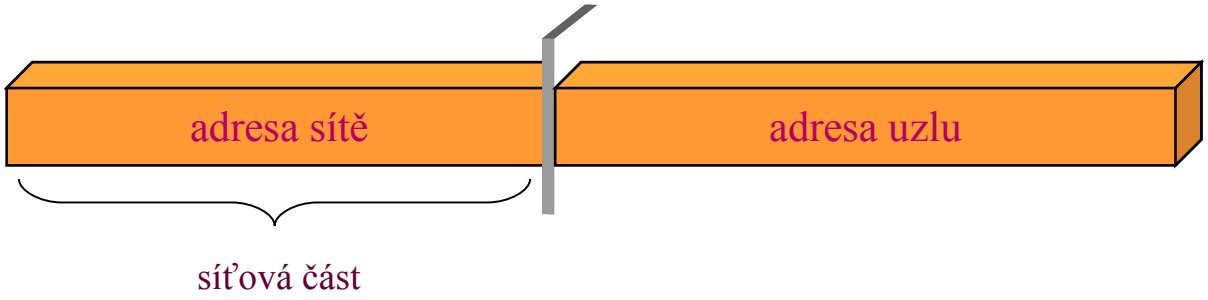
Protokol IP verze 6

- Motivace pro implementaci IPv6
- Charakteristika IPv6
- Formát paketu IPv6
- Aktuální stav penetrace IPv6

Adresní prostor

- V polovině 90. let → obrovský rozvoj Internetu
  - začalo se rýsovat několik problémů a nových vlastností
  - Nejvýznamnějším problémem v pol. 90. let
    - byl **krátící se adresní prostor**
  - stav v IP verze 4
    - IPv4 ⇨ teoreticky 4 miliardy adres
    - ve třech třídách

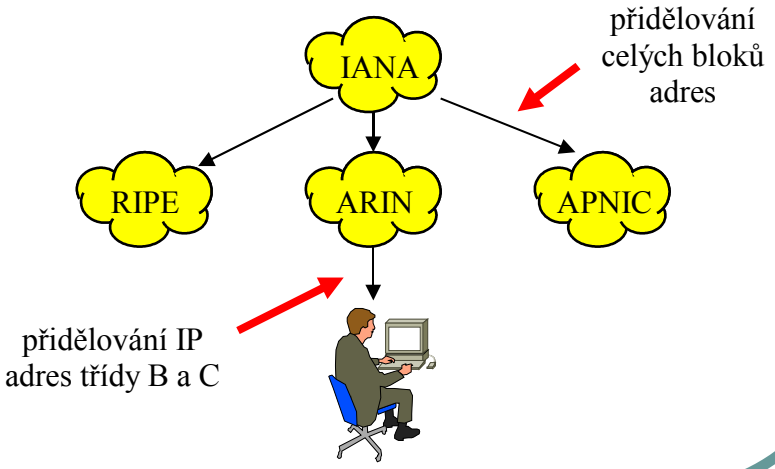


- Ve skutečnosti mnohem méně
  - souvisí se způsobem distribuce IP adres

Distribuce IP adres

Internet Assigned Numbers Authority

- Zásada :
  - žádná IP adresa nesmí být přidělena dvakrát
    - dnes již existují výjimky
- Řešení:
  - bude existovat centrální autorita, která je bude přidělovat
  - původně bylo touto autoritou středisko SRI NIC (při Univ. of Stanford v USA)
    - každý zájemce z celého světa žádal přímo SRI NIC, ta přidělovala adresy přímo
    - časem se to stalo organizačně neúnosné
- Další vývojové stádium:
  - centrální autoritou se stala organizace IANA
  - IANA přidělovala celé bloky IP adres regionálním "přidělovatelům"
    - RIPE (Evropa)
    - APNIC (Asie a Pacific)
    - ARIN, v USA (Internic do r. 98)



Problém s (původními) IP adresami

Internet Architecture Board

- Úbytek IP adres byl velký
  - původně se nepočítalo s tak velkým zájmem
  - přidělování po celých třídách (A, B a C) bylo ve většině případů plýtváním
    - řešilo se přidělováním více "menších" adres, například místo 1 síťové adresy B se přidělilo více adres C
    - adresy třídy A se prakticky přestaly přidělovat

problémem byla malá "granularita" tříd IP adres
(nebylo možné se jemněji přizpůsobit skutečné velikosti sítě)

- IAB začala zvonit na poplach
  - založila v IETF[*1] celou oblast skupin věnovanou řešení tohoto problému
  - vypsala se výzva k předkládání řešení
  - začalo se měřit, jak dlouho adresy ještě vydrží …

začalo hrozit vyčerpání 32-bitového prostoru všech IP adres !!!!

[*1] IETF Internet Engineering Task Force komise pro „techniky“ Internetu

Doba kamenná

- V dobách „třídní adresace“ - značně neefektivní rozdělování
  - dodnes pozůstatek z doby „divokého Internetu“
    - 3.0.0.0/8 *General Electric Company*
    - 9.0.0.0/8 *IBM*
    - 12.0.0.0/8 *AT&T Bell Laboratories*
    - 13.0.0.0/8 *Xerox Corporation*
    - 15.0.0.0/8 *Hewlett-Packard Company*
    - 16.0.0.0/8 *Digital Equipment Corporation*
    - 17.0.0.0/8 *Apple Computer Inc.*
    - 19.0.0.0/8 *Ford Motor Company*
    - 32.0.0.0/8 *AT&T Global Network Services*
    - ....
  - takže dnes máme „několik bloků adres RIPE, mezi nimi Ford....

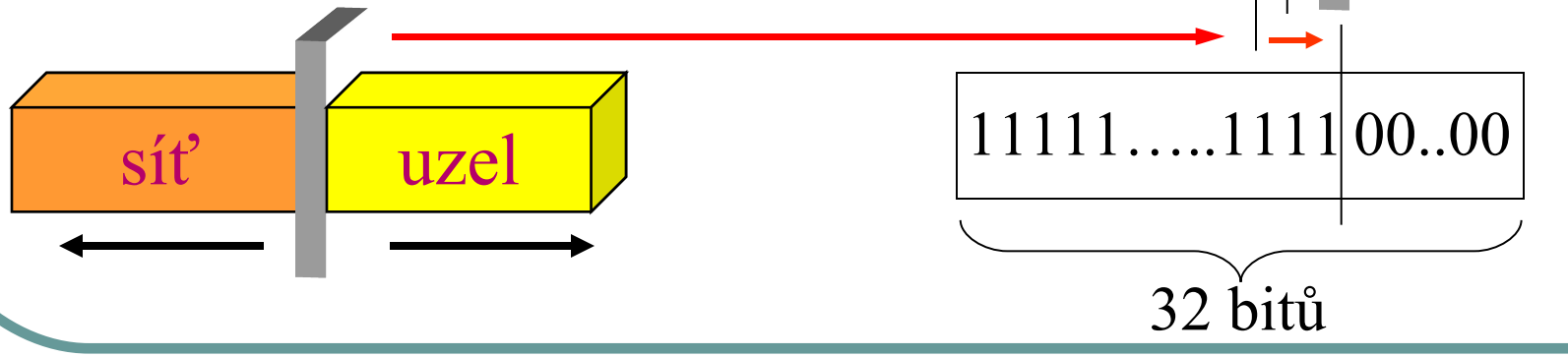
http://www.caida.org/research/id-consumption/whois-map/images/whois-20071001.png

Jaké je možné řešení?

- Princip řešení:
  - malou "granularitu" tříd IP adres by bylo možné řešit posunem hranice (bitové pozice) mezi síťovou částí a relativní adresou uzlu
- Problém:
  - původní mechanismy práce s IP adresami na to obecně nebyly připraveny
    - některé ano, ale nešlo se na to spoléhat

nutnost použití masky:

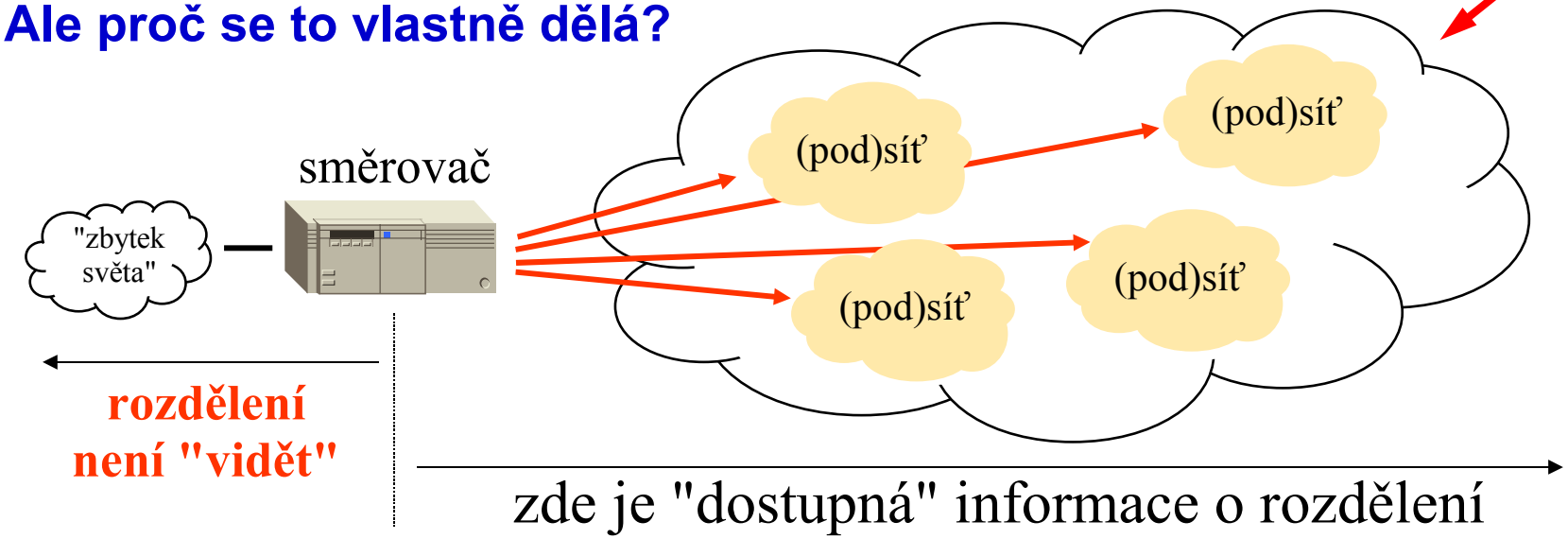
- u tříd je hranice (bitová pozice) určena nejvyššími bity
- jemnější nastavení hranice musí být určeno jiným způsobem – pomocí tzv. **masky**



Princip podsítí

- Idea dělení na podsítě (subnettingu):
  - hranice (bitová pozice) se posune směrem k nižším bitům
    - tj. adresy uzlů se rozdělí na **několik skupin**
      - velikosti mocniny 2, aby to byl posun o celé bitové pozice
    - použijí se masky
    - **vše se udělá někde "izolovaně" (v rámci jedné soustavy dílčích sítí)**
      - **a informace o tomto rozdělení není šířena "do světa„**

- **Ale proč se to vlastně dělá?**



Smysl dělení na podsítě

- Jde o možnost využít 1 síťovou adresu (třídy A, B či C) pro více sítí
  - jinak by to musely být samostatné síťové adresy
  - příklad:
    - díky subnettingu 4 malé sítě po 20 uzlech vystačí dohromady s 1xC (256 individuálních adres)
    - bez subnettingu by spotřebovaly 4xC (4x256, tj. 1024 individuálních IP adres)
- Lze ale využít jen tam, kde soustava sítí má jeden vstupní bod
  - neboť informace o rozdělení (pomocí masky) není šířena "do světa"
    - a kdyby bylo více vstupních bodů, nevědělo by se který z nich vybrat
- Není to problémem tam, kde má soustava sítí stromovitou strukturu
  - subnetting lze použít v podstromu

Problém s IP adresami II.

