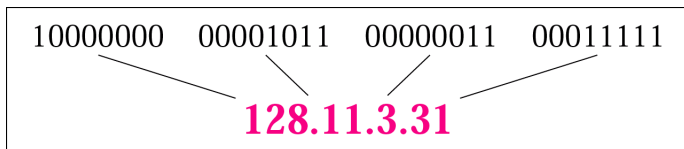


Struktura přednášky

- 1 Přehled
- 2 Úvod
- 3 Poskytované služby
- 4 Internetworking
- 5 Adresace**
 - IPv4: typy adres
 - IPv4: Classful Addressing
 - IPv4: Classless Addressing
 - IPv4: Network Address Translation (NAT)
 - IPv4: Vyčerpávání adresového prostoru
 - IPv6 adresy
- 6 Interakce L3 se spojovou vrstvou (L2)
 - ARP protokol
- 7 IP protokol
 - IP protokol verze 4 (IPv4)
 - ICMP
 - IP protokol verze 6 (IPv6)
 - ICMPv6
 - Mechanismy pro podporu přechodu IPv4 → IPv6
 - IPv6: Literatura
- 8 IPv6: Kde se nacházíme?

Adresace na L3

- požadavek *jednoznačné identifikace* každého zařízení připojeného k Internetu
- nutnost *systematického přidělování adres*
 - za účelem snadnějšího směrování
- každému zařízení/rozhraní přiřazena *Internetová adresa (IP adresa)*
 - *IPv4 adresa* (32 bitů) vs. *IPv6 adresa* (128 bitů)



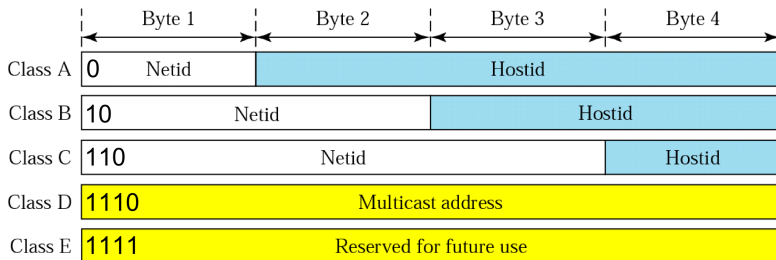
IPv4 – typy adres

- *Individuální (unicast) adresy* – identifikace jednoho síťového rozhraní
 - identifikace jediného odesílatele/příjemce
- *Broadcast adresy* – slouží pro zasílání dat všem možným příjemcům na dané LAN („all-hosts broadcast“)
 - zdrojová adresa datagramu (identifikace odesílatele) je unicastová
- *Skupinové (multicast) adresy* – slouží pro adresování skupiny příjemců (síťových rozhraní), kteří o data **projevili zájem**
 - data směrovači rozesílána všem členům skupiny
 - zdrojová adresa datagramu (identifikace odesílatele) je unicastová

Přidělování adres – Classful Addressing

• Classful Addressing:

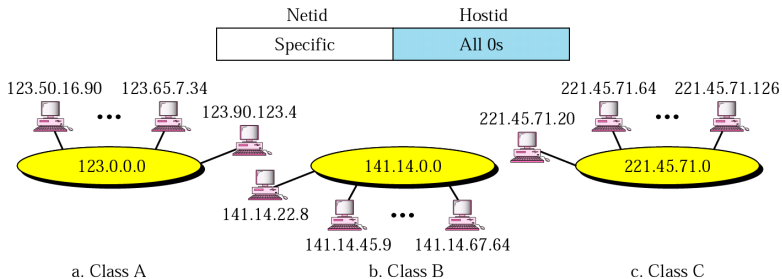
- zcela první metoda přidělování adres
- adresní prostor rozdělen do 5 tříd:
 - **třída A:** 2^7 sítí, každá z nich 2^{24} uzlů
 - **třída B:** 2^{14} sítí, každá z nich 2^{16} uzlů
 - **třída C:** 2^{21} sítí, každá z nich 2^8 uzlů
 - **třída D:** multicastové adresy
 - **třída E:** rezervovaný prostor



Přidělování adres – Classful Addressing

NetID vs. HostID

- *Adresa sítě (NetID):*
 - identifikuje danou síť (nemůže být přidělena uzlu/rozhraní)
 - tuto identifikaci lze využít pro směrování (viz později)
- *Adresa uzlu/rozhraní (HostID):*
 - identifikuje jedinečný uzel v síti NetID



Příklad: HostID = 147.251.48.1 \Rightarrow třída B \Rightarrow NetID = 147.251.0.0

Problémy Classful adresování

- nedostatečná granularita – každá třída rozdělena na pevný počet sítí s pevnou maximální velikostí
 - = plýtvání adresním rozsahem
 - organizace chce využít 10 IP adres? Dostane C třídu (256 adres)
 - organizace chce využít 270 IP adres? Dostane B třídu (65536 adres)
 - organizace chce využít 70000 IP adres? Dostane A třídu (2097152 adres)
 - *možné řešení*: přidělování více síťových adres menší třídy
- popsání řešení generuje nárůst *směrovacích tabulek*
 - roste objem směrovacích informací, které musí být zpracovávány při rozhodování o volbě dalšího směru procházejícího paketu
 - nutnost prohledávání tabulek (lineární složitost)

