ďalších 32 bitov = IPv4 adresa smerovača
    - ktorý je na vstupe/výstupe nášho IPv6 ostrova a ktorý realizuje tunelovanie
  - Výsledný 48-bitový prefix je prefix spoločný pre celý IPv6 ostrov.
  - Zostáva 16 bitov pre Subnet ID a 64 bitov pre Interface ID
    - Rovnako ako v bežných Global Unicast adresách
  - Tunel destination nie je treba
    - Tunely sú vytvárané automaticky edge smerovačmi

# 6to4 tunely (RFC 3056)

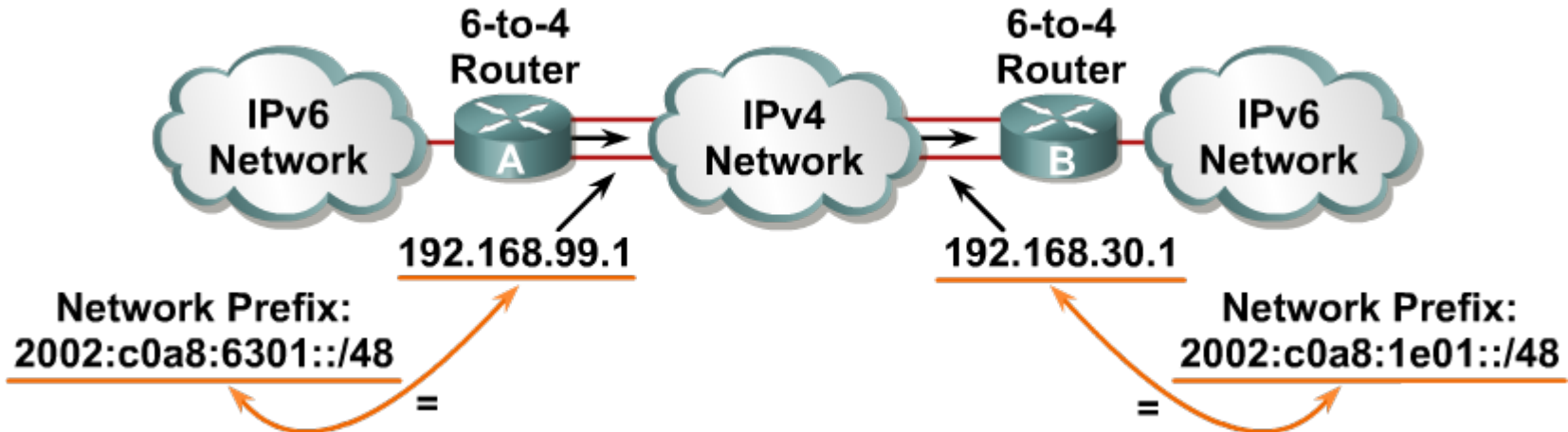
## ■ Príklad:

- Router na vstupe do nášho IPv6 ostrovčeka má verejnú IPv4 adresu 192.168.30.1
- Hexadecimálny prepis tejto adresy je **C0A8:1E01**
- Všetky IPv6 zariadenia v našom ostrovčeku majú teda IPv6 prefix **2002:C0A8:1E01::/48**
- Routery na susedných IPv6 lokalitách musia mať akurát vhodne nastavené smerovanie, aby pre prístup do IPv6 sietí s prefixom 2002::/16 používali 6to4 tunel
- Tu je aj nevýhoda:
  - Podporované sú statické cesty a BGP, lebo IGP používajú ako next hop link-local adresu

## ■ Postup konfigurácie:

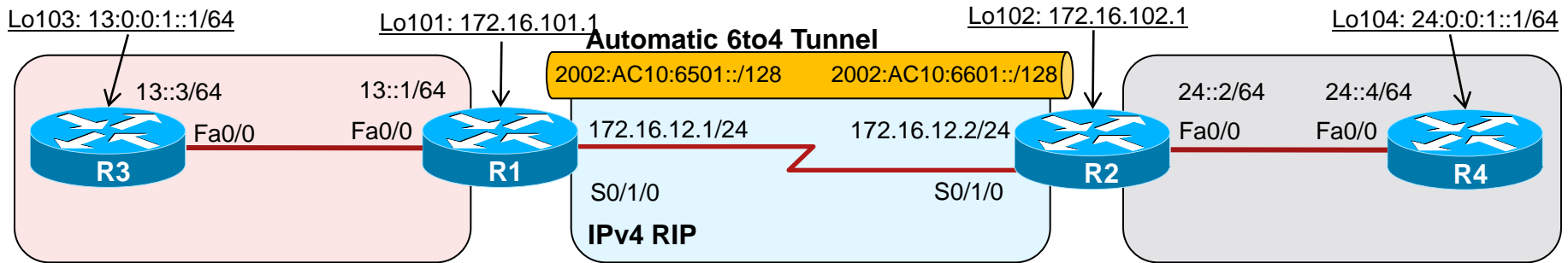
- Vytvoriť rozhranie pre tunel
- Nastaviť režim tunela: **tunnel mode ipv6ip 6to4**
- Vytvoriť primeraný adresový plán a nastaviť IPv6 adresy na rozhraniach
- Nastaviť vhodnú IPv6 adresu na tuneli, alebo si ju vypožičať cez príkaz `unnumbered`
- Nastaviť IPv6 smerovanie cez tunel

