Katedra softwarového inženýrství,
Matematicko-fyzikální fakulta,
Univerzita Karlova, Praha

Rodina protokolů TCP/IP,
verze 2.3

Část 2: Architektura TCP/IP

Jiří Peterka, 2006

Motto

Víš-li, jak na to, čtyři vrstvy ti plně postačí ....

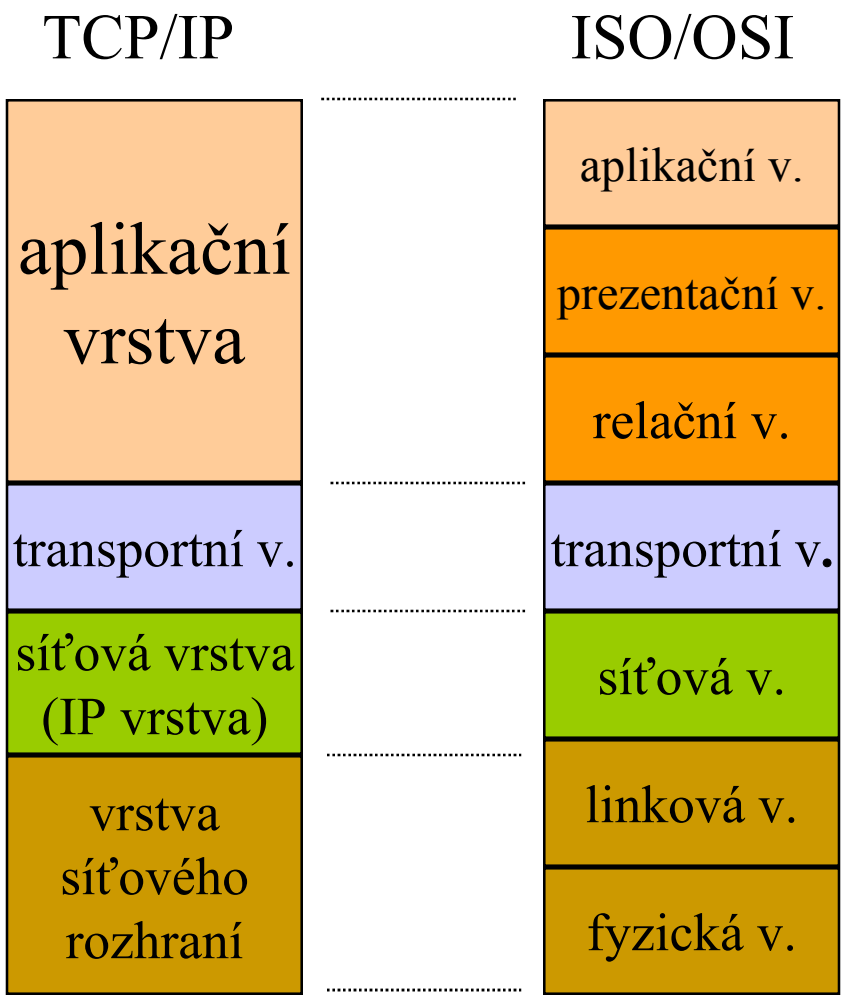
... nevíš-li, ani sedm ti jich nepomůže

V čem se liší TCP/IP a ISO/OSI?

- v celkovém přístupu autorů
  - **ISO/OSI:** všechno musíme vymyslet sami (nebo alespoň převzít to, co vymysleli jiní, a udělat z toho vlastní standard)
    - příklad: ISO vydává Ethernet jako svůj standard ISO 8802.3
  - **TCP/IP:** to co je rozumné převezmeme a využijeme
    - soustředí se na "provázání" vlastních řešení s cizími
    - řeší např. jak provozovat IP nad Ethernetem
- ve způsobu tvorby nových řešení:
  - **ISO/OSI:** od složitého k jednoduššímu
    - řešení vznikají od začátku jako "dokonalá"
      - nejprve navymýšlí vzdušné zámky, pak musí slevovat
    - nejprve vznikne standard, pak se zkoumá praktická realizovatelnost
  - **TCP/IP:** od jednoduššího ke složitějšímu
    - řešení vznikají nejprve jako "skromná", postupně se obohacují
    - nejprve se řešení ověří, a teprve pak vzniká standard

Konkrétně ….