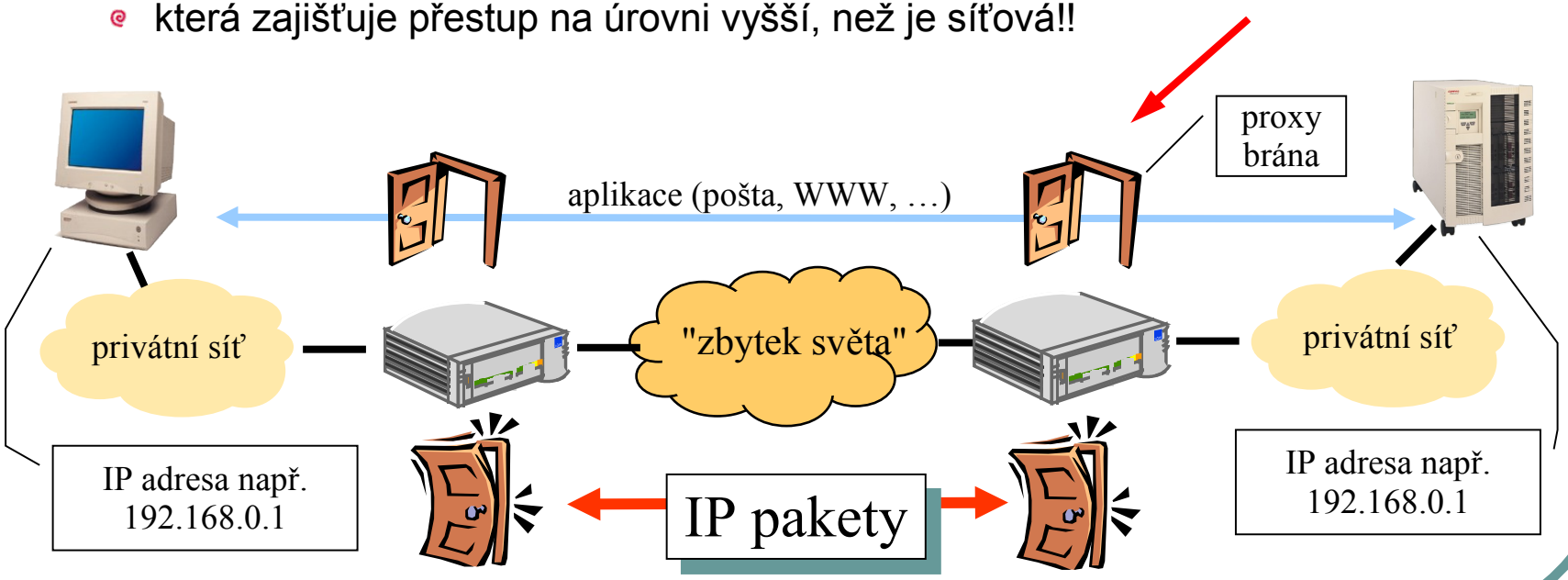
- Subnetting hodně pomohl
  - byl okamžitým řešením, které šlo použít "lokálně"
  - zpomalil úbytek IP adres, ale neřešil jej z principu
- Principiální řešení:
  - nové (větší) IP adresy
    - ale "klasické" 32 bitové adresy jsou natolik zakořeněné hlavně v protokolu IP, že je nutné udělat **novou verzi tohoto protokolu**

    - začalo se pracovat **na protokolu příští generace**
- Vedle subnettingu se prosadila i další "dočasná" řešení
  - která neřeší podstatu problému, ale zmírňují jeho dopady
  - **neveřejné (privátní) IP adresy**
    - umožňují vícenásobné použití IP adres
  - **CIDR (supersítě)**
    - Classless InterDomain Routing
  - **NAT**
    - Network Address Translation

podařilo se výrazně zpomalit úbytek IP adres, naléhavost principiálního řešení klesla

Neveřejné IP adresy

- Co brání vícenásobnému použití IP adres?
  - to, že by směrovací algoritmy nevěděly, kam doručovat IP pakety
- Idea: tam, kde nebude existovat přímá komunikace (nutnost směrovat) by se adresy mohly opakovat
  - tato situace nastává v sítích bez přímé IP konektivity ("**privátních sítích**"), které jsou odděleny od "ostatního světa" vhodnou bránou (firewallem)
    - která zajišťuje přestup na úrovni vyšší, než je síťová!!



Privátní IP adresy (RFC1918, dříve RFC1597)

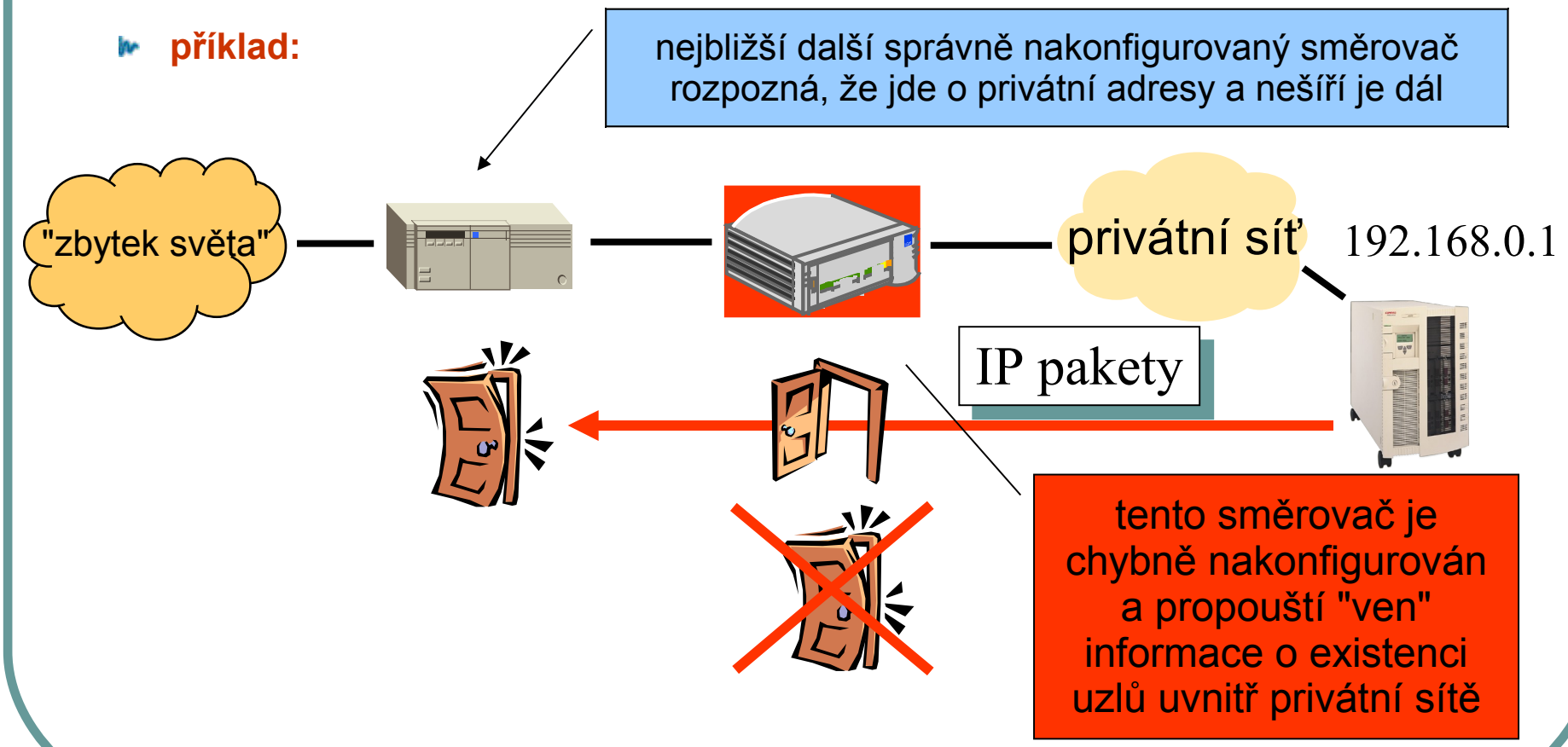
- Podmínka fungování:
  - na hranicích privátních sítí je třeba zastavit šíření směrovacích informací
    - "ohlašujících" existenci uzlů uvnitř privátních sítí
- Důsledek:
  - v privátních sítích lze použít v zásadě libovolné IP adresy
    - uvnitř jedné privátní sítě musí být jednoznačně
    - v různých privátních sítích mohou být použity stejné IP adresy

- Doporučení:
  - nepoužívat úplně libovolné IP adresy, ale takové, které byly k tomuto účelu vyhrazeny (RFC 1918)
  - jsou to adresy:
    - 1síťová adresa třídy A:
      10.0.0.0 – 10.255.255.255
    - 16 adres třídy B:
      - 172.16.0.0 – 172.31.255.255
    - 256 adres třídy C
      - 192.168.0.0 – 192.168.255.255

je vhodné používat i tam, kde síť není (nechce, nebude) připojena k Internetu

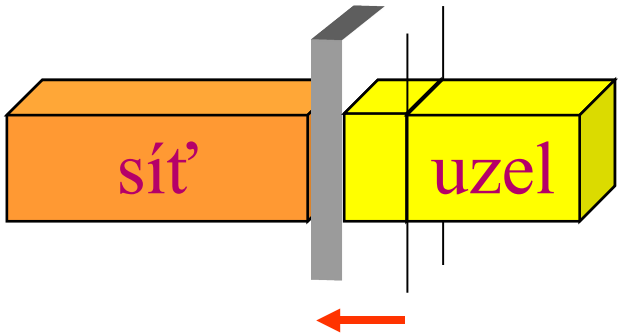
Privátní IP adresy

- Proč je vhodné používat v privátních sítích vyhrazené ("privátní") IP adresy, a ne libovolné IP adresy?
  - **příklad:**