# 6to4 tunely (RFC 3056)



- 6to4:
  - Na tuneloch nie je potrebné definovať endpoint (druhý koniec)
  - Každý IPv6 ostrov má jednoznačný globálne platný prefix
- Keď A prijme paket do siete 2002:c0a8:1e01::/48, určí že paket musí ísť do tunela
  - Smerovač extrahuje z IPv6 adresy IPv4 adresu konca tunela
  - Zabalí IPv6 paket do IPv4 paketu, kde cieľová IP je IPv4 adresa B
  - Pošle paket do tunela
- Smerovač B vykoná deenkapsuláciu

# Príklad konfigurácie: 6to4 tunel



```
R1(config)# interface tunnel 12
R1(config-if)#
R1(config-if)# no ip address
R1(config-if)# ipv6 address
2002:AC10:6501::/128
R1(config-if)# tunnel source loopback 101
R1(config-if)# tunnel mode ipv6ip 6to4
R1(config-if)#
R1(config-if)# exit
R1(config)# ipv6 unicast-routing
R1(config)# ipv6 route 2002::/16 tunnel 12
R1(config)# ipv6 route 24::/48 2002:AC10:6601::
R1(config)#
```

```
R2(config)# interface tunnel 12
R2(config-if)#
R2(config-if)# no ip address
R2(config-if)# ipv6 address
2002:AC10:6601::/128
R2(config-if)# tunnel source loopback 102
R2(config-if)# tunnel mode ipv6ip 6to4
R2(config-if)#
R2(config-if)# exit
R1(config)# ipv6 unicast-routing
R2(config)# ipv6 route 2002::/16 tunnel 12
R2(config)# ipv6 route 13::/48 2002:AC10:6501::
```

Next hop oboch smerovačov je IPv6  
adresa prot'ajšieho konca tunela  
Musí byť uvedené takto, aby vedel  
spraviť tunnel na cieľovú IPv4

# 6to4 Relay – prestup do IPv6 sveta

- Systém 6to4 umožňuje spojenie s IPv6 ostrovom cez medziľahlú IPv4 sieť
- Dá sa teda aj použiť na prístup do IPv6 sveta
  - Stačí vhodná brána, ktorá akceptuje 6to4 tunely
  - Na túto bránu budeme smerovať všetko, čo pôjde do IPv6
- RFC 3068 stanovuje, že provideri podľa svojho rozhodnutia môžu takúto bránu vytvoriť
  - Táto brána musí mať adresu 192.88.99.1
  - Smerovače na prístup do IPv6 internetu použijú statický smer

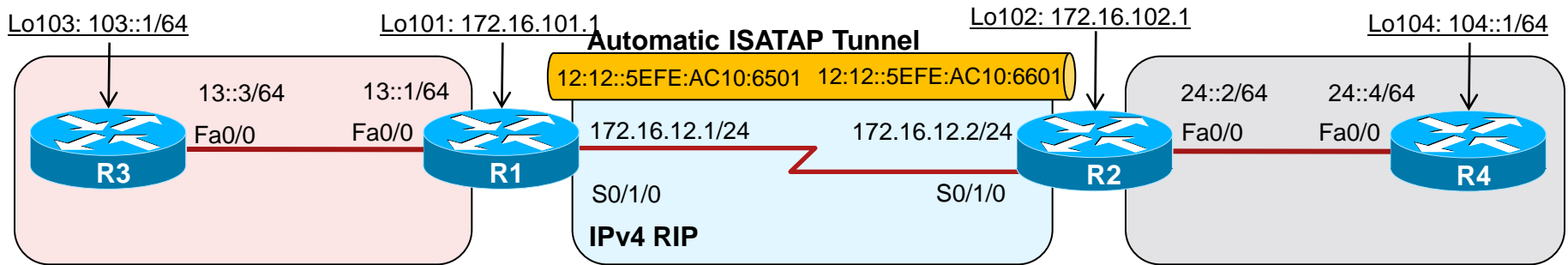
**ipv6 route 2000::/3 2002:c058:6301::**

# ISATAP tunely

- ISATAP (Intra-Site Automatic Tunnel Addressing Protocol) tunely (RFC 5214) sú obdobou 6to4 tunelov
  - Cieľom ISATAP je pripojenie IPv6 hostov cez IPv4 sieť na najbližší IPv6 smerovač
  - Primárne určené na zabezpečenie vnútrofiremnej (intra-site) komunikácie
  - Takisto využívajú mapovanie medzi IPv4 a IPv6 adresou
  - Tvar ISATAP IPv6 adresy:
    - Prvých 64 bitov: link-local alebo global unicast prefix
    - Ďalších 32 bitov: **0000:5EFE** (konštanta)
    - Zvyšných 32 bitov: IPv4 adresa stanice
    - Dĺžka prefixu: 64 bitov
  - Novinka: Využitie pomocných mechanizmov (DHCP, DNS, manuálna konfigurácia) pre lokalizáciu smerovačov schopných zabezpečiť smerovanie medzi rôznymi IPv6 sieťami
- ISATAP nepodporuje multicast
  - Problém s RIPng a OSPFv3