Ilustrace problému: organizace s 1500 uzly

- ① přidělena adresa třídy B \Rightarrow zabráno 65536 adres \Rightarrow 1 záznam ve sm.tabulce
- ② přiděleno 8 adres třídy C \Rightarrow zabráno 2048 adres \Rightarrow 8 záznamů ve sm.tabulce

Problémy Classful adresování – řešení

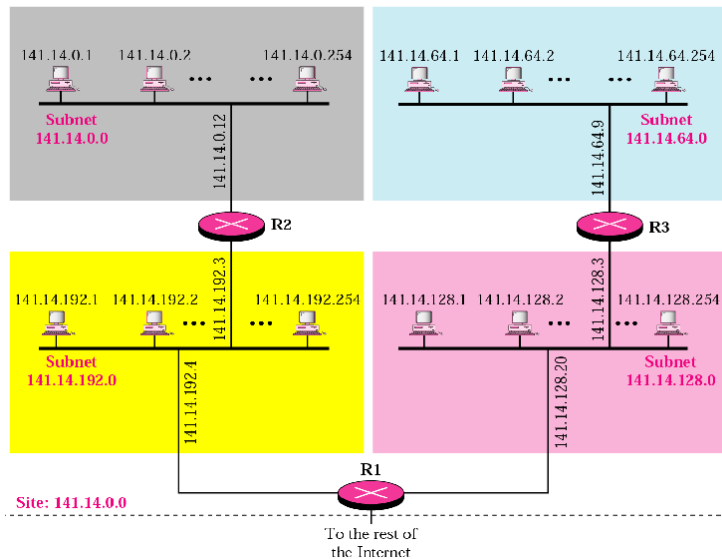
Subnetting, Supernetting

- Lze přidělenou adresu sítě dále dělit do menších podsítí?
 - např. rozdělení sítě dle organizačních složek v rámci jedné organizace
 - *Subnetting*
- Lze využít skutečnosti, že organizace má přidělen souvislý blok adres určité třídy?
 - a snižovat tak velikost směrovacích tabulek
 - *Supernetting*

Classful adresování – Subnetting

- standardní IP adresa poskytuje dvouúrovňovou hierarchii
 - adresa sítě a adresa uzlu
- *Subnetting* zavádí možnost tříúrovňové hierarchie
 - adresa sítě, adresa podsítě a adresa uzlu
 - využitelné v nějaké geograficky omezené oblasti (velké organizace, univerzity, ISPs)
 - síť rozdělena na menší podsítě (*subnetworks (subnets)*)
 - důležitý princip *uzavřenosti*:
 - „zvenčí“ (z pohledu Internetu) se jeví jako 1 síť (1 záznam ve sm. tabulkách), podsítě se rozlišují až na hraničním směrovači
 - tj. má pouze lokální platnost, nikoli platnost globální

Classful adresování – Subnetting



Classful adresování – Supernetting

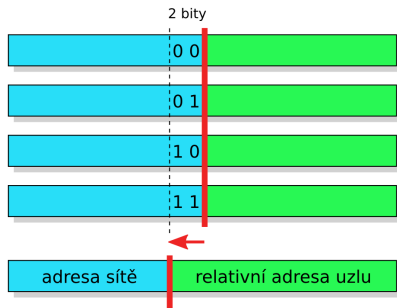
- *Supernetting:*

- pravý opak subnettingu, posouvá pomyslnou dělicí čáru mezi oběma složkami IP adresy směrem k vyšším bitům
- spojuje (agreguje) několik původně samostatných síťových IP adres v jednu výslednou
- musí však jít o „sousední“ síťové adresy
 - síťové IP adresy se musí shodovat v určitém počtu vyšších bitů své síťové části
 - a musí vyčerpávat všechny bitové kombinace v příslušném počtu nižších bitů (své síťové části)

Classful adresování – Subnetting vs. Supernetting



(a) Subnetting



(b) Supernetting

Classful adresování – Maska sítě/podsítě

- oba způsoby vyžadují mechanismus pro identifikaci bitů, které identifikují síť
 - v rámci subnettingu nezbytné jen na hraničních směrovačích
 - v rámci supernettingu nezbytné na všech směrovačích
- využitý mechanismus – *maska sítě*
 - 32-bitový řetězec (v rámci IPv4)
 - obsahuje 1 v těch bitech, které odpovídají síťové části adresy, 0 tam, kde jde o relativní adresu uzlu v rámci sítě
 - IP adresa uzlu && maska sítě = adresa sítě

| Class | Binary form | Decimal form | Using slash |
|-------|--|------------------------|-------------|
| A | 11111111 00000000 00000000 00000000 | 255.0.0.0 | /8 |
| B | 11111111 11111111 00000000 00000000 | 255.255.0.0 | /16 |
| C | 11111111 11111111 11111111 00000000 | 255.255.255.0 | /24 |
| --- | 11111111 11111111 000 00000000 00000000 | 255.248.0.0 | /13 |
| --- | 11111111 11111111 11111111 1 0000000 | 255.255.255.128 | /25 |