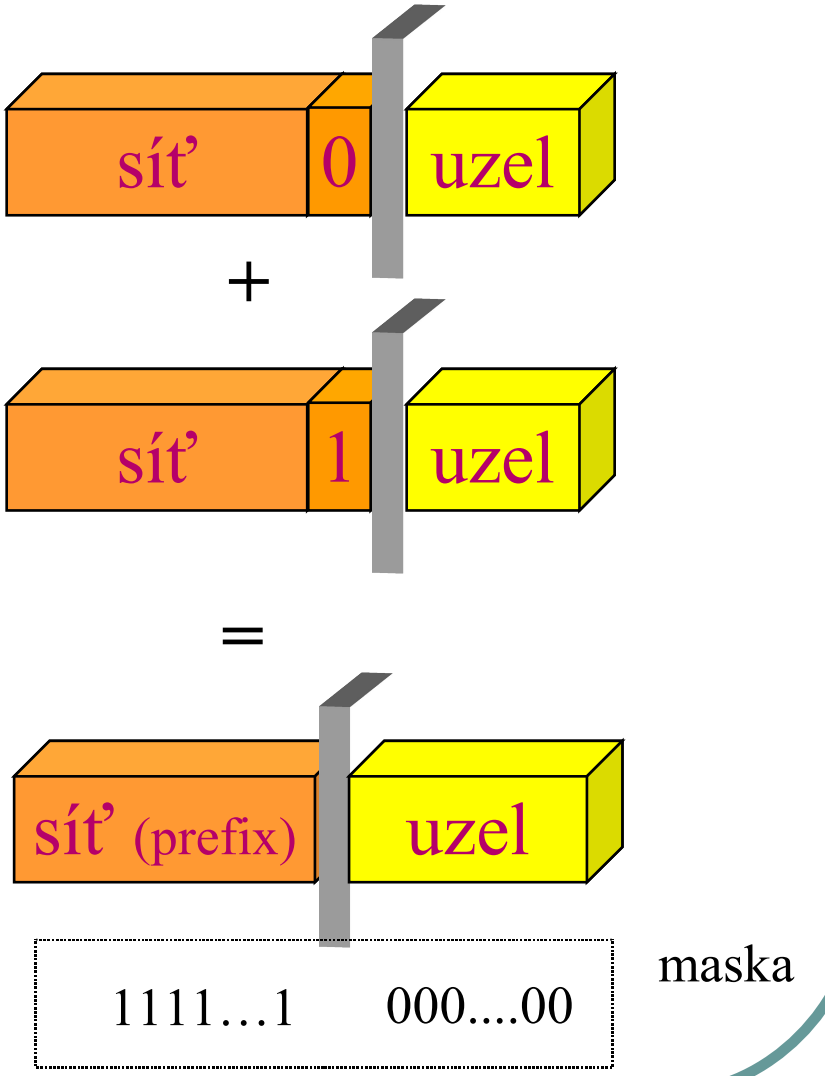
Mechanismus CIDR - supersítě
Classless InterDomain Routing (RFC 1517 - 1520)

- Řeší problém úbytku IP adres
  - umožňuje přidělovat koncovým sítím "přesně velké" skupiny IP adres
    - v zásadě to nahrazuje původní systém tříd A, B a C
- Řeší problém nárůstu směrovacích tabulek
  - dosud platilo: co 1 síťová adresa třídy A, B nebo C, to jedna položka ve směrovací tabulce
    - směrovací tabulka se prohledává při každém rozhodnutí o volbě směru
  - adresy se přidělují hierarchicky – agregace směrování – méně záznamů ve ST
  - TŘÍDY – byly v podstatě zrušeny
- Princip mechanismu CIDR
  - je v zásadě inverzní k subnettingu
    - také se tomu říká **supernetting**
  - předpokládá posun hranice (bitové pozice) mezi síťovou částí a adresou uzlu směrem "doleva" (k vyšším bitům)



Princip supersítí

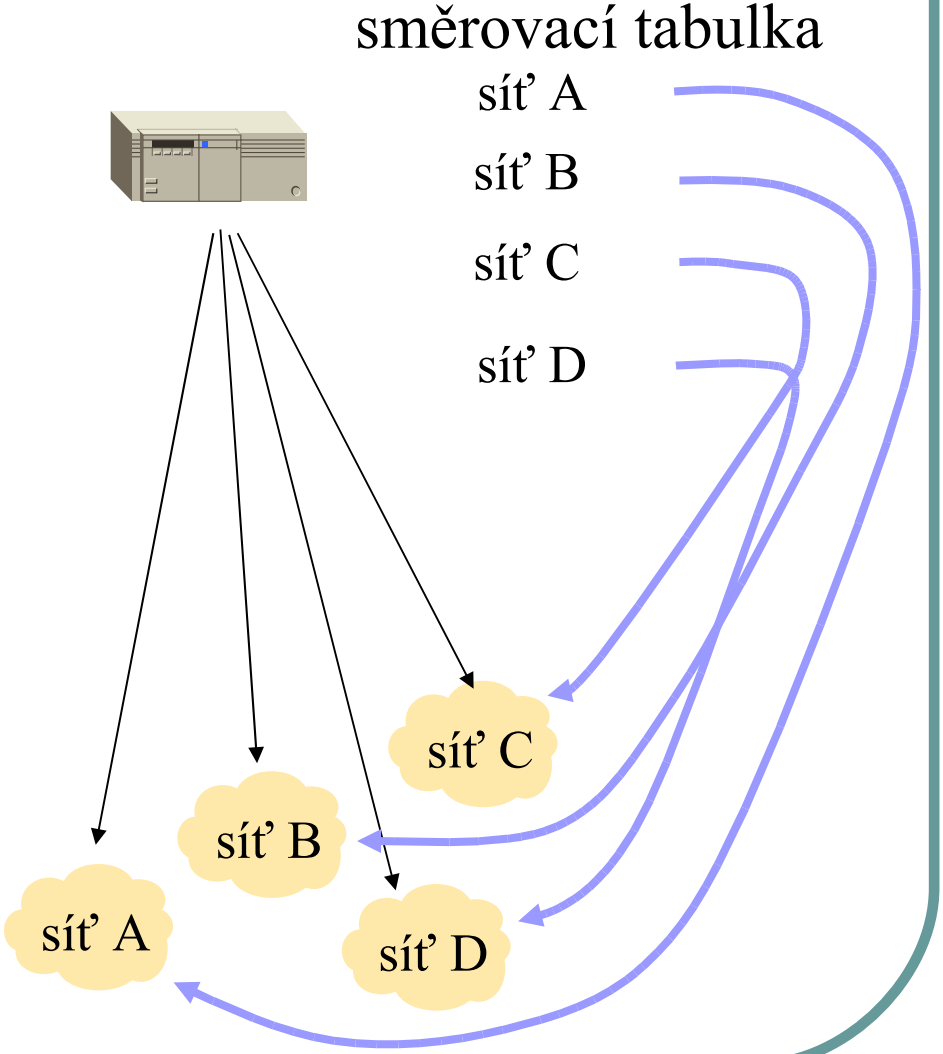
- Dochází k tzv. agregaci
  - slučování "sousedních" síťových IP adres
  - vzniká 1 výsledná "agregovaná" adresa (adresa supernet-u)
- Síťová část je nyní označována jako "prefix"
  - a jeho velikost je vyjadřována v počtu bitů (síťové části)
- **Adresy jsou dnes přidělovány zásadně jako tzv. CIDR bloky**
  - např. 194.213.228/24 je CIDR blok odpovídající 1 dřívější síťové adrese C (má 24 bitů prefixu, zbývá 8 na adresu uzlu)



Problém směrovacích tabulek

- Dříve platilo:
  - přidělovaly se celé síťové adresy, a to systémem "kdo první přišel …"
    - nebyl v tom žádný systém, kromě distribuce mezi regionální přidělovatele
  - pro každou síťovou adresu (A, B nebo C) musela být ve směrovacích tabulkách samostatná položka
  - směrovačům v páteřních částech Internetu začaly přetékat směrovací tabulky
    - RAM, CPU, latency...

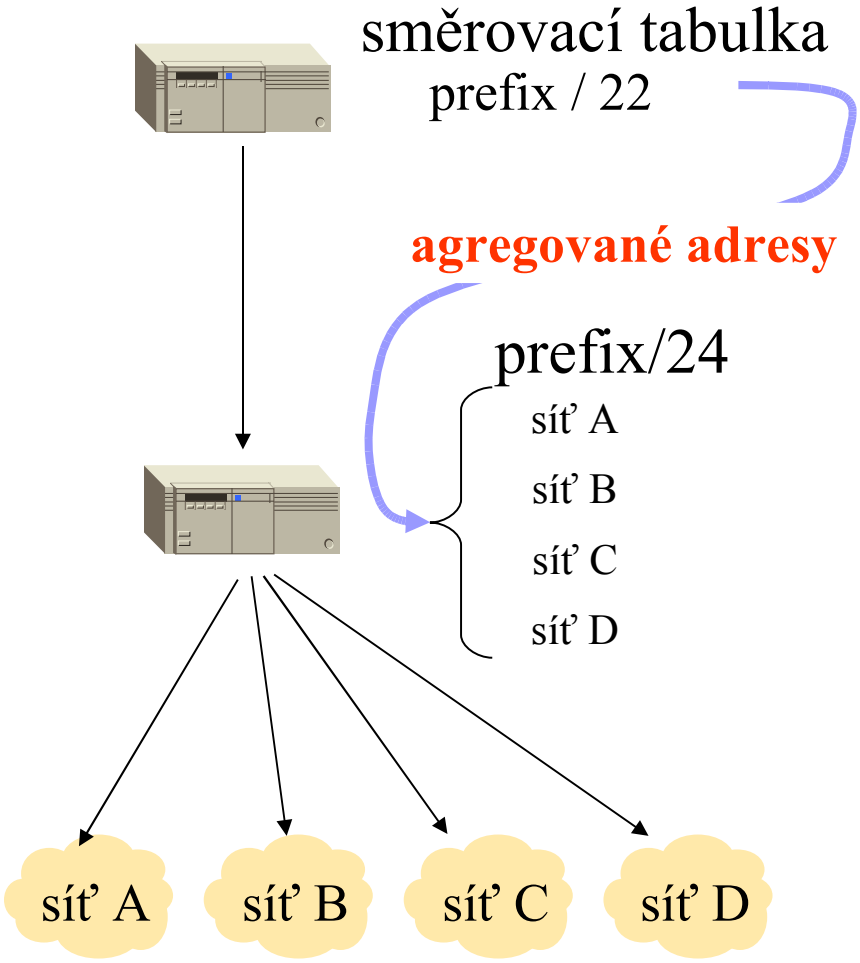
IP adresy byly nezávislé na způsobu připojení !!