- v pohledu na počet vrstev a způsob jejich fungování
  - jaké služby mají být nabízeny
    - a na jaké úrovni mají být poskytovány
      - kde má být zajišťována spolehlivost
  - jak mají služby fungovat
    - spolehlivost/nespolehlivost, spojovanost/nespojovanost, princip maximální snahy vs. garance kvality služeb, …
  - zda má být ponechána možnost volby
    - mají aplikace právo si vybrat např. mezi spolehlivým a nespolehlivým přenosem?



Pohled do historie

již při budování sítě ARPANET se uplatnila některá koncepční rozhodnutí, která "vydržela" až do dnešního Internetu a TCP/IP

- nedělat žádný centrální prvek (uzel, ústřednu, ...)
  - protože nepřítel by jej odstřelil jako první
- *dnešní Internet stále nemá žádný centrální prvek (řídící centrum, …)*
  - *platí to jak pro technické fungování, tak i pro řízení*

- předem počítat s výpadky a s nespolehlivostí
  - jako kdyby kterákoli část již byla odstřelena či poškozena
- *důsledek: ARPANET (i Internet) jsou řešeny velmi robustně*
  - *mají velmi dobrou schopnost adaptability, dokáží se přizpůsobit podmínkám, …*

"vydržel" i samotný princip paketového přenosu (packet switching), namísto přepojování okruhů (circuit switching).

Důsledky

- obliba nespojované (connectionless) komunikace
  - přenosové mechanismy fungují na nespojovaném principu, teprve vyšší vrstvy mohou fungovat spojovaně, resp. komunikovat se svými protějšky na spojovaném principu
- obliba nespolehlivého přenosu
  - teze: přenosové mechanismy se mají starat o přenos a dělat jej co nejefektivněji
    - nemají se ohlížet na event. ztráty dat
- fungování na principu maximální snahy, ale nezaručeného výsledku
  - přenosové mechanismy se maximálně snaží, ale když se jim něco objektivně nedaří, mají právo se na to "vykašlat"

Nespojovaná komunikace

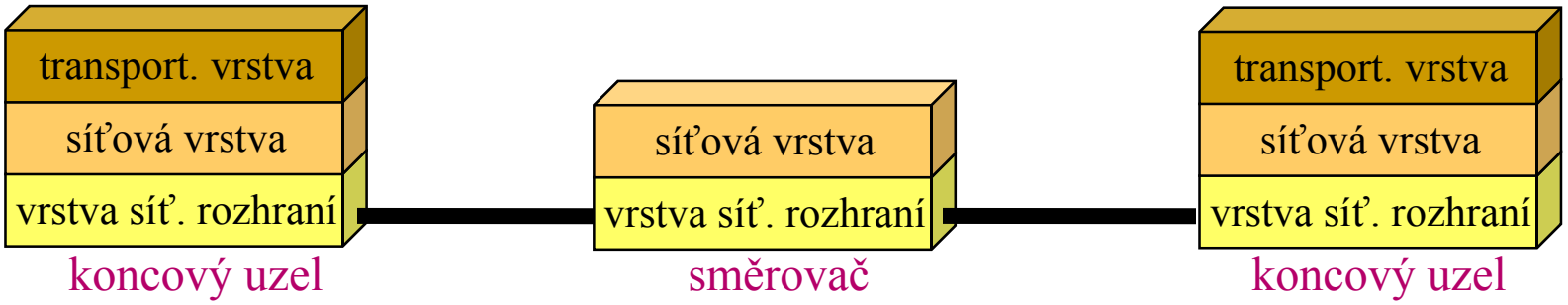
- přenosové služby TCP/IP fungují na nespojovaném principu
  - nenavazují spojení, posílají data v dobré víře že příjemce existuje a bude ochoten je přijmout
  - hlavní přenosový protokol síťové vrstvy (protokol IP) je nespojovaný
- výhody:
  - je to bezestavové
    - nemění se stav odesilatele ani příjemce
  - není nutné složitě reagovat na změny v přenosové infrastruktuře, rušením a novým navazováním spojení
    - vše zajistí adaptivní mechanismy směrování
- výhody/nevýhody:
  - je to výhodné pro "řídké" přenosy
    - přenosy menších objemů dat, hodně rozložené v čase
  - není to výhodné pro "intenzivní" přenosy
    - přenosy větších objemů dat v krátkém časovém intervalu
- vyšší vrstvy mohou fungovat spojitě
  - týká se to především jejich komunikace, ne samotného přenosu (na úrovní síťové vrstvy)
  - transportní protokol (TCP) vytváří iluzi spojovaného přenosu