Přidělování adres – Classless Addressing

- do poloviny 90. let adresy přidělovány pouze v rámci tříd
 - nejmenší počet přidělených adres – 256 (třída C)
- *Classless Addressing*:
 - zobecnění a rozšíření subnettingu/supernettingu
 - zavádí zcela variabilní délku bloku adresy sítě
 - identifikace sítě = adresa sítě a maska sítě
 - adresy se přidělují *hierarchicky*
 - umožnění agregace směrování (viz později) \Rightarrow snaha o minimalizaci velikosti směrovacích tabulek
 - opodstatnění subnettingu zůstává

Classless Addressing – *Classless Inter-Domain Routing (CIDR)*

- konvence popisující „pravidla hry“ – použití IP adres, významu masek, supernetting a subnetting
- nahrazuje původní „třídní“ charakter IP adres (třídy A, B a C)
 - IP adresy přidělovány po tzv. *CIDR blocích*
- velikost CIDR bloku dána příslušnou maskou
 - možno velmi pružně přizpůsobovat
- ⇒ snížení tempa vyčerpávání adresového prostoru
- *Důsledek CIDRu*: adresy závislé na poskytovateli
 - původně IP adresy nezávislé na způsobu jejich připojení
 - zavedení závislosti
 - poskytovatel získává CIDR blok, který si rozděluje dle svého uvážení
 - vnější směrovače směřují jen na základě CIDR bloku
 - při změně poskytovatele je potřeba síť přeadresovat (přechíslovat)

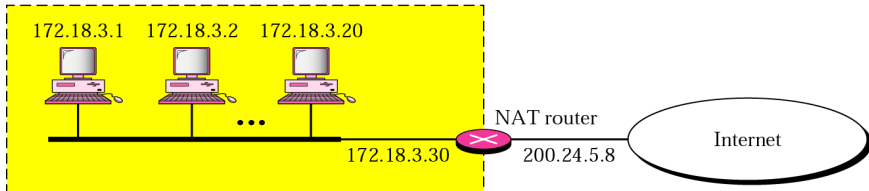
Network Address Translation (NAT)

- další mechanismus pro snížení tempa vyčerpávání adresového prostoru
- určeno zejména pro domácí uživatele
 - původně připojování modemů → možnost dynamického přidělování adres
 - nyní ADSL, kabelová připojení – (většinou) trvalá alokace adres
 - časté požadavky na přidělení více IP adres
- **řešení:** *Network Address Translation (NAT)*
 - „skrývání“ vnitřní sítě za jednu/několik externích adres
 - v rámci vnitřní sítě možnost využít mnoho interních adres
 - rezervované privátní adresy (viz obrázek), unikátní v rámci organizace
 - *vedlejší efekt:* ochrana vnitřní sítě
 - *překlad adres* procházejících síťovým prvkem (např. NAT směrovačem)

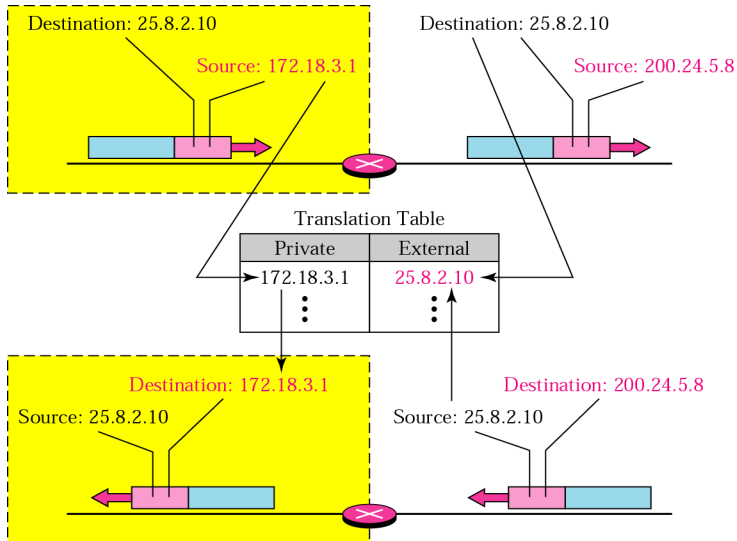
| Range | | | Total |
|-------------|----|-----------------|----------|
| 10.0.0.0 | to | 10.255.255.255 | 2^{24} |
| 172.16.0.0 | to | 172.31.255.255 | 2^{20} |
| 192.168.0.0 | to | 192.168.255.255 | 2^{16} |

Network Address Translation (NAT) – ilustrace

Site using private addresses



Network Address Translation (NAT) – překlad adres



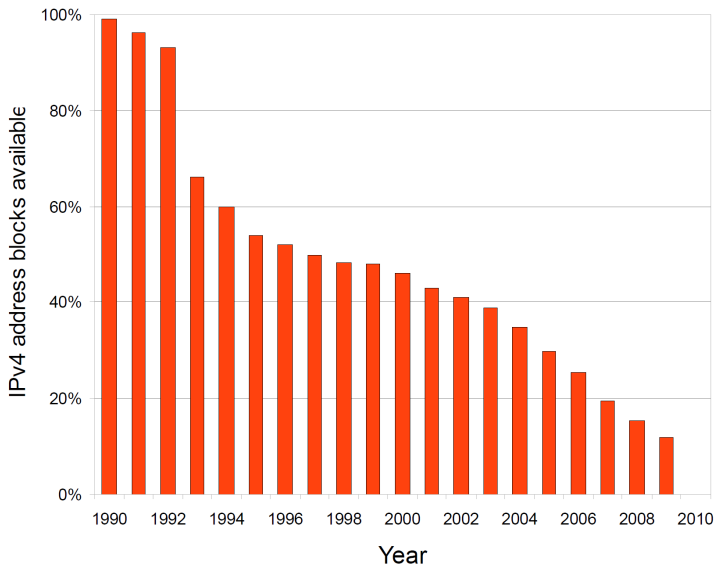
Network Address Translation (NAT) – překlad adres II.