Agregace směrovacích informací

- CIDR bloky umožňují agregovat (slučovat) i směrovací informace
  - jakoby: slučovat dohromady i položky směrovacích tabulek
  - detailní směrovací informace nemusejí být zbytečně šířeny "do světa"
    - mohou zůstat lokalizovány tam, kde jsou zapotřebí, kde vznikají a kde se mění

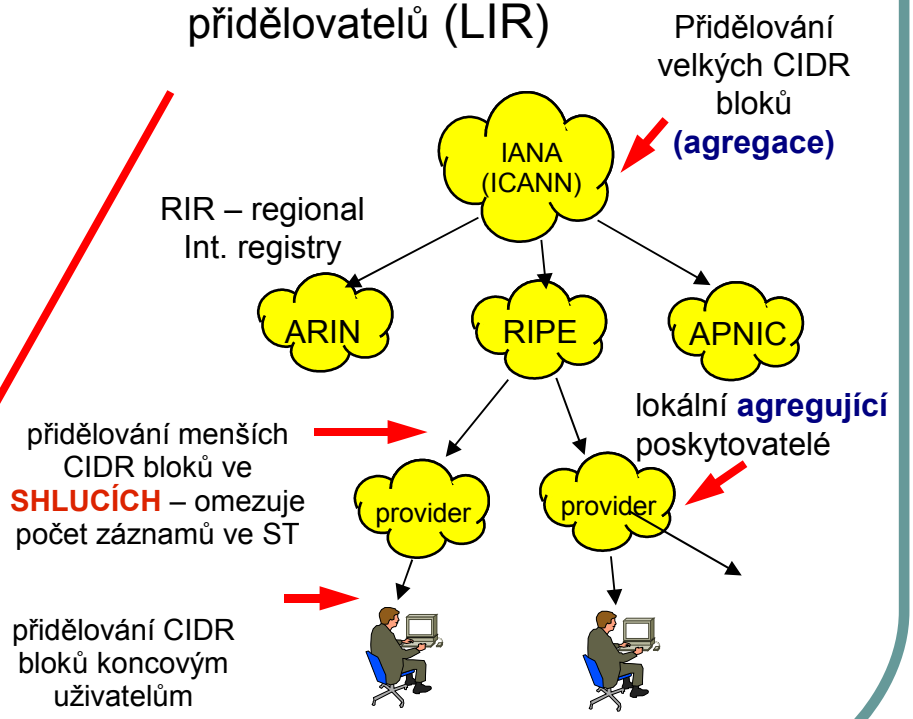
pozor:
IP adresy se stávají závislými na způsobu připojení !!!!



Důsledky mechanismu CIDR

- Šetří se IP adresami
  - CIDR bloky ⇨ byl dále zpomalen úbytek adres
    - ale příčina problému nebyla odstraněna
- Šetří se směrovací tabulky
  - umožnilo to redukovat jejich objem, tím zrychlit směrování
    - ale **nepostačuje**, tabulky jsou již tak neúnosně velké
    - jsou nutná ještě jiná řešení, např. autonomní systémy (zavádí další stupeň agregace směrovacích informací)

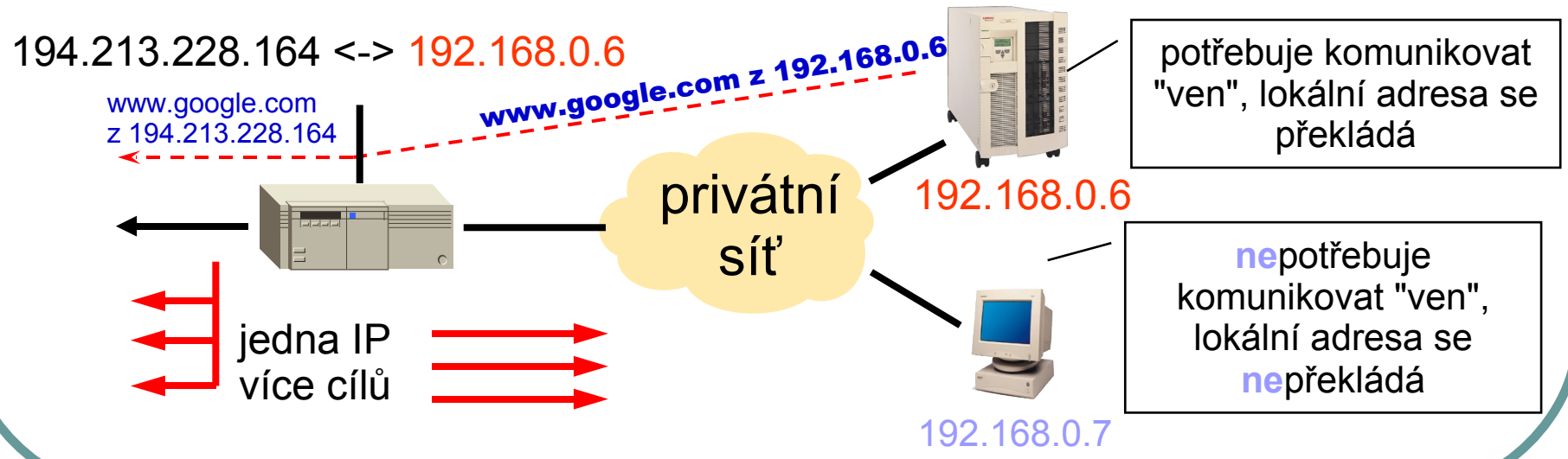
- Musel se změnit způsob distribuce IP adres
  - "přidělovatelem" nyní musí být i jednotliví provideři
    - musí se registrovat u regionálních přidělovatelů (LIR)



při změně providera musí uživatelé změnit IP adresy svých uzlů !!!

NAT – Network Address Translation (RFC 1631)

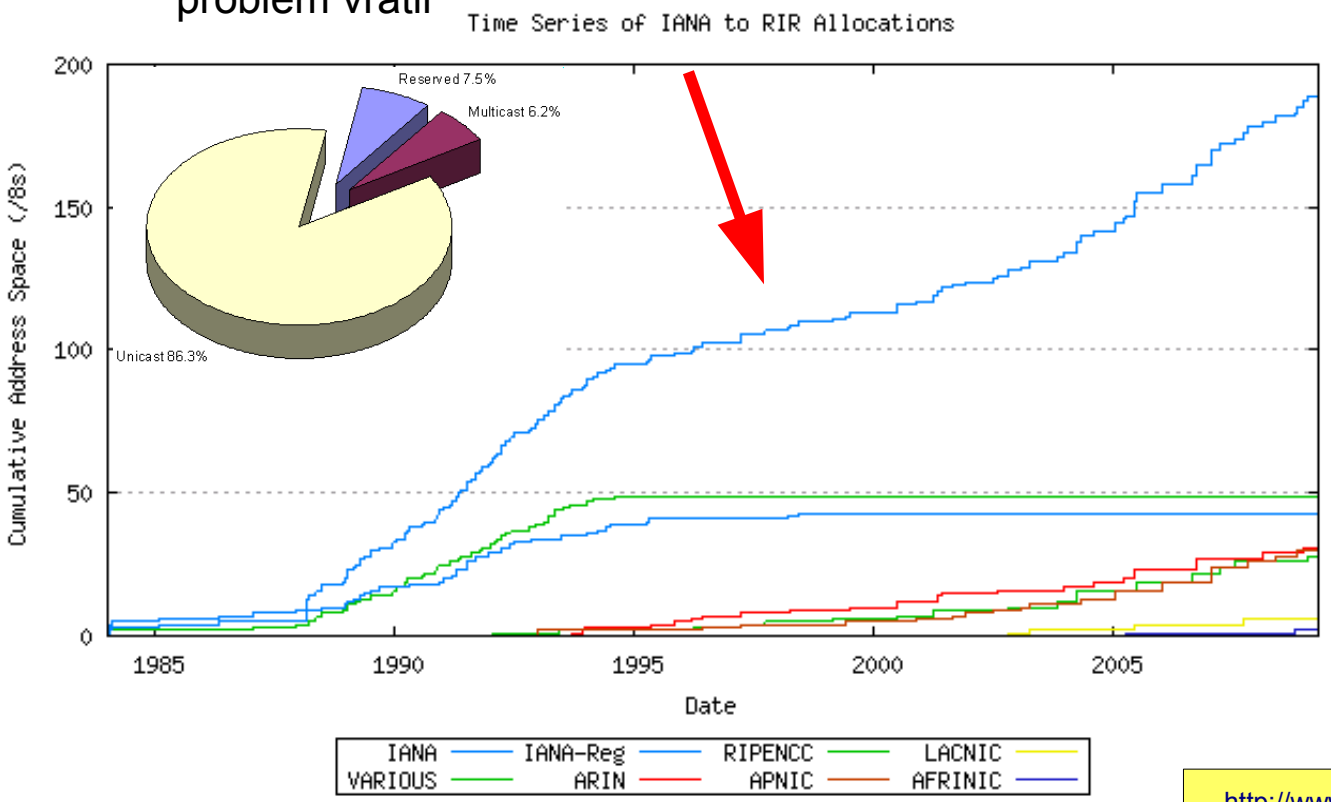
- NAT - překládá (mění "za chodu") IP adresy
  - používá se na rozhraní mezi privátní sítí a veřejným Internetem
    - překládá lokální (privátní, vícenásobně použitelné) adresy na veřejné (unikátní) adresy
  - poskytuje zabezpečení
    - lokální adresy "nejsou vidět"
  - šetří IP adresy
    - pokud jen část lokálních uzlů potřebuje komunikovat s vnějším světem !!!!