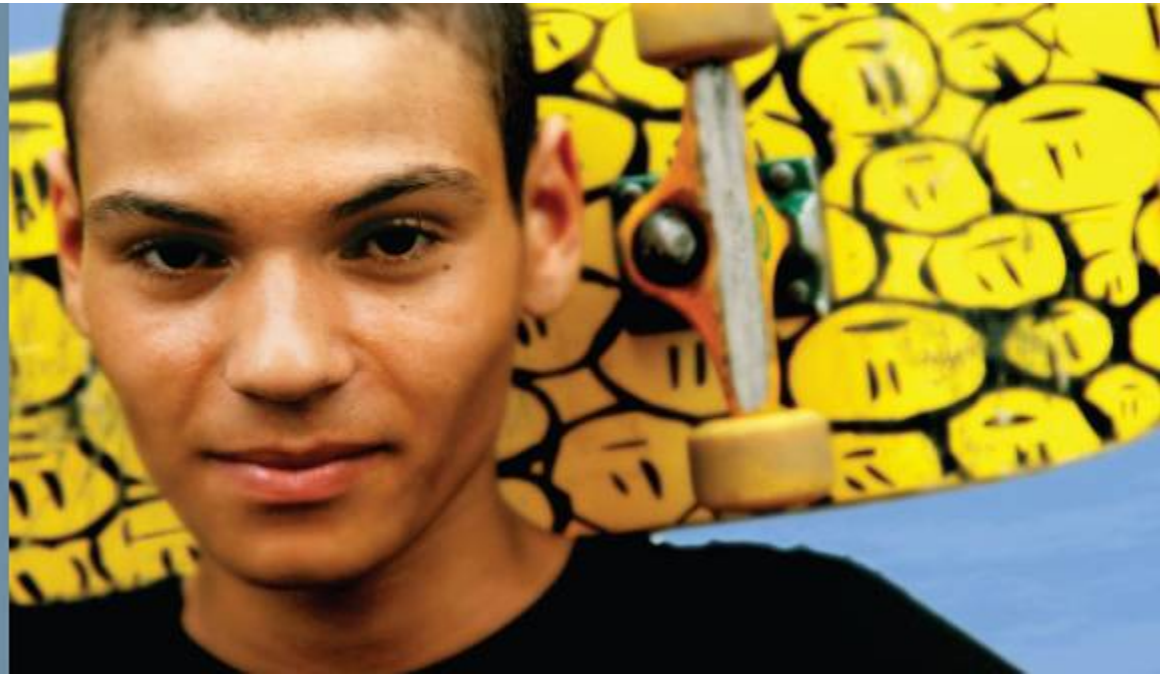
# Príklad konfigurácie: ISATAP



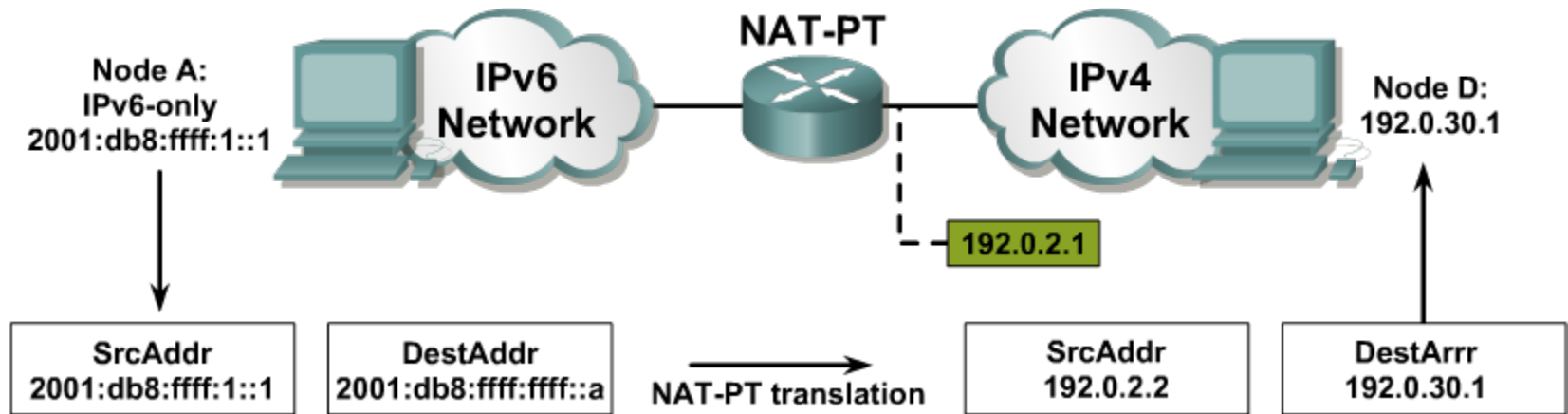
```
R1(config)# interface tunnel 12
R1(config-if)#
R1(config-if)# no ip address
R1(config-if)# ipv6 address 12:12::/64 eui-64
R1(config-if)# tunnel source loopback 101
R1(config-if)# tunnel mode ipv6ip isatap
R1(config-if)# exit
R1(config)# ipv6 route 24::/48 tunnel12
FE80::5EFE:AC10:6601
R1(config)#
```

```
R2(config)# interface tunnel 12
R2(config-if)#
R2(config-if)# no ip address
R2(config-if)# ipv6 address 12:12::/64 eui-64
R2(config-if)# tunnel source loopback 102
R2(config-if)# tunnel mode ipv6ip isatap
R2(config-if)# exit
R2(config)# ipv6 route 13::/48 tunnel12
FE80::5EFE:AC10:6501
R2(config)#
```