- překlad adres odchozích paketů je triviální
- překlad adres příchozích paketů vyžaduje dodatečné informace:
 - kterému stroji z vnitřní sítě mají být data přeposlána?
 - *překladové tabulky (translation tables)*

| <i>Private Address</i> | <i>Private Port</i> | <i>External Address</i> | <i>External Port</i> | <i>Transport Protocol</i> |
|------------------------|---------------------|-------------------------|----------------------|---------------------------|
| 172.18.3.1 | 1400 | 25.8.3.2 | 80 | TCP |
| 172.18.3.2 | 1401 | 25.8.3.2 | 80 | TCP |
| ... | ... | ... | ... | ... |

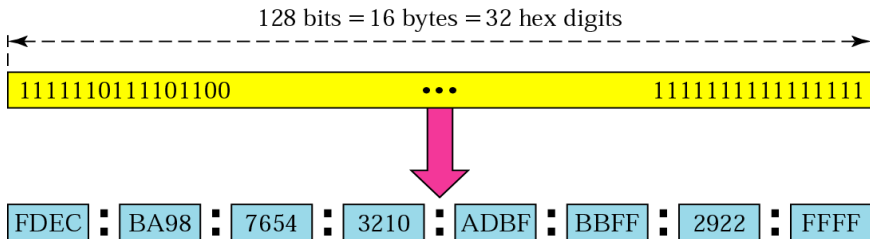
Obrázek: Ukázka překladové tabulky.

IPv4 – vyčerpávání adresového prostoru



IPv6 adresy

- adresy využívané protokolem IPv6 (viz dále)
- (prozatím) finální řešení nedostatku IP adres
- IPv6 adresa má 128 bitů (= 16 bajtů):
 - 2^{128} možných adres ($\approx 3 \times 10^{38}$ adres $\Rightarrow \approx 5 \times 10^{28}$ adres na každého obyvatele Země)
 - hexadecimální zápis místo dekadického (po dvojicích bajtů oddělených znakem „:“)



IPv6 adresy – zkracování zápisu

Úvodní nuly lze ze zápisu každé skupiny vynechat:

- 0074 lze psát jako 74, 000F jako F, ...
- 3210 **nelze** zkracovat!

Unabbreviated

FDEC : BA98 : 0074 : 3210 : 000F : BBFF : 0000 : FFFF



FDEC : BA98 : 74 : 3210 : F : BBFF : 0 : FFFF

Abbreviated

Sekvence po sobě jdoucích nulových skupin lze vynechat:

- vždy však **pouze jednu sekvenci** takovýchto nulových skupin!

Abbreviated

FDEC : 0 : 0 : 0 : 0 : BBFF : 0 : FFFF



FDEC : : BBFF : 0 : FFFF

More Abbreviated

IPv6 adresy – hierarchie

- cílem opět usnadnění směrování
- strukturu individuálních IPv6 adres definuje RFC 3587
- základní struktura:

| n bitů | 64-n bitů | 64 bitů |
|---------------------------|------------------|-----------------|
| globální směrovací prefix | adresa podsítě | adresa rozhraní |

- globální směrovací prefix \approx adresa sítě
- adresa podsítě obvykle 16 bitů \Rightarrow globální prefix 48 bitů
 - prvních 16 bitů obsahuje hodnotu 2001_{16}
 - dalších 16 bitů přiděluje regionální registrátor (RIR)
 - dalších 16 bitů přiděluje lokální registrátor (LIR)

| 16 bitů | 16 bitů | 16 bitů | 16 bitů | 64 bitů |
|----------------|----------------|----------------|----------------|-----------------|
| 2001 | přiděluje RIR | přiděluje LIR | adresa podsítě | adresa rozhraní |

IPv6 adresy && CIDR

- IPv6 adresace je pouze *classless*, třídy neexistují
- sítě v IPv6 popisovány s využitím notace CIDR (stejně jako v IPv4)
- např. *FDEC:0:0:0:0:BBFF:0:FFFF/60*

IPv6 adresy – typy adres

- *Individuální (unicast) adresy* – totéž co v IPv4, identifikace jednoho síťového rozhraní
- *Skupinové (multicast) adresy* – totéž co v IPv4, slouží pro adresování skupin počítačů či jiných síťových zařízení
 - data jsou vždy doručena všem členům skupiny
 - prefix `ff00::/8`
- *Výběrové (anycast) adresy* – novinka v IPv6
 - také označují skupinu příjemců
 - data se však doručí jen jedinému jejímu členovi (tomu, který je nejbližší)
- broadcast adresy IPv4 protokolu se v IPv6 nevyužívají
 - nahrazeny speciálními multicastovými skupinami (např. všechny uzly na dané lince)

Struktura přednášky

- 1 Přehled
- 2 Úvod
- 3 Poskytované služby
- 4 Internetworking
- 5 Adresace
 - IPv4: typy adres
 - IPv4: Classful Addressing
 - IPv4: Classless Addressing
 - IPv4: Network Address Translation (NAT)
 - IPv4: Vyčerpávání adresového prostoru
 - IPv6 adresy
- 6 Interakce L3 se spojovou vrstvou (L2)**
 - **ARP protokol**
- 7 IP protokol
 - IP protokol verze 4 (IPv4)
 - ICMP
 - IP protokol verze 6 (IPv6)
 - ICMPv6
 - Mechanismy pro podporu přechodu IPv4 → IPv6
 - IPv6: Literatura
- 8 IPv6: Kde se nacházíme?