Na chvíli klid

- Pomocí všech výše popsaných metod se podařilo problém vyčerpání adresního prostoru minimalizovat
  - ale pouze **dočasně**
    - když se objevila **mobilní zařízení**, začala se připojovat **Čína** a další státy, tak se problém vrátil

IPv6 ztratilo hlavní hnací sílu
to silně pozdrželo nástup IPng

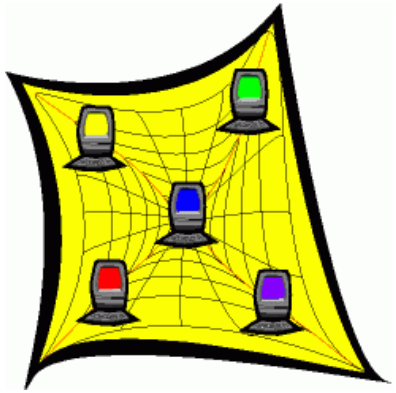


Aktuální předpověď civilizace se zhroutí: † 20.8.2011

http://www.potaroo.net/tools/ipv4/index.html

Další problémy

- Navíc zvolená opatření silně porušují základní principy Internetu:
  - možnost přímé komunikace
  - zmenšující se adresní prostor ale nebyl jediným problémem...
- S rozvojem Internetu se objevily nové požadavky na přenosové služby
  - IP verze 4 neřeší (rozhodně ne elegantně) například tyto problémy:
    - již zmíněný nedostatek dres
    - nedostatečná podpora služeb se zaručenou kvalitou (QoS)
    - design neodpovídající vysokorychlostním sítím
    - bezpečnostní mechanismy nejsou obsaženy přímo v IP
    - nedostatečná podpora mobilních zařízení
    - neexistující automatická konfigurace
    - ...
  - stále tedy intenzivně pokračoval vývoj nového protokolu



Geneze

- Původní protokol IP verze 4 (IPv4)
  - byl specifikován v r. 1980/81
- Specifikace nové verze IP se objevila:
  - až v roce 1995 (po 15 letech) ⇨ je vidět, že původní IP byl navržen velmi dobře
- Nový protokol je označován:
  - nejprve jako protokol příští generace - IP next generation (IPng)
  - později se vžilo označení IPv6 (IP verze 5 - exp. proudový protokol)
  - nový protokol byl vyvíjen s cílem postupně nahradit protokol IPv4
    - při násilném vnucení by byl uživateli odmítnut
  - podmínkou nového protokolu tedy byl co nejsnazší přechod na novou verzi

Geneze

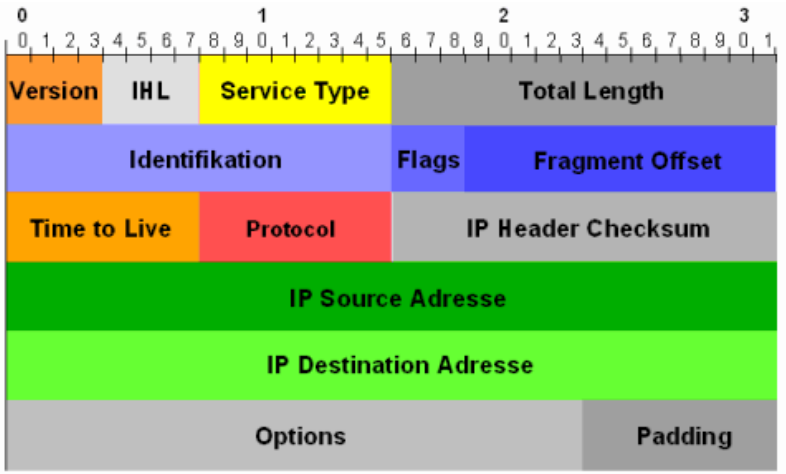
- Vývoj na papíře předběhl reálné možnosti
  - téměř se opakoval katastrofální „model“ vývoje ISO
- Základní dokument IPv6: RFC1883 z r. 1995, (revize až 2006)
  - autoři měli velké plány
  - ale v prvních letech reálná implementace značně pokulhávala
    - mnoho výrobců implementovalo jenom okleštěnou část IPv6
      - marketing – my to máme, *our device is IPv6 compatible*
  - spousta věcí se dlouho vůbec neřešila
    - například deklarovaná podpora mobility, autentizace, bezpečnostních vlastností... prostě nebyla

→ **frustrace uživatelů**

- Nový „boom“ - rozbuška v Asii – Asie zaspala IPv4
  - mezi lety 2004 – 2007 rozsáhlý „*update*“
    - přepracována podpora deklarovaných nových vlastností
    - postupná penetrace do zařízení a OS

Struktura paketu protokolu IPv6

- Struktura hlavičky byla volena s ohledem na:
  - zvětšení adresního prostoru
  - optimalizaci průchodu paketů směrovači

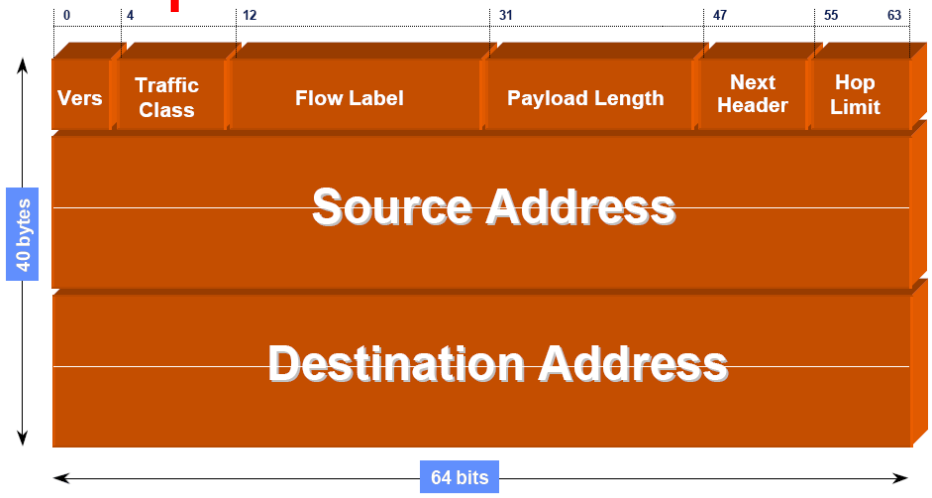


- Původní hlavička protokolu IPv4
  - obsahovala značné množství informací
    - některé z těchto informací se používají jen málo
    - jiné se při průchodu směrovači nemění
- Základní myšlenkou IPv6 je přesunutí značné části těchto inf. do volitelné části
  - v hlavičce zůstaly pouze nejdůležitější údaje
- Jiné údaje byly zcela vypuštěny
  - např. kontrolní součet!

Struktura hlavičky IPv6

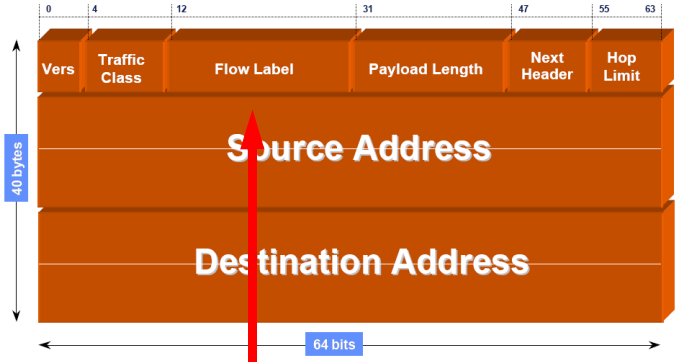
40B se může zdát hodně (u IPv4 je 40B IP+TCP) ale jen 32 B tvoří adresy

- Struktura hlavičky se skládá ze 40B záhlaví následovaného rozšířeními
  - pole Verze (4b) obsahuje 6 (u IPv4 4)
  - pole třída dat specifikuje naléhavost dat
    - jinak řečeno, která data budou zahazována v případě zahlcení sítě



- 0 - nespecifikovaná data
- 1 - provoz na pozadí (např. news)
- 2 - automatický provoz (např. mail)
- 4 - uživatelské velké přenosy (ftp. ...)
- 6 - interaktivní přenos (VNC, telnet...)
- 7 - management sítě (RIP, SNMP...)
- 8 - 15 přenosy v reálném čase
  - multimediální data
  - realtime řízení technolog. procesů
  - data s vyšším číslem (≥8) mají vyšší prioritu

Struktura hlavičky IPv6

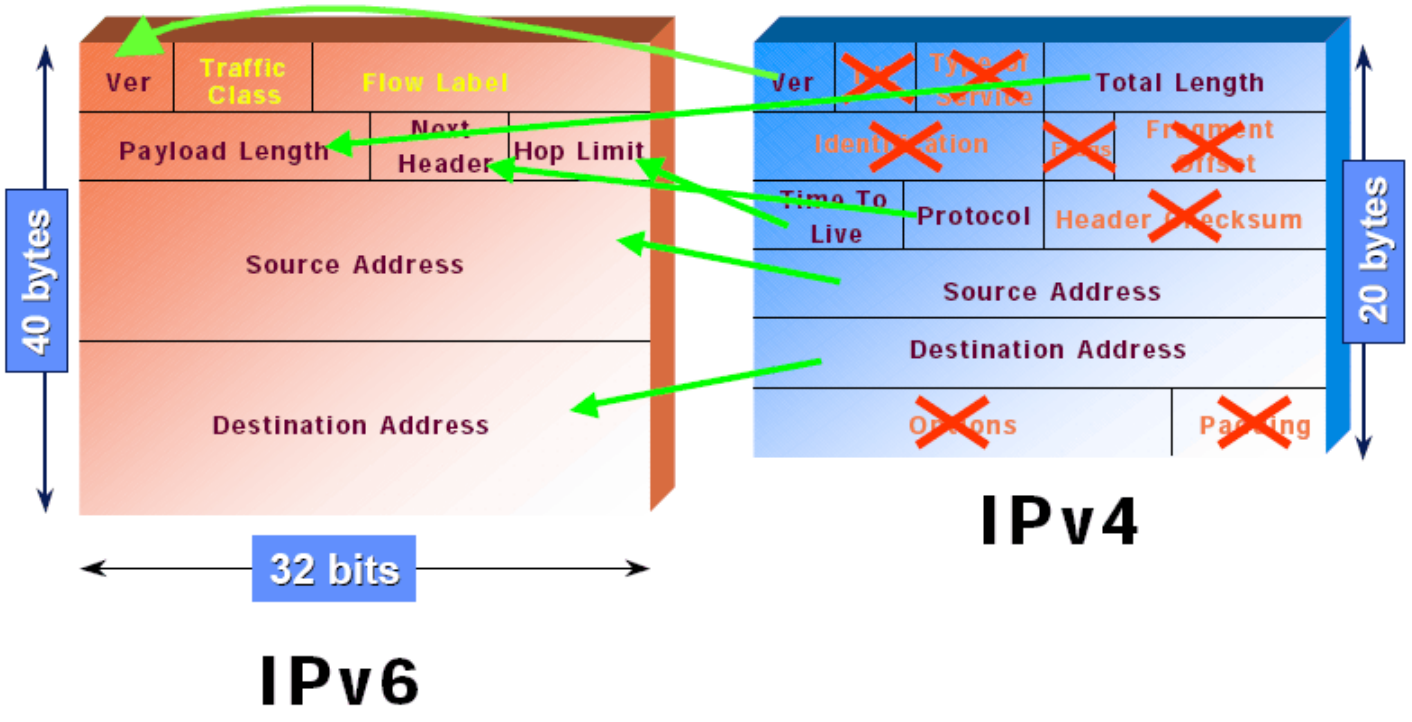


- Další položky tvoří:
  - délka dat (2B = 65535B), bez základní hlavičky
    - s použitím příznaku „ohromný datagram“ v další hlavičce i více
  - typ další hlavičky
    - TCP, UDP, IPv4, rozšíření hlavičky IPv6
  - identifikace toku dat
    - slouží ke dvěma účelům
      - **snížení zátěže směrovačů**
    - datagramy jednoho toku dostanou shodný identifikátor
    - směrovače pak řeší úlohu směrování pouze pro první datagram
    - další datagramy odesílá stále do stejného rozhraní (max. 6s)
      - **další možností je zajištění QoS**
    - směrovače se nakonfigurují tak, aby pro pakety s určitým FL upřednostňovaly jejich směrování
    - směrovače pak neobsluhují datagramy jako sekvenční frontu ale vybírají pakety s vhodným FL