# Preklad s NAT- PT

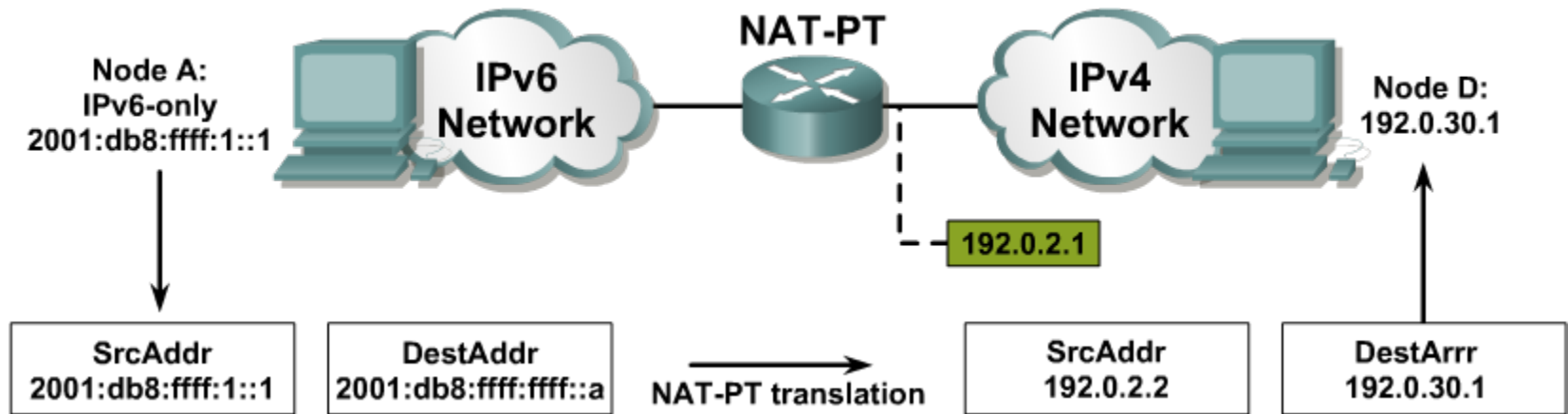


# Preklad protokolov: NAT-PT



- NAT-Protocol Translation (NAT-PT) je prekladový mechanizmus na rozhraní medzi IPv6 a IPv4 sieťou
- Jeho úlohou je prekladať IPv6 pakety na IPv4 a naopak
- Tento prístup je vhodný pre umožnenie spolupráce medzi uzlami, z ktorých jeden je IPv4-only a druhý IPv6-only
  - Nie je vhodný do prostredia prepojenia dvoch IPv6 sietí