Nespolehlivá komunikace

- spolehlivost přenosu
  - není nikdy absolutní (100%), je vždy pouze relativní (např. 99%)
    - důvod: již samotné mechanismy detekce chyb nejsou 100%
    - někomu může konkrétní míra spolehlivosti stačit, jinému ne
  - zajištění spolehlivosti je vždy spojeno s nenulovou režií
    - spotřebovává to výpočetní kapacitu, přenosovou kapacitu, …
    - pokud by spolehlivost zajišťovalo více vrstev nad sebou, režie se sčítá (násobí)
      - není to rozumné
      - TCP/IP to nechce dělat, ISO/OSI to dělá
  - způsobuje nerovnoměrnosti (nepravidelnosti) v doručování dat
    - tím, že se opakuje přenos chybně přenesených dat
    - vadí to hlavně u multimediálních přenosů
- zvolené řešení v rámci TCP/IP:
  - spolehlivost není nikomu vnucována
    - každá aplikace si může vybrat:
      - zda vystačí s nespolehlivým přenosem, event. si spolehlivost zajistí sama
      - nebo zda využije spolehlivost kterou nabízí spolehlivý transportní protokol

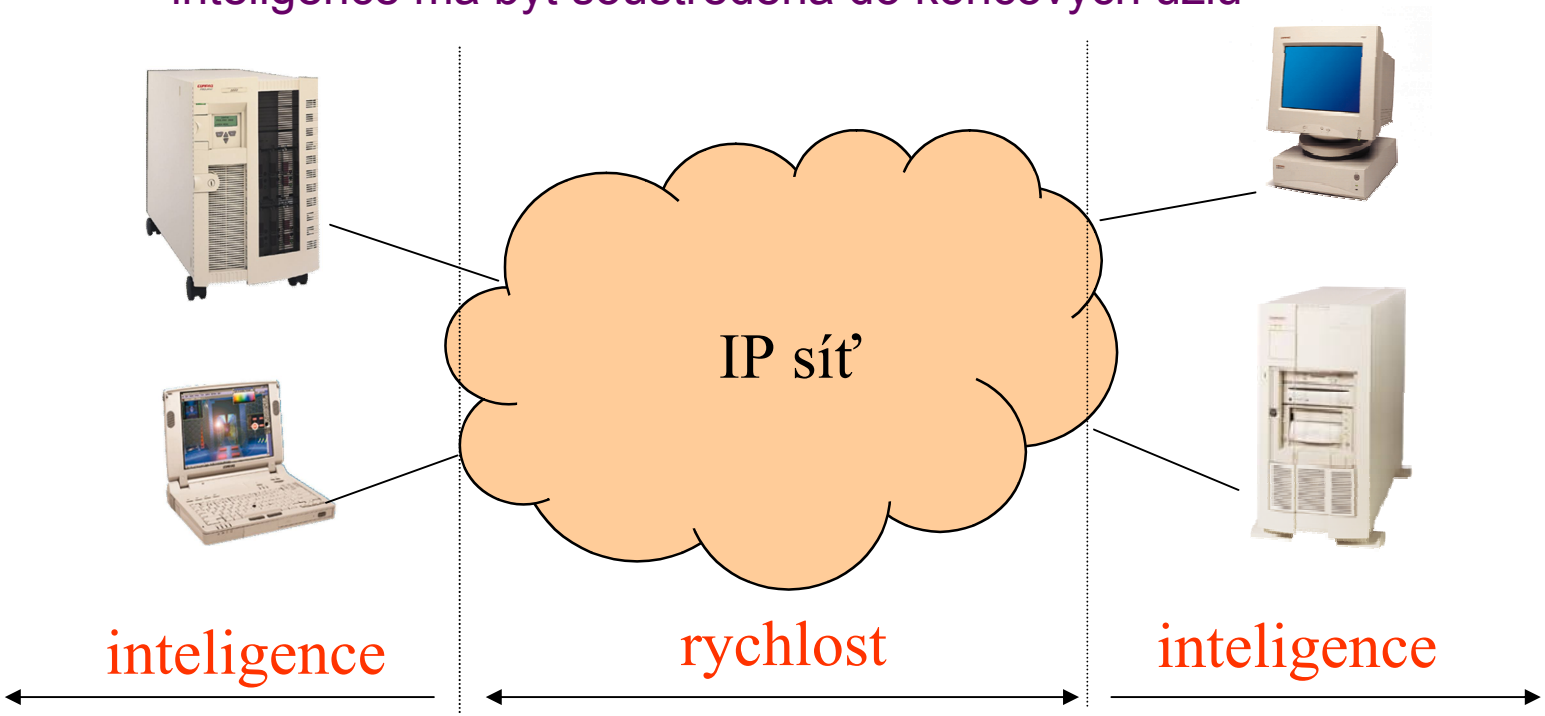
Jiný pohled na spolehlivost

- spolehlivost je o tom, kde v síti má být umístěna "inteligence"
  - výpočetní kapacita, logika implementující zajištění spolehlivosti
    - připomenutí: síťová vrstva je ještě ve všech uzlech, transportní již jen v koncových uzlech
- **ISO/OSI:**
  - inteligence má být v síti
    - spolehlivost musí být řešena na úrovni síťové vrstvy
      - inteligence je ve směrovačích
      - je to drahé a nepružné
      - nedává to možnost výběru
- **TCP/IP:**
  - inteligence má být v koncových uzlech
    - spolehlivost je řešena až v transportní vrstvě
      - je to lacinější, pružnější