nová myšlenka

Porovnání hlavičky IPv4 a IPv6

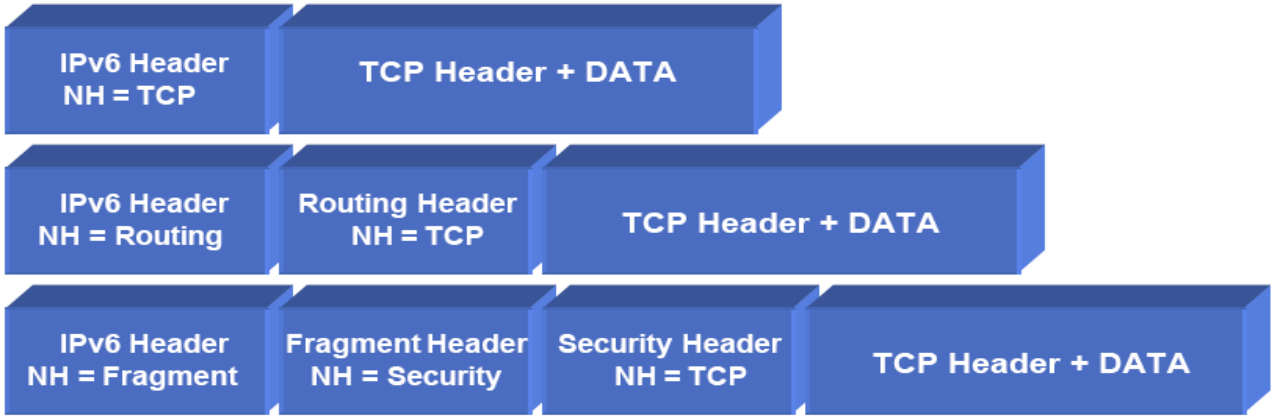
- V hlavičce IPv6 zůstaly pouze nejdůležitější informace
  - zejména takové, které se uplatňují při průchodu paketu směrovači



- i přes přesun značné části dat do volitelných položek má IP hlavička:
  - **40B** (z toho 32B zabírají jen adresy = 80%)

Rozšíření hlaviček (Next Headers)

- Na rozdíl od IPv4 důsledná koncepce
  - *„Méně významné“* informace jsou **přesunuty** do volitelných záhlaví



- Pole „Next Header“
  - ukazuje jaký typ hlavičky následuje (TCP, UDP, IPv4 nebo další IPv6)
    - v další hlavičce je za polem *Next Header* pole specifikující posunutí k další hlavičce
    - základní hlavička toto pole nemá, má vždy 40B
  - v dodatečných hlavičkách IPv6 se vyskytují méně často používané údaje

Dodatečné hlavičky IPv6

Např. využití u *RSVP* prot. rezervace kapacit po cestě

- *Volby pro všechny*
  - informace zajímavé pro každého po cestě (např. **upozornění pro směrovače**, že paket nese data, která by jej mohla zajímat)
- *Explicitní směrování*
  - datagram musí projít předepsanou cestou

Patrně stejně „nestravitelné“ jako u IPv4

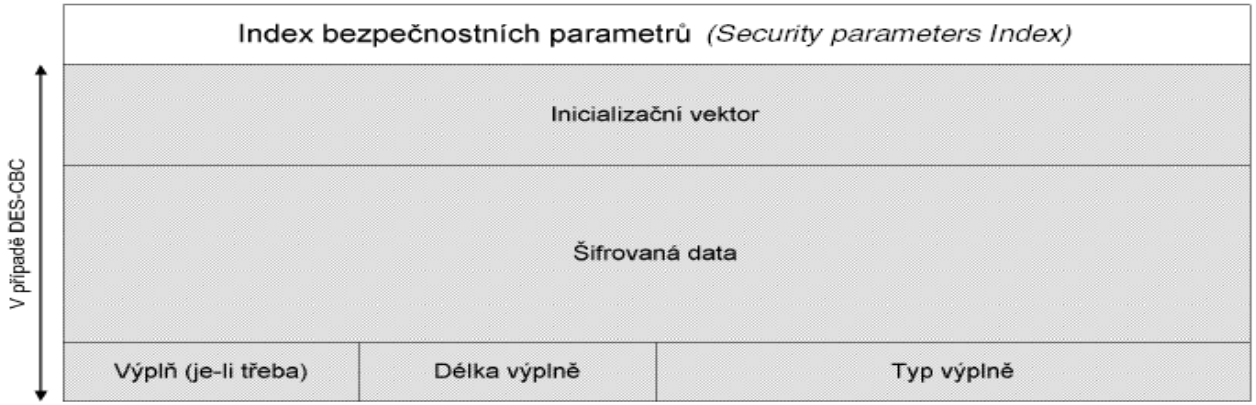
- *Fragmentace*
  - při fragmentaci paketu nese informace nutné pro jeho složení do původní podoby
- *Šifrování obsahu (ESP)*
  - obsah datagramu je zašifrován, ESP hlavička nese odkaz na parametry pro dešifrování
- *Autentizace (AH)*
  - data pro ověření totožnosti odesilatele a původnosti obsahu
- *Volby pro cíl*
  - informace určené příjemci datagramu (např. domácí adresa mobilního uzlu)
- *Mobilita*
  - hlavička pro potřeby komunikace s mobilními zařízeními
  - v podstatě explicitní směrování →pevná IP (domácí) + mobilní IP (přesměrování)

Fragmentace paketů

- U IPv4 běžná věc
  - perfektně zvládnutá technologie
- Negativum
  - minimum paketů prochází přes směrovače fragmentovaných
    - v hlavičce zbytečně položky týkající se fragmentace
  - specifikace IPv6 doporučuj → LV alespoň 1280B
    - Eth 1500B...
  - fragmentace bude u IPv6 ještě řidší jev než u IPv4
- Celá problematika fragmentace vyčleněna do samostatné hlavičky
  - navíc fragmentaci u IPv6 vždy **pouze odesilatel**

Šifrovací a autentizační hlavička

- IPv6 nativně podporuje autentizaci a šifrování
  - autentizace je zajišťována vypočítáním CRC za pomocí MD-5 a šifrovacího 128bit. klíče → ten musí mít odesilatel i příjemce



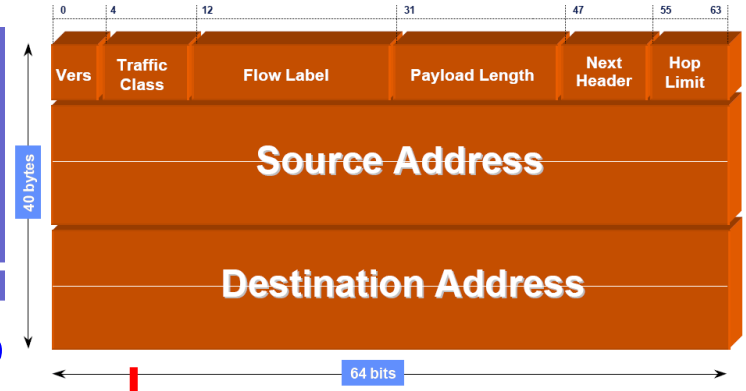
pole IBP je ukazatel (index) do tabulky více předem dohodnutých klíčů

  - bezpečnostní hlavička umožňuje data i šifrovat
    - jedná se o poslední hlavičku – všechna následující data jsou šifrována

šifrovat mohou:

1. odesilatel a příjemce
2. mezilehlé směrovače

Formát adresy



- Pro adresy zdroje a cíle je v IPv6 vyhrazeno
  - 2 x 128b (2 x 16B) = 3.4 x 10^{38}*

Frustrace z „adresního kolapsu“ musela být opravdu značná

- Rozeznáváme tři typy adres
  - *Unicast* - jednoznačná adresa síťového rozhraní
  - *Anycast* - adresa skupiny síťových rozhraní
    - adresována je skupina uzlů, ale paket je doručen pouze jednomu (nejbližšímu z hlediska topologie)
    - typicky: hledám nejbližší přístupový bod
  - *Multicast* – oběžník
    - adresována je skupina uzlů
    - paket je doručen všem

- datagramy typu všeobecného oběžníku (*Broadcat*) v IPv6 **neexistují**

*** - někdo si dal práci a spočítal že na každý mm^2 povrchu Země připadá $667*10^{38}$ adres**
- na každého člověka dnes připadá prostor $1.5*10^{19}$ dnešních internetů