# Preklad protokolov: NAT-PT

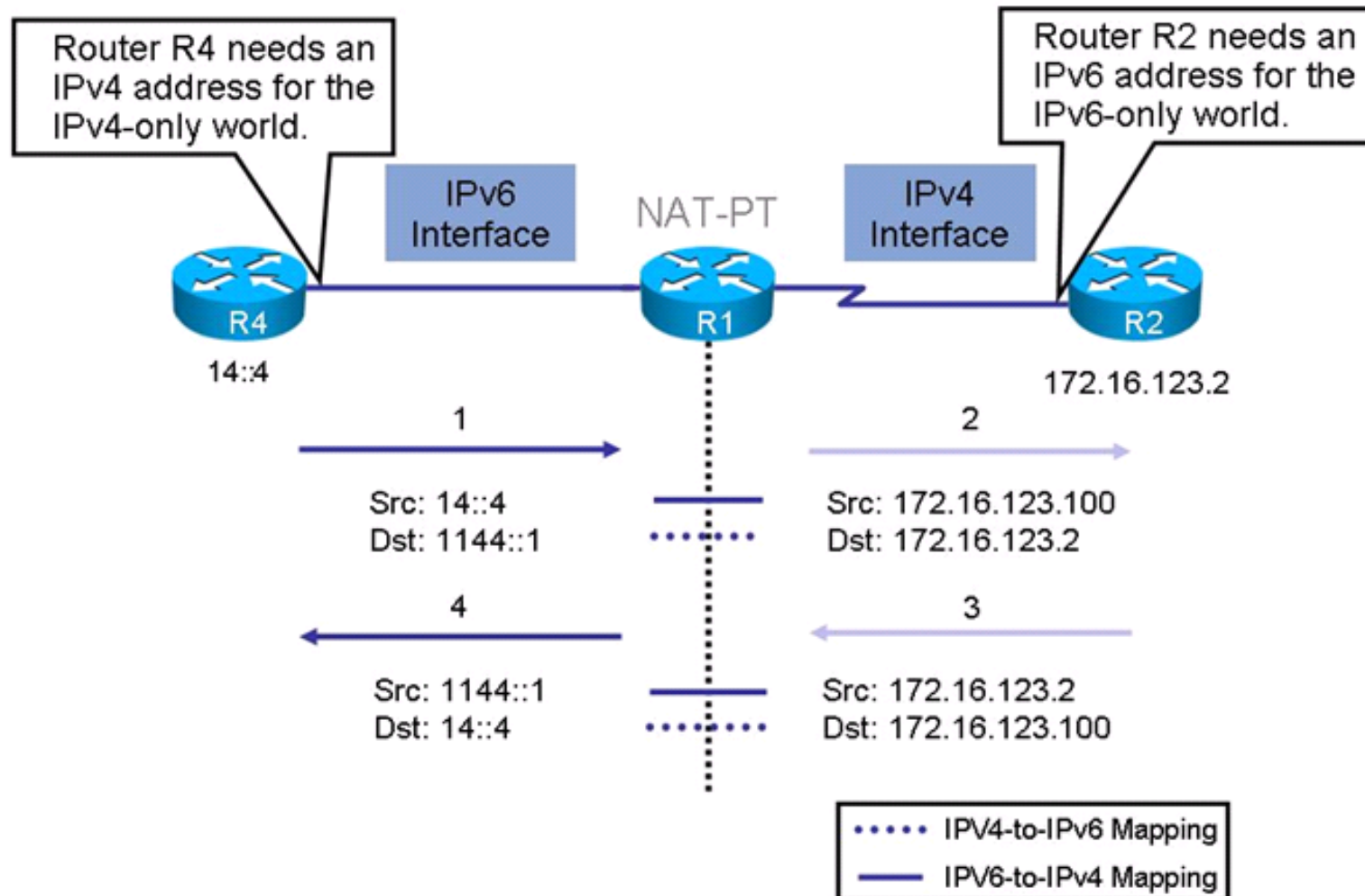


- Smerovač pomocou NAT-PT modifikuje zdrojovú i cieľovú adresu v oboch smeroch toku dát
  - Z IPv4 siete do IPv6 siete
  - Z IPv6 siete do IPv4 siete
- Smerovač musí preto využívať dual stack a mať dve sady prekladov pre obojsmerný preklad
- Komunikujúce stanice nemajú tušenie že druhá strana používa inú protokolovú sadu

# NAT-PT

- Pri NAT-PT využívame **96-bitový** IPv6 sieťový prefix na reprezentáciu celého IPv4 sveta
  - Tento prefix môže byť ľubovoľný dosiahnuteľný prefix v našej IPv6 sieti
  - Akákoľvek IPv4 stanica je reprezentovaná z pohľadu IPv6 sveta ako adresa v tomto prefixe
  - Tento prefix musí byť ohlasovaný zo smerovača, ktorý robí NAT-PT
    - Prevádzka pre tento cieľ musí byť smerovaná na NAT-PT
- Keď príde IPv6 paket na cieľovú adresu v priestore tohto prefixu, smerovač ho transformuje na IPv4 paket
  - Zmení obe adresy – zdroj i cieľ
  - Mapovanie medzi IPv4 a IPv6 adresami je možné konfigurovať staticky alebo dynamicky

# Príklad činnosti NAT-PT



# Konfigurácia statického NAT-PT

- Preklad zdrojovej IPv4 na zdrojovú IPv6 (paket idúci z IPv4 siete do IPv6 siete)

```
Router(config)#
```

```
ipv6 nat v4v6 source ipv4-address ipv6-address
```

- Preklad zdrojovej IPv6 na zdrojovú IPv4 (paket idúci z IPv6 siete do IPv4 siete)

```
Router(config)#
```

```
ipv6 nat v6v4 source ipv6-address ipv4-address
```

- Definovanie IPv6 prefixu reprezentujúceho IPv4 svet

- Priradí sieť do smerovacej tabuľky ako Directly connected (redistribute static)

```
Router(config)#      alebo      Router(config-if)#
```

```
ipv6 nat prefix ipv6-prefix/96
```

- Zaradenie rozhraní do NAT-PT

```
Router(config-if)#
```

```
ipv6 nat
```

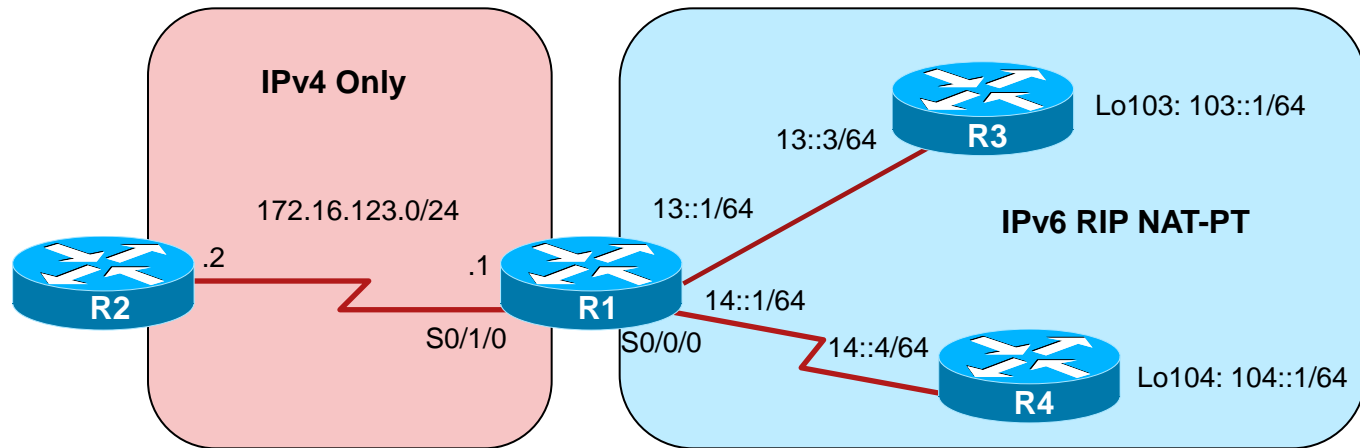
- Špecifikácia Inside a outside rozhrania nie je treba

# Overenie a diagnostika NAT-PT

Command	Description
<code>show ipv6 nat translations</code>	Displays active NAT-PT translations. Each translation is displayed over two lines.
<code>show ipv6 nat statistics</code>	Displays NAT-PT statistics.
<code>debug ip icmp</code>	Displays ICMPv6 events in real time.
<code>debug ipv6 nat</code>	Displays debug messages for NAT-PT translation events