    - umožňuje to, aby si aplikace vybíraly zda spolehlivost chtějí či nechtějí
- **teze** (TCP/IP):
  - ***přenosová vrstva se má starat o přenos dat***
    - ***má to dělat co nejefektivněji***
    - ***nemá se rozptylovat dalšími úkoly (např. zajišťováním spolehlivosti, když to si snáze a lépe zajistí koncové uzly sítě)***
  - ***zajištění spolehlivosti je úkolem koncových uzlů***



Hloupá síť vs. chytré uzly

- jiná interpretace:
  - přenosová část sítě (IP síť) má být "hloupá"
    - ale efektivní, má co nejrychleji a nejefektivněji plnit své základní úkoly
  - "chytré" mají být koncové uzly
    - inteligence má být soustředěna do koncových uzlů



Princip maximální snahy

- anglicky"best effort"
  - přenosová část sítě se maximálně snaží vyhovět všem požadavkům, které jsou na ni kladeny
  - pokud se jí to nedaří, má právo krátit požadavky (limitovat, ignorovat je, nevyhovět jim, …)
    - např. pozdržet přenášené pakety do doby, než je bude moci zpracovat
    - může i zahazovat pakety, které vůbec nedokáže zpracovat
  - dělá to rovnoměrně vůči všem požadavkům
    - "měří všem stejně", nepracuje s prioritami
- je to celková filosofie TCP/IP
  - je praktickým důsledkem použití paketového přenosu a přístupu ke spolehlivosti
- alternativa:
  - garance služeb (QoS, Quality of Service)
    - QoS nabízí telekomunikační sítě
- výhoda:
  - sítě fungující na principu "best effort" jsou mnohem efektivnější (i ekonomicky) než sítě nabízející QoS
    - kdyby Internet poskytoval QoS, byl by mnohem dražší než dnes a méně rozvinutý
- nevýhoda:
  - vadí to multimediálním přenosům

Pohled do historie II.