Zápis adresy

na to si člověk jen tak nezvykne ☹

- Používají se tři zápisy IP adresy:
  - plné vyjádření - hhhh:hhhh:hhhh:hhhh:hhhh:hhhh:hhhh:hhhh
    - kde **h** je **hexa** číslice reprezentující **4 bity adresy**
    - příklad: ABCE:3:89AD:134:FEDC:E4D1:34:4321 {**vedoucí nuly se nemusí uvádět**}
  - zkrácený zápis pomocí zdvojené dvojtečky
    - zdvojená dvojtečka se může v adrese vyskytnout **pouze jednou**
    - zdvojená dvojtečka nahrazuje libovolné množství čtveřic nul
    - příklad: adresu 12A1:0:0:0:5:15:500C:44 je možné zapsat: 12A1::5:15:500C:44
    - adresu 1234:0:0:0:0:0:0:14 je možné zapsat jako 1234::14
    - adresu 0:0:0:0:0:0:0:1 je možné zapsat jako ::1
  - kombinovaný zápis h---h:d.d.d.d sloužící zejména v heterogenních sítích
    - příklad: 1234::195.47.103.12
      - = adresa 1234:0:0:0:0:0:C32F:670C
- Adresy sítí se zapisují podobně jako v IPv4:
  - např.: 80:1::1/64

jenom je tam až 128 jedniček

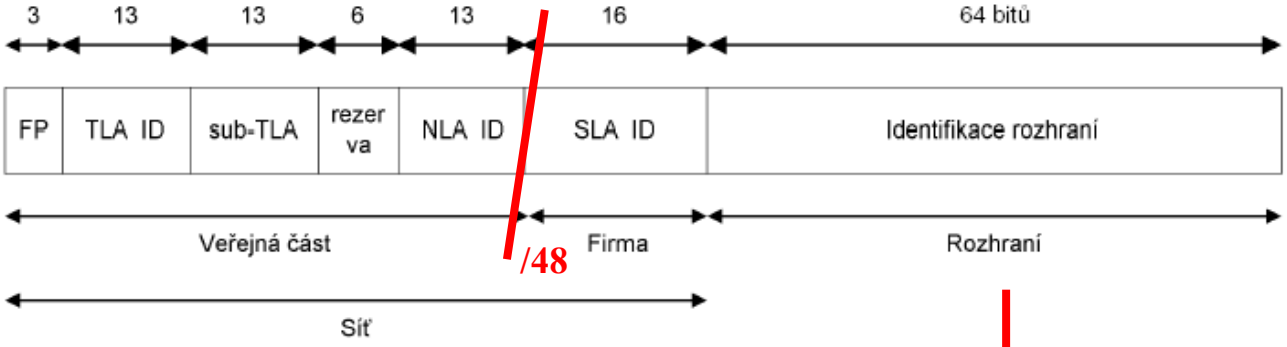
Adresní prostor

- Značný adresní prostor IPv6 ($2^{128} = 3{,}4 \times 10^{38}$) je rozdělen:
  - 0:0:0:0:0:0:0:0 - nespecifikovaná adresa
    - nepřiřazuje se
    - pokud je použita, znamená to, že rozhraní ještě nebyla adresa přiřazena
  - **0:0:0:0:0:0:0:1 (::1) - smyčka**
    - loopback - obdoba 127.0.0.1
- 0012/3 - Unicast adresy

- 2001::/16 - adresy přidělované Internet Registry poskytovatelům
  - 2001:0000::/29 - 2001:01F8::/29 - IANA
  - 2001:0200::/29 - 2001:03F8::/29 - APNIC (Asie)
  - 2001:0400::/29 - 2001:05F8::/29 - ARIN (Amerika)
  - **2001:0600::/29 - 2001:07F8::/29 - RIPE NCC (Evropa)**
  - 2002::/16 - tunelování 6to4
  - 1111 1110 102/10 (např. FE80::) - jednoznačné adresy v rámci LAN
  - 1111 1110 11_2/10 (např. FEC0::) - jednoznačné adresy v rámci firmy
  - 3FFE::/16 – experimentální síť 6Bone (přestala platit 6.6.2006)
  - FF/8 - oběžníky (multicast)

Jednoznačné adresy

- Rozdělení adres pro IPv6 se řídí následujícím schématem

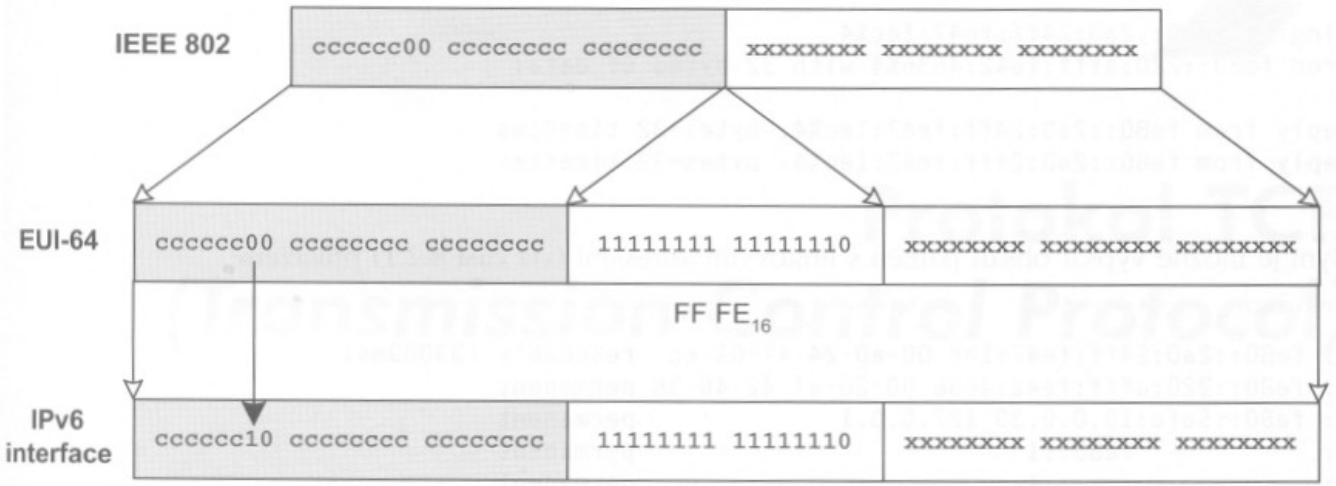


- Položky
  - FP + TLA ID – specifikuje účel použití adres
    - 2001:/16 - adresy určené pro poskytování Internetu
  - SUB TLA – rozdělení podle regionů
    - pro Evropu 2001:0600::/29 až 2001:07F8::/29
  - NLA ID – přidělováno poskytovatelům – ti rozdělují firmám
  - SLA ID – firemní prostor
    - 2B = 65536 sítí po 2^64 uzlech
  - Jak ale jednoznačně určit zbylých 64 bitů adres uzlů?

Určení jednoznačných adres

- Jednoznačné určení identifikace rozhraní
  - je odvozeno od adresace IEEE 802.X – MAC adresy
    - první tři B identifikují výrobce
    - další tři B jsou výrobcem zařízení přiděleny
  - MAC adresa je ale 6B
    - proto je nutná konverze

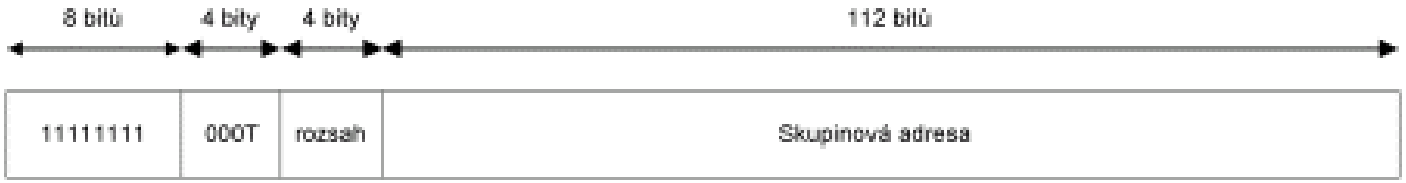
první vážná kritika:
omezení práv a soukromí uživatelů – jednoduchá detekce kdo co kdy kde dělal...



- Příklad:
  - MAC = **00-A0-24-47-01-EC** ⇨ IPv6 adresa = **:02A0:24FF:FE47:01EC**

Oběžníky

- Oběžníky mají v prvním B samé jedničky – FF::

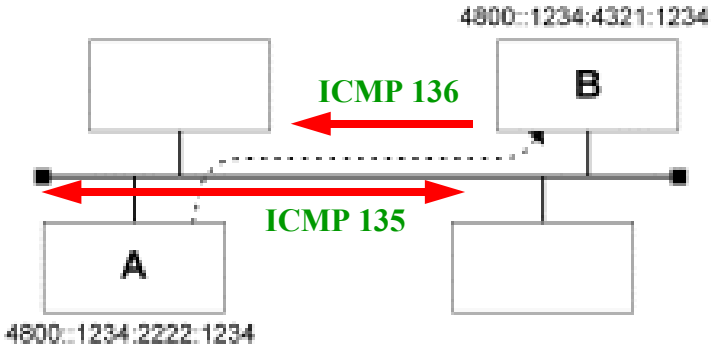


- Druhý B je rozdělen:
  - v druhé části jsou neseny dodatečné informace:
    - 1 – oběžník v rámci lokálního uzlu
    - 2 – oběžník v rámci LAN (II. vrstva)
    - 5 – oběžník v rámci sítě
    - 8 – oběžník v rámci firmy
    - E – globální oběžník
  - Existují vyhrazené oběžníky, např.:
    - FFxx::1 – oběžník pro všechny stanice (počítače i směrovače)
    - FFxx::2 – oběžník pro všechny směrovače
    - FFxx::9 – oběžník pro všechny směrovače provozující protokol RIPatd.
      - konkrétně FF02::2 je oběžník určený všem směrovačům na LAN...

ICMPv6

Na rozdíl od IPv4 několik ochran:
- regulace šířky pásma ICMPv6 – ochrana před DoS
- autentizace ICMPv6 zpráv – pouze důvěryhodné zdroje...

- Podobně, jako u IPv4 se o signalizaci chybových stavů stará:
  - Internet Control Message Protocol verze 6
    - protokol ICMPv6 ale řeší nové **zcela odlišné funkce oproti ICMP**
    - řeší například **překlad IP adres** na linkové adresy
      - o to se v IPv4 staraly protokoly ARP a RARP
  - z hlediska struktury se ICMPv6 jeví jako protokol vyšší vrstvy
  - ICMPv6 zprávy se dělí na dva typy
    - interval 0 - 127 - chybové zprávy → smysl obdobný jako v ICMPv4
    - interval 128 - 255 - informativní zprávy
  - příklad: získání HW adresy souseda
    - podobně je možné získat např.
    - adresu implicitní brány...