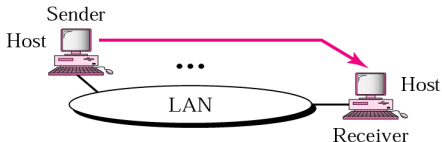
Interakce L3 se spojovou vrstvou (L2) – mapování adres

- mechanismus doručení dat v IP sítích – *hop-by-hop*
- vlastní předání/doručení zprávy na základě *fyzických (MAC) adres*
- 2 alternativy:
 - příjemce na stejné LAN jako odesílatel
 - IP datagram obsahuje IP adresu příjemce, rámec L2 vrstvy MAC adresu příjemce
 - příjemce na jiné LAN než odesílatel
 - IP datagram obsahuje IP adresu příjemce, rámec L2 vrstvy MAC adresu směrovače
 - směrovač po přijetí (a zpracování) datagramu jej vloží do nového rámce s MAC adresou dalšího směrovače ve snaze přiblížit se cíli (odtud *hop-by-hop*)
 - po dosažení cílové LAN platí alternativa 1 (lokální „odesílatel“ = poslední směrovač)

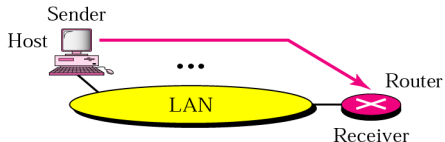
Interakce L3 se spojovou vrstvou (L2) – mapování adres II.

- \Rightarrow nutnost mapování IP adres na fyzické (MAC) adresy
 - *statické mapování*
 - vytvoření statické tabulky párů (*IP adresa, MAC adresa*)
 - obtížně spravovatelné
 - *dynamické mapování*
 - **Address Resolution Protocol (ARP)**

Interakce L3 se spojovou vrstvou (L2) – mapování adres III.



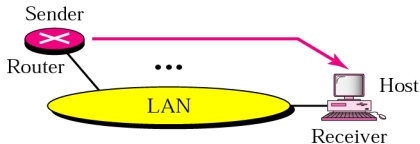
Case 1. A host has a packet to send to another host on the same network.



Case 2. A host wants to send a packet to another host on another network. It must first be delivered to the appropriate router.



Case 3. A router receives a packet to be sent to a host on another network. It must first be delivered to the appropriate router.

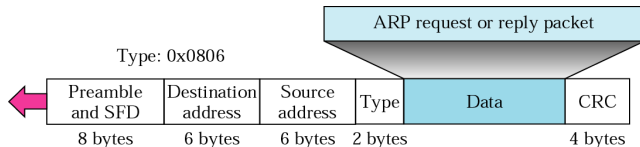


Case 4. A router receives a packet to be sent to a host on the same network.

Obrázek: Případové ilustrace využití ARP protokolu (*hop-by-hop* doručení).

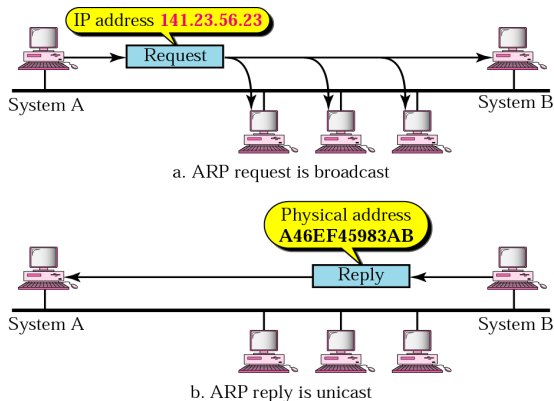
Interakce L3 se spojovou vrstvou (L2) – ARP protokol

- protokol pro zjištění MAC adresy uzlu/směrovače na základě IP adresy
- mechanismus:
 - ➊ zaslání tzv. *ARP request* paketu **všem** uzlům na dané LAN (broadcast)
 - paket obsahuje IP & MAC adresu odesílatele a IP adresu hledaného uzlu
 - ➋ paket zpracován všemi uzly; odpoví jen ten, jehož IP adresa se shoduje s hledanou
 - ostatní paket zahodí
 - ➌ hledaný uzel žadateli odpovídá tzv. *ARP reply* paketem
- ARP pakety baleny přímo do rámců L2 vrstvy



- protokol **RARP (Reverse Address Resolution Protocol)**
 - zpětný překlad MAC adres na IP adresy; již se nevyužívá

Interakce L3 se spojovou vrstvou (L2) – ARP protokol II.



Obrázek: Ilustrace mechanismu operace ARP protokolu.