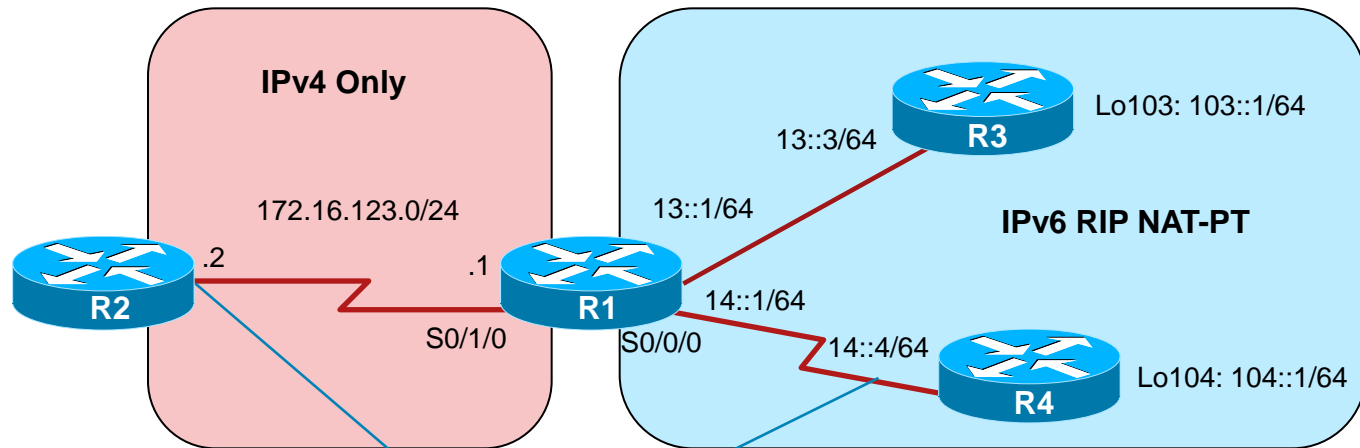


# Príklad pre statické NAT-PT (1)



```
R1(config)# interface s0/0/0
R1(config-if)# ipv6 nat
%LINEPROTO-5-UPDOWN: Line protocol on Interface NVI0, changed state to up
R1(config-if)# interface s0/1/0
R1(config-if)# ipv6 nat
R1(config-if)# exit
R1(config)#
```

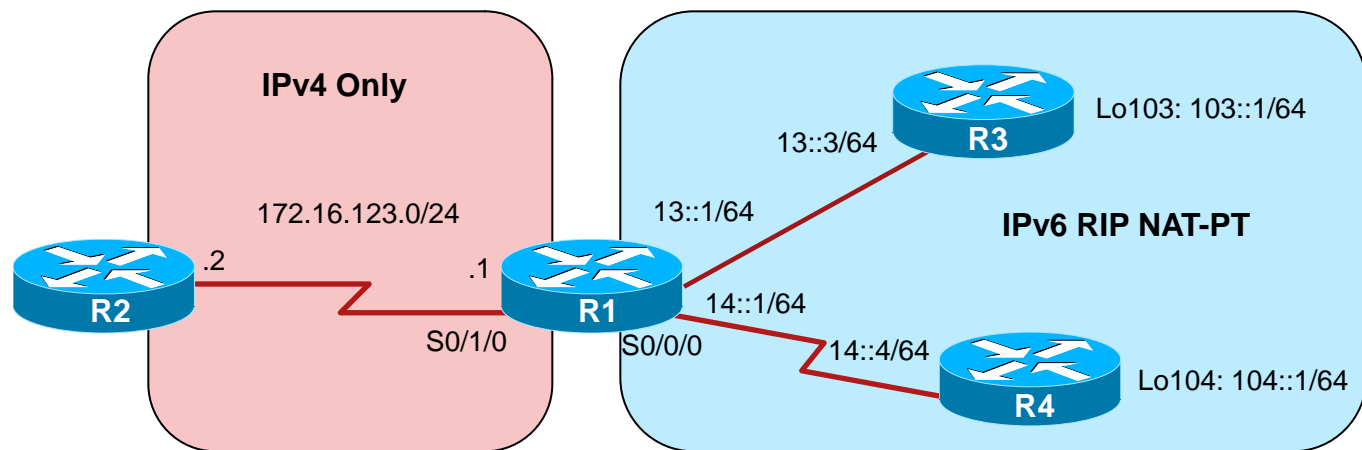
# Príklad pre statické NAT-PT (2)



```
R1(config)# ipv6 nat v6v4 source 14::4 172.16.123.100
R1(config)# ipv6 nat v4v6 source 172.16.123.2 1144::1
R1(config)#
R1(config)# ipv6 nat prefix 1144::/96
R1(config)#
```

- `ipv6 nat v6v4 source 14::4 172.16.123.100`
  - definuje ako bude vyzerat' ipv6 adresa routra R4 v IPv4 svete
- `ipv6 nat v4v6 source 172.16.123.2 1144::1`
  - definuje ako bude vyzerat' IPv4 adresa routra R2 v IPv6 svete