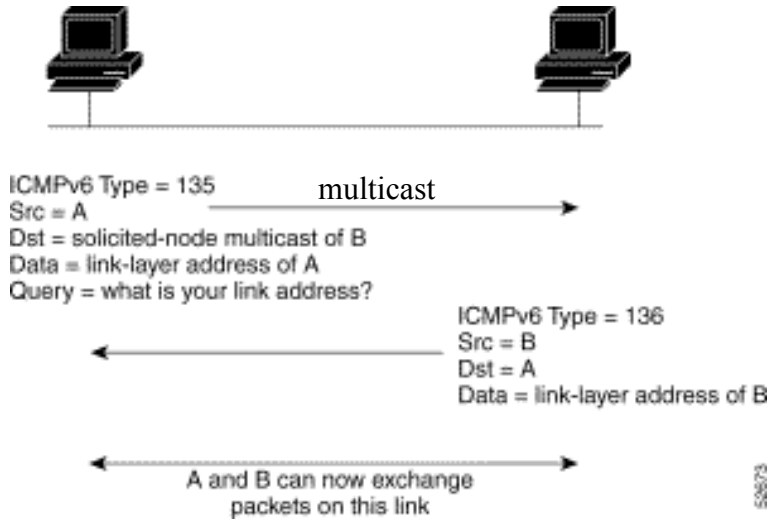
Autokonfigurace

zejména v případě WiFi
a přestupu mezi sítěmi

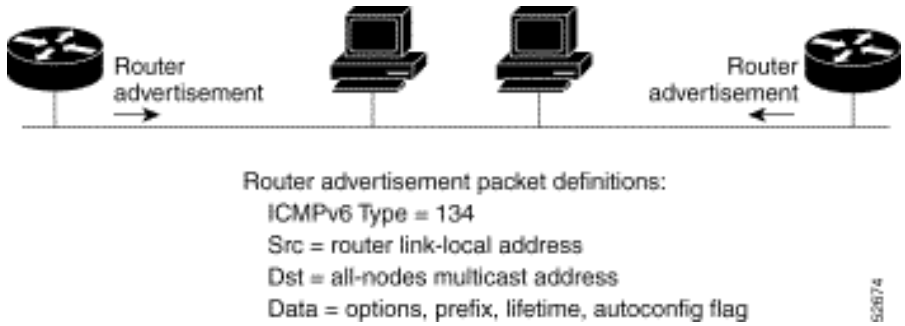
- Jedna z vlastností, která dělá IPv6 atraktivní
  - Idea: zařízení se po připojení do sítě dokáže automaticky zkonfigurovat
  - v IPv4 je toto řešeno pomocí nadstavbových protokolu DHCP
    - nevýhodou je nutnost správy DHCP serverů
    - co když ale chceme jen propojit dvě zařízení? **– jak zařídit autokonfiguraci?**
  - Funkce:
    1. zařízení samo vyhledá své sousedy
    2. zařízení se po připojení do sítě dotáže pomocí multicastu na svou identitu a nechá si přidělit adresu od routeru
- Používají se dva mechanismy
  - Neighbor Discovery
    - ND je proces, při kterém zařízení objevuje na síti ostatní IPv6 zařízení
  - Router Discovery
    - proces objevování routerů a získávání informací od nich

Autokonfigurace

- Neighbor Discovery
  - myšlenka: zařízení samo najde své sousedy a zahájí s nimi komunikaci
  - používá **ICMP pakety:**
    - 135 - neighbor solicitation message
    - 136 - neighbor advertisement message



- Router Discovery
  - proces objevování routerů
    - router může odpovědět na ICMPv6 typ 135
    - nebo se sám ohlásí sám
    - nebo odpoví na ICMPv6 typ 133 - Router solicitation paket
    - typicky je 133 odvysílán stanicemi po bootu



- správně zkonfigurovaný router posílá Router advertisement zprávy do dané sítě periodicky, kde jsou také k dispozici ostatním routerům

Autokonfigurace

paket odvysílaný po bootu
typ 133: „*hledám router*“

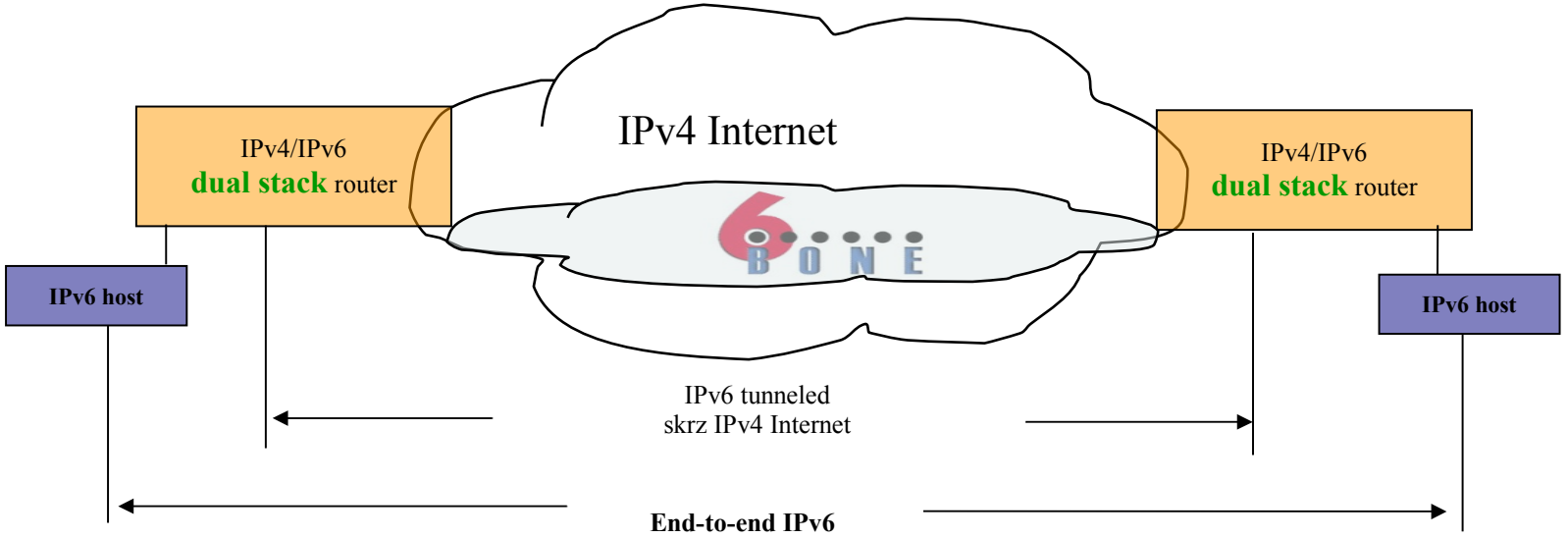
- Pokud zařízení posílající Router *solicitaion* paket má ručně nakonfigurovánu unicast adresu
  - použije jí a zdrojovou adresa je v tomto paketu
- v opačném případě

zřízení ještě není nakonfigurováno

  - se jedná o **nespecifikovanou** IPv6 adresa typu unicast, **tedy 0:0:0:0:0:0:0:0**
  - pokud router odpoví, je v Router advertisement paketu nová adresa
- Pomocí Router advertisement zpráv tedy
  - **probíhá bezestavová konfigurace** zařízení ve vnitřní síti
  - bezstavová znamená, že nikde **neexistuje tabulka**, která by k **linkové adrese přiřazovala IPv6 adresu**
  - toto řešení je výhodnou alternativou oproti protokolu DHCP v IPv4, který pracuje na transportní vrstvě a je stavový
    - to s sebou nese v případě velkým sítí velkou režii na správu.
- Nevýhodou autokonfigurace je absence informace o DNS serverch
  - ty si musí uživatel (při současném návrhu autokonf. protokolů) nastavit ručně

Testovací období (6Bone: *1996 - † 6.6.2006)

- Pro fungování IPv6 byla velice důležitá experimentální síť 6Bone *
  - síť začala jako virtuální s pomocí tunelování IPv6 nad IPv4
  - jejím hlavním účelem bylo testování standardů a implementací IPv6
    - síť byla dále zdokonalována a postupně přešla k nativní IPv6 síti
    - počítače zapojené do 6Bone měly na rozhraní IP 3ffe::/16



- Pro představu:
  - maximální „popularita“ kolem r. 2003 – asi 1000 sítí z 50 zemí

Současný stav

- Základní vývoj IPv6 dokončen
  - Revize základních dokumentů
  - Postupem času se ukázalo, že bez některých věcí to prostě nepůjde
    - dobrým příkladem je např. DHCPv6, DNSv6
- V podstatě již vyřešena i implementační část
  - velcí hráči na trhu již nemají problém (Cisco, Microsoft...)
- IPv6 již existuje zcela separátně a nezávisle na IPv4
  - páteřní síť
- Velký problém u lokálních ISP
  - zatím stále fáze „oťukávání“
  - nikomu se moc nechce „bourat“ stávající fungující stav
  - nízký tlak „zespoda“
- První vážnější kritika
  - rozsáhlý adresní prostor - plýtvání

Podpora IPv6 v zařízeních

- Implementace IPv6 v zařízeních se vyvíjela dosti živelně
  - často byla implementována pouze podmnožina
  - → **rozčarování uživatelů, zoufalá podpora**

- Myšlenka:
  - u *WiFi* se velmi osvědčila *WiFi aliance*
  - na podobném principu vzniklo ***IPv6 fórum***
  - testování kompatibility IPv6
    - 1. fáze – základní kompatibilita
      - IPv6 addressing, ICMPv6, Neighbor Discovery, bezest. aut. konfigurace
    - 2. fáze - včetně doporučených prvků
      - IPsec, mobilní IP, DHCPv6...

Podpora IPv6 v OS

- Moderní distribuce Linuxu již IPv6 plně podporují (částečně již od jádra 2.1.8)
  - ve většině případů není třeba ani nic nastavovat
  - informace lze získat po zadání příkazu *ifconfig*
    - **dokonce dokáží fungovat i bez IPv4**
- OS Win XP/2000/Vista implementují IPv6 jako samostatný na IPv4 nezávislý protokol
  - IPv6 je třeba nejprve nainstalovat příkazem *ipv6 install*
    - v OS Longhorn (Vista) je již možné přidat IPv6 za pomocí ovládacího panelu
  - po instalaci přibudou rozhraní - příkaz *ipv6 if*
    - loopback
    - tunelování IPv6 přes IPv4
    - síťová karta (FE80::hhh...)
  - otestovat připojení je možné pomocí příkazů:
    - ping6, tracert6

  - propojíme li dvě PC na lokální síti, je ihned možné komunikovat

Podpora IPv6 ve Windows

- Pro management mohou pomoci tyto příkazy:
  - ipv6 rc – View the route cache
  - ipv6 nc – View the neighbor cache
  - ipv6 if – View interface information
  - ipv6 ifc – Configure interface attributes
  - ipv6 rtu – Add IPv6 route
  - ipv6 adu – Configure IPv6 with manual addresses

Odkazy - literatura

- Literatura:
  - Strapa P.: „IPv6“, CZ.NIC - CESNET 2008, ISBN: 978-80-904248-0-7
    - dostupná zdarma ke stažení
- Zajímavé zdroje:
  - http://www.potaroo.net/tools/ipv4/index.html
    - – automaticky generované statistiky využití adreního prostoru
  - http://www.potaroo.net/ispcolumn/2003-07-v4-address-lifetime/ale.html
    - – automaticky generované statistiky využití adreního prostoru
  - www.ipv6.org
  - www.ipv6.cz
    - - asi nejobsáhlejší “wiki” servery o IPv6

Konec přednášky

Děkuji vám za trpělivost