- více viz animace:

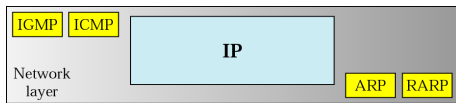
http://frakira.fi.muni.cz/~jeronimo/vyuka/OsiSchool_ARP.swf

Struktura přednášky

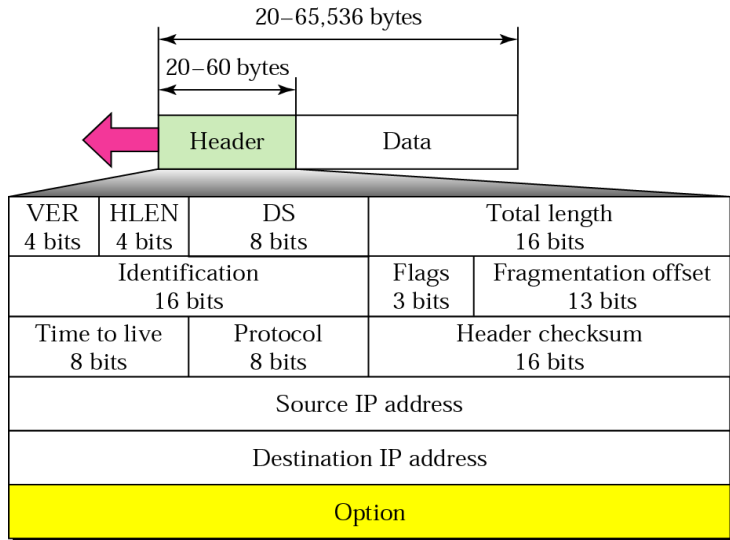
- 1 Přehled
- 2 Úvod
- 3 Poskytované služby
- 4 Internetworking
- 5 Adresace
 - IPv4: typy adres
 - IPv4: Classful Addressing
 - IPv4: Classless Addressing
 - IPv4: Network Address Translation (NAT)
 - IPv4: Vyčerpávání adresového prostoru
 - IPv6 adresy
- 6 Interakce L3 se spojovou vrstvou (L2)
 - ARP protokol
- 7 IP protokol
 - IP protokol verze 4 (IPv4)
 - ICMP
 - IP protokol verze 6 (IPv6)
 - ICMPv6
 - Mechanismy pro podporu přechodu IPv4 → IPv6
 - IPv6: Literatura
- 8 IPv6: Kde se nacházíme?

Internet Protocol (IP protokol)

- nejrozšířenější protokol síťové vrstvy
 - doprava dat (datagramů) na místo jejich určení, a to i přes mezilehlé uzly (směrovače) – *host-to-host delivery*
 - uzly/rozhraní v rámci IP protokolu jednoznačně identifikovány IP adresami
 - využívá *datagramový přístup* k přepínání paketů, komunikace je *nespojovaná*
 - ⇒ směrování (příští přednáška)
 - poskytuje nespolehlivou (tzv. *best-effort*) službu
 - doplněn dalšími podpůrnými protokoly (ICMP, ARP, RARP, IGMP)
 - ošetření nestandardních situací, šíření informací potřebných ke korektnímu směrování, identifikace rozhraní na LAN atd.
- navržen a standardizován ve dvou verzích:
 - *Internet Protocol verze 4 (IPv4)* – 1981, RFC 791
 - *Internet Protocol verze 6 (IPv6)* – 1998, RFC 2460



IPv4 datagram

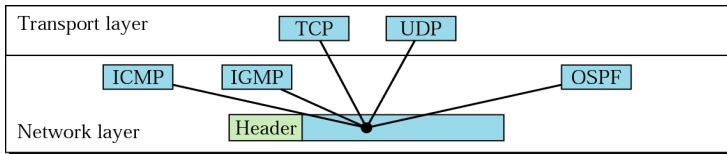


IPv4 datagram II.

- **Version (VER)** – verze IP protokolu
- **Header length (HLEN)** – délka hlavičky IP datagramu (ve 4B slovech)
 - nezbytné kvůli poli *Option* (proměnná délka datagramu)
- **Differentiated services (DS)**, také **Type of service (TOS)** – třída datagramu v rámci kvality služby (QoS)
 - nezbytné pro odlišení „důležitých“ (řídící datagramy, provoz v reálném čase) a „méně důležitých“ datagramů
 - později (konec semestru)
- **Total length** – délka celého IP datagramu (v B)
 - max. $2^{16} - 1 = 65535$ bajtů
- **Identification, Flags, Offset** – viz Fragmentace v IPv4, slide 55
- **Time to live (TTL)** – řízení maximálního počtu skoků (= směrovačů) navštívených datagramem
 - odesílací uzel vloží číslo ($\approx 2 \times$ největší počet skoků mezi libovolnými dvěma uzly)
 - po průchodu směrovačem TTL dekrementováno o 1
 - pokud po dekrementování platí $TTL = 0$, datagram je zahozen

IPv4 datagram III.