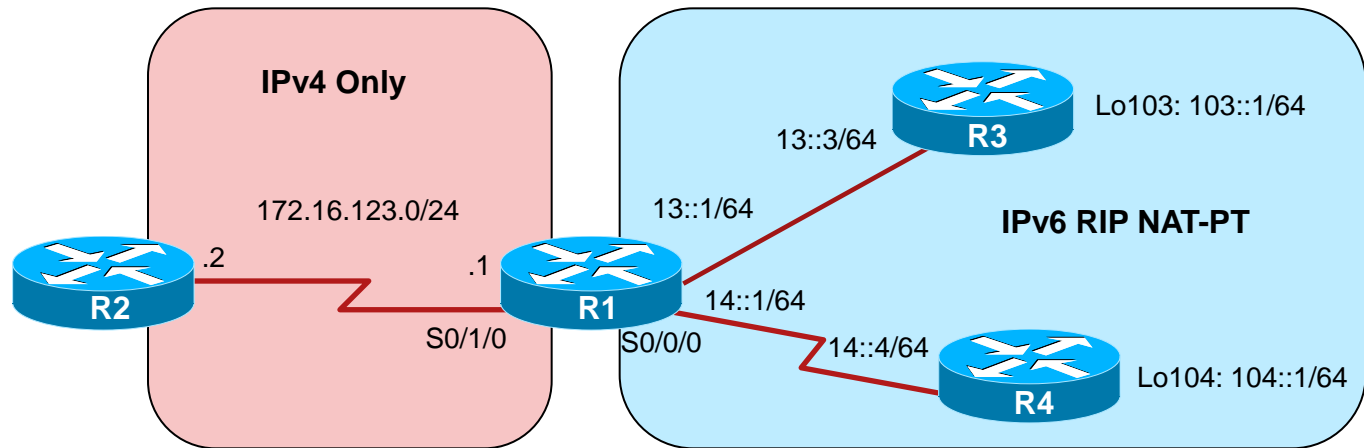
# Príklad pre statické NAT-PT (3)



```
R1# show ipv6 route connected
<output omitted>
C   13::64 [0/0]
    via FastEthernet0/0, directly connected
C   14::/64 [0/0]
    via Serial0/0/0, directly connected
C   1144::/96 [0/0]
    via NVI0, directly connected
R1#
R1# config t
R1(config)# ipv6 router rip NAT-PT
R1(config-rtr)# redistribute connected metric 3
R1(config-rtr)# exit
R1#
```

- NVI je NAT virtual interface, priamo pripojený
- Musíme redistribuovať (ohlásiť) do IPv6 siete

# Príklad pre statické NAT-PT (5)



```
R4# show ipv6 route rip
```

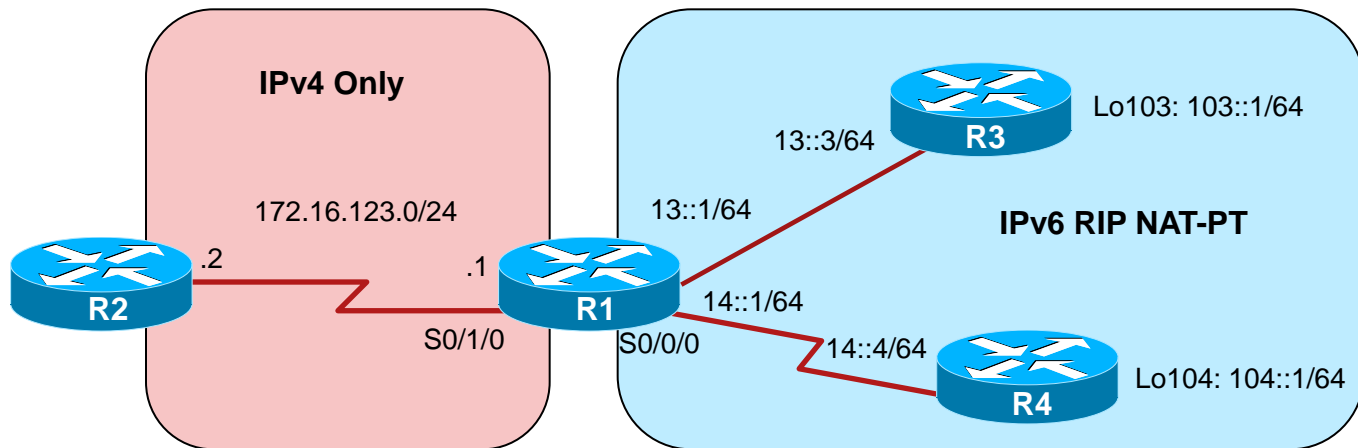
```
<output omitted>
```

```
R 13::/64 [120/2]  
   via FE80::1, Serial 1/1.7
```

```
R 1144::/96 [120/4]  
   via FE80::1, Serial 1/1.7
```

```
R4#
```

# Príklad pre statické NAT-PT (6)



```
R4# ping 1144::1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 1144::1, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 68/70/73 ms
R4#
```

```
R1# show ipv6 nat translations
Prot  IPv4 source          IPv6 source
     IPv4 destination  IPv6 destination
--    --                --
     172.16.123.2      1144::1
icmp  172.16.123.100,7364 14::4, 7364
     172.16.123.2, 7364 1144::1, 7364
--    172.16.123.100    14::4
     --                --
R1#
```

# Dynamické NAT-PT pre IPv6

- Pri dynamických NAT-PT sú adresy alokované z dynamického rozsahu
  - Ako pri dynamickom IPv4 NAT
  - Podobne aj syntax príkazov je ako pre IPv4 NAT
- Keď NAT-PT smerovač prijme paket s cieľovou IPv6 adresou, ktorá patri do NAT /96 prefix rozsahu (NAT-PT prefix)
  - Preloží IPv6 adresu za IPv4 adresu z preddefinovaného pool-u