- **Protocol** – identifikace protokolu vyšší vrstvy využívajícího služeb IP vrstvy
 - nezbytné pro specifikaci cílového protokolu, kterému má být datagram doručen
 - forma multiplexingu/demultiplexingu
 - identifikátory určeny v online databázi asociace IANA
 - např. 1 = ICMP, 2 = IGMP, 6 = TCP, 17 = UDP, atd.
 - viz <http://www.iana.org/assignments/protocol-numbers>

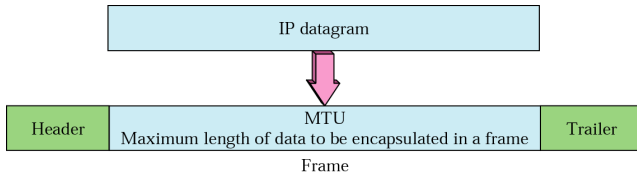


IPv4 datagram IV.

- **Header checksum** – kontrolní součet *hlavičky* IP datagramu
 - bez dat
 - data (resp. transportní protokoly) mají vlastní kontrolní součty
 - hlavní důvod pro zdvojení:
 - nutnost přepočítávání kontrolního součtu na směrovačích díky proměnlivým polím IP datagramu (např. TTL)
 - \Rightarrow počítání kontrolního součtu jen hlavičky = úspora času (data se stejně nemění)
- **Source IP address, Destination IP address** – 32-bitová IPv4 adresa identifikující odesílací/přijímající uzel
- **Options** – volitelná součást IP datagramů, určeno zejména pro budoucí rozšíření IPv4
- **Data** – vlastní přenášená data

IPv4 – fragmentace datagramů

- datagram při cestě k cíli prochází různými sítěmi
- ne všechny sítě (resp. využitě L2 protokoly) mohou přenášet data stejné velikosti
- *Maximum Transfer Unit (MTU)* – maximální velikost dat, které lze přenést využitým L2 protokolem
 - určuje maximální velikost přenositelného IP datagramu (*Total size*)



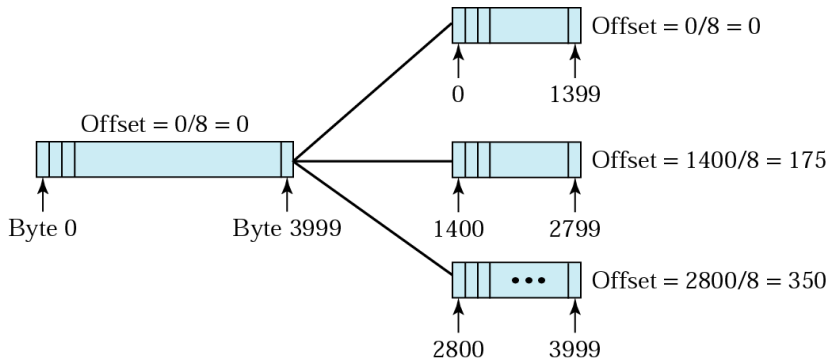
IPv4 – fragmentace datagramů II.

- *situace:*
 - zdrojový uzel chce odeslat datagram, který je větší než MTU výstupní linky
 - směrovač přijme datagram, který je větší než MTU výstupní linky
- *řešení:* provedení tzv. *fragmentace IP datagramu*
 - původní datagram je rozdělen na několik menších datagramů (tzv. *fragmenty*)
 - každý fragment získá svou vlastní IP hlavičku (= stane se z něj nový, plnohodnotný datagram)
 - fragmenty na cílovém uzlu složeny do původního datagramu (před předáním transportnímu protokolu)
- složení fragmentů do původního datagramu vyžaduje:
 - identifikaci datagramu, kterému fragmenty náleží
 - znalost počtu fragmentů
 - znalost pozice každého fragmentu v původním datagramu
- využití polí IP hlavičky: *Identification*, *Flags* a *Offset*

IPv4 – fragmentace datagramů III.

- **Identification** – pole identifikuje původní datagram, kterému fragmenty náleží
 - tj. všechny fragmenty jednoho datagramu mají stejné identifikační číslo
- **Flags** – 3-bitová hodnota:
 - 1 bit rezervovaný
 - *do-not-fragment bit* – hodnota 1 = datagram nesmí být fragmentován (v případě nutnosti generována ICMP zpráva – viz dále)
 - *more-fragment bit* – hodnota 1 = fragment není posledním fragmentem (0 určuje poslední fragment daného datagramu)
- **Offset** – relativní pozice fragmentu v původním datagramu
 - 13 bitů \Rightarrow offset max. 8191 \Rightarrow nelze pokrýt větší datagramy
 - \Rightarrow jednotka offsetu stanovena na 8 B

IPv4 – fragmentace datagramů IV.



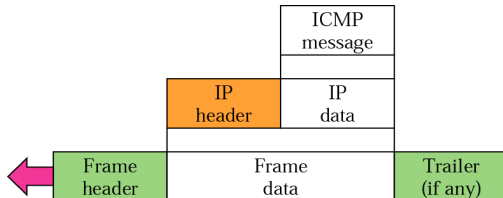
Obrázek: Ukázka fragmentace 4000B datagramu do 3 fragmentů.

IPv4 – fragmentace datagramů V.

- Kde se fragmentace provádí?
 - na zdrojovém uzlu
 - na směrovači/směrovačích
- Kde se provádí skládání fragmentů?
 - **pouze** na cílovém uzlu
 - ztráta fragmentu = ztráta datagramu
 - na směrovačích nelze skládat ze dvou důvodů:
 - zbytečná zátěž směrovače
 - fragmenty putují sítí nezávisle na sobě (tj. i jinými cestami)
- Možno provádět vícenásobnou fragmentaci
 - „fragmentaci fragmentu“
 - *otázka*: jak bude vypadat hlavička fragmentů fragmentu?

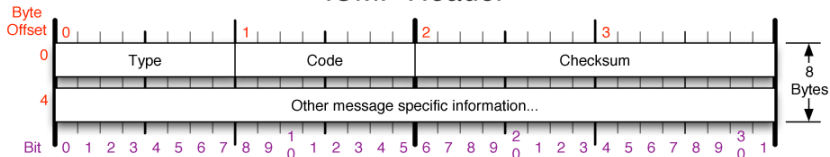
Internet Control Message Protocol (ICMP)

- IP protokol poskytuje nespolehlivou (best-effort) službu
 - bez mechanismů pro informování odesílatele o vzniklých chybách
 - bez podpůrných mechanismů pro zjišťování stavu sítě
- *Internet Control Message Protocol (ICMP)*
 - RFC 792
 - doprovodný protokol IP protokolu
 - poskytuje informace o chybách při přenosu IP datagramů
 - poskytuje základní informace o stavu sítě
- přestože je ICMP protokolem síťové vrstvy, zprávy nejsou předávány linkové vrstvě, ale baleny do IP protokolu
 - hodnota pole *Protocol* v hlavičce IP datagramu nastavena na 1



Internet Control Message Protocol (ICMP) – hlavička

ICMP Header



ICMP Message Types

| Type | Code/Name | Type | Code/Name | Type | Code/Name |
|------|---|------|---|------|----------------------|
| 0 | Echo Reply | 4 | Source Quench | 13 | Timestamp |
| 3 | Destination Unreachable | 5 | Redirect | 14 | Timestamp Reply |
| 0 | Net Unreachable | 0 | Redirect Datagram for the Network | 15 | Information Request |
| 1 | Host Unreachable | 1 | Redirect Datagram for the Host | 16 | Information Reply |
| 2 | Protocol Unreachable | 2 | Redirect Datagram for the TOS & Network | 17 | Address Mask Request |
| 3 | Port Unreachable | 3 | Redirect Datagram for the TOS & Host | 18 | Address Mask Reply |
| 4 | Fragmentation required, and DF set | 8 | Echo | 30 | Traceroute |
| 5 | Source Route Failed | 9 | Router Advertisement | | |
| 6 | Destination Network Unknown | 10 | Router Selection | | |
| 7 | Destination Host Unknown | 11 | Time Exceeded | | |
| 8 | Source Host Isolated | 0 | TTL Exceeded in Transit | | |
| 9 | Network Administratively Prohibited | 1 | Fragment Reassembly Time Exceeded | | |
| 10 | Host Administratively Prohibited | 12 | Parameter Problem | | |
| 11 | Network Unreachable for TOS | 0 | Pointer indicates the error | | |
| 12 | Host Unreachable for TOS | 1 | Missing a Required Option | | |
| 13 | Communication Administratively Prohibited | 2 | Bad Length | | |

Checksum

Checksum of entire
UDP segment and
pseudo header (parts
of IP header) (for UDP)

Checksum of ICMP
header (for ICMP)

RFC 768 and 792

Please refer to RFC
768 for the complete
User Datagram
Protocol (UDP)
Specification, and to
RFC 792 for the
Internet Control
Message protocol
(ICMP) specification.

Copyright 2004 - Matt Baxter - mjb@fatpipe.org

Aktuální přehled definovaných typů ICMP zpráv dostupný na adrese
<http://www.iana.org/assignments/icmp-parameters>

Internet Control Message Protocol (ICMP) – příklady zpráv

- oznamy o chybách:
 - *Destination unreachable* – „Destination“ může být protokol, port, uzel nebo celá síť
 - *Time exceeded* – informace o vypršení TTL či informace o vypršení času pro znovusložení fragmentů IP datagramu
- dotazy na stav sítě/uzlu:
 - *Echo request/reply* – požadavek na odpověď
- zprávy obsahují část paketu, který
 - způsobil chybu
 - na který se váže odpověď
- přímé využití ICMP v aplikacích:
 - program ping – využití ICMP Echo request/reply
 - program traceroute – využití ICMP Time exceeded (TTL expired)

Internet Control Message Protocol (ICMP) – omezení

- ochrana proti rekurzivnímu generování:
 - Chybový ICMP paket *není generován* jako reakce na:
 - ICMP chybu
 - broadcast nebo multicast zprávu
 - poškozenou IP hlavičku (špatná cílová adresa)
 - chybu fragmentu (kromě prvního)
- generování ICMP zpráv často výkonnostně omezeno

IP protokol verze 6 (IPv6) – Proč nový protokol?

- *hlavní impulz pro návrh nového IP protokolu*: relativně rychlé vyčerpávání adresního prostoru IPv4 protokolu
 - (počátek 90. let 20. století)
- další důvody: problémy IPv4, které vyvstaly s rozvojem Internetu, zejména
 - slabá podpora přenosů aplikací reálného času
 - žádná podpora zabezpečené komunikace na úrovni IP
 - žádná podpora autokonfigurace zařízení
 - žádná podpora mobility
 - atp.
- (mnoho vlastností do IPv4 zpětně doimplementováno)