# Konfigurácia dynamických NAT-PT - z v6 do v4

- Konfiguruj IPv4 rozsah na ktorý sa budú prekladať IPv6 adresy

```
Router(config)#
```

```
ipv6 nat v6v4 pool NAME START-IPV4 END-IPV4  
prefix-length PREFIX-LENGTH
```

```
R1(config)# ipv6 nat v6v4 pool POOL-12 172.16.12.100 172.16.12.101 prefix-length 24  
R1(config)# ipv6 nat v6v4 pool POOL-123 172.16.123.100 172.16.123.101 prefix-length 24  
R1(config)#
```

# Konfigurácia dynamických NAT-PT - z v6 do v4

- Vytvor IPv6 ACL a spoj ho s rozsahom
    - IPv6 ACL definuje zoznam rozsahov, ktoré majú povolené NAT
- Router(config)#

```
ipv6 nat v6v4 source {list {access-list-number / name}  
pool name}
```

```
R1(config)# ipv6 access-list LOOPBACK  
R1(config-ipv6-acl)# permit ipv6 104::/64 any  
R1(config-ipv6-acl)# permit ipv6 103::/64 any  
R1(config-ipv6-acl)# exit  
R1(config)# ipv6 access-list PHYSICAL  
R1(config-ipv6-acl)# permit ipv6 13::/64 any  
R1(config-ipv6-acl)# permit ipv6 14::/64 any  
R1(config-ipv6-acl)# exit  
R1(config)#  
R1(config)# ipv6 nat v6v4 source list LOOPBACK pool POOL-12  
R1(config)# ipv6 nat v6v4 source list PHYSICAL pool POOL-123  
R1(config)#
```



# Konfigurácia dynamických NAT-PT

## - z v4 do v6

- Definuj pool IPv6 adres na ktoré sa budú prekladať zdrojové IPv4 adresy

```
Router(config)#
```

```
ipv6 nat v4v6 pool name start-ipv6 end-ipv6 prefix-length prefix-length
```

```
R1(config)# ipv6 nat v4v6 pool POOL-1144 1144::1 1144::2 prefix-length 96  
R1(config)#
```

# Konfigurácia dynamických NAT-PT

## - z v4 do v6

- Zadefinuj NAT-PT prefix
- Vytvor IPv4 ACL a spoj ho s rozsahom
  - IPv4 ACL definuje zoznam povolených IPv4 adres

```
Router(config)#
```

```
ipv6 nat v4v6 source {list {access-list-number / name}  
pool name}
```

```
R1(config)# ipv6 nat prefix 1144::/96  
R1(config)# ip access-list standard IPV4  
R1(config-std-nacl)# permit 172.16.123.0 0.0.0.255  
R1(config-std-nacl)# permit 172.16.12.0 0.0.0.255  
R1(config-std-nacl)# exit  
R1(config)#  
R1(config)# ipv6 nat v4v6 source list IPV4 pool POOL-1144  
R1(config)#
```

# Porovnanie IPv4 a IPv6 (1)

IPv4	IPv6
Zdrojová a cieľová adresa majú veľkosť 32 bitov	Zdrojová a cieľová adresa majú veľkosť 128 bitov
Podpora IPsec je voliteľná	Podpora IPsec je vyžadovaná štandardom
Žiadna identifikácia tokov paketov používaná smerovačmi na zabezpečenie QoS nie je v hlavičke IPv4 prítomná	Identifikácia tokov paketov používaná smerovačmi na zabezpečenie QoS je v hlavičke IPv6 prítomná v poli Flow Label
Fragmentácia je zabezpečovaná smerovačmi aj vysielačmi	Fragmentácia je zabezpečovaná výhradne vysielačmi. Smerovače fragmentáciu nezabezpečujú
Hlavička obsahuje kontrolnú sumu na kontrolu integrity paketu	Hlavička kontrolnú sumu na kontrolu integrity paketu neobsahuje
Hlavička obsahuje voliteľné polia (options)	Všetky voliteľné polia sú presunuté do IPv6 rozširujúcich hlavičiek, ktoré nie sú súčasťou základnej hlavičky
Na rozpoznanie adresy linkovej vrstvy (MAC adresy) mapovanej na IPv4 adresu sa používa ARP (Address Resolution Protocol) ako broadcast	ARP funkcie na seba preberá ICMPv6 a využíva multicast

# Porovnanie IPv4 a IPv6 (2)

IPv4	IPv6
Na správu členstva skupín lokálnych sietí sa používa protokol IGMP (Internet Group Management Protocol)	Protokol IGMP je nahradený správami Multicast Listener Discovery (MLD)
Na určenie najlepšej predvolenej brány (default gateway) sa používa obvykle DHCP	Využívajú sa ICMPv6 Router Solicitation a Router Advertisement správy. Implementácia týchto správ je povinne vyžadovaná špecifikáciou
Na adresovanie všetkých uzlov na určitej sieti sa používajú broadcast adresy	V IPv6 neexistujú broadcast adresy. Namiesto toho existuje tzv. link-local scope all-nodes multicast adresa, ktorou môžu byť adresované všetky uzly jednej podsiete naraz
Konfigurácia prebieha manuálne alebo použitím DHCP	Nepotrebuje ani manuálnu konfiguráciu, ani DHCP. Protokol obsahuje autokonfiguráciu
Musí podporovať minimálne 576 bajtový paket (s možnosťou fragmentácie).	Musí podporovať minimálne 1280 bajtový paket (bez fragmentácie).