- koncepce protokolů TCP/IP vznikala v době, kdy se rodil Internet
  - když se na tehdejší zárodečný ARPANET nabalovaly další sítě
  - byly to sítě, které často fungovaly na jiné technologické platformě
    - používaly různé přenosové technologie, různé adresy, různé přístupy ke spolehlivosti, ke spojovanosti, různé velikosti rámců atd.
- cíl TCP/IP:
  - umožnit plnohodnotné připojení jakýchkoli sítí
  - fakticky: **důraz na internetworking**
- konkrétně:
  - šlo o to, aby protokoly TCP/IP mohly být používány nad nejrůznějšími přenosovými technologiemi
  - týká se to hlavně protokolu IP
    - výsledek: podařilo se, dnes je možné provozovat IP nad čímkoli
      - **"IP over everything"**
    - bylo třeba přijmout mnoho koncepčních rozhodnutí, které s tím byly spojeny
      - způsob propojení dílčích sítí
      - adresování
      - "viditelnost" specifických vlastností přenosových technologií
      - vazbu na fyzické (linkové) přenosové technologie
      - ….

Příčiny úspěchu TCP/IP

- vlastní systém adresování
  - zabudovaný do mechanismů fungování
  - umožňuje identifikovat a adresovat uzly (zařízení) i dílčí entity (služby atd.) bez znalosti detailů jejich připojení
    - IP a DNS, nově ENUM …
    - součástí je systém celosvětové koordinace
      - přidělování IP adres, stromová struktura DNS
  - systém adresování se dokázal uzpůsobit stále většímu rozsahu sítí
    - výjimka: rozsah adres IPv4, vznik IPv6
- dobrá škálovatelnost
  - původní řešení vzniklo pro sítě s desítkami uzlů
  - dnes funguje pro Internet s miliony uzlů
    - v zásadě beze změny
    - výjimka: IPv4 – IPv6
  - základní rozhodnutí vznikla před 30 lety
    - a dodnes se nemusela měnit
  - změny v TCP/IP byly spíše "inkrementálního" charakteru
    - něco se přidalo

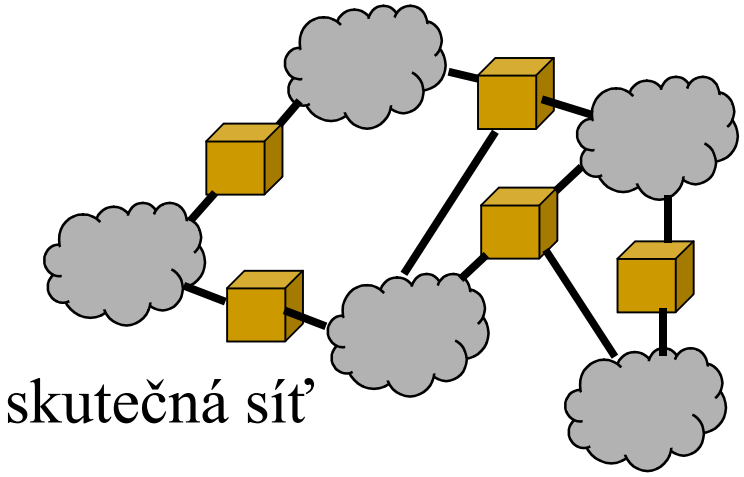
Příčiny úspěchu TCP/IP

- podpora internetworkingu a směrování
  - protokoly TCP/IP vychází dobře vstříc vzájemnému propojování sítí
  - obsahují další protokoly pro podporu práce směrovačů
    - ICMP, Interior a Exterior Routing Protocols
- nezávislost na fyzických (linkových) technologiích
  - protokoly TCP/IP (hlavně IP) dokáží běžet nad každou přenosovou technologií nižších vrstev
    - "**IP over everything**"
- univerzálnost, dobrá podpora aplikací
  - lze využít pro všechny aplikace
    - byť některým s evychází vstříc méně
  - aplikacím není vnucováno, co a jak mají používat
    - volba TCP vs. UDP
  - výsledek:
    - "**Everything over IP**"
- otevřený, neproprietární charakter
  - standardy jsou otevřené a přístupné každému
  - proces vzniku standardů je otevřený

Koncepce TCP/IP: katenetový model

- TCP/IP předpokládá že "svět" (internetwork, internet) je:
  - tvořen soustavou dílčích sítí
    - chápaných jako celky na úrovni síťové vrstvy, tzv. IP sítí
  - dílčí sítě jsou vzájemně propojeny na úrovni síťové vrstvy
    - pomocí směrovačů (dříve nazývaných IP Gateways, dnes: IP Routers)
  - toto propojení může být libovolné
    - může být stylem "každý s každým", nebo "do řetězce" apod.
      - jedinou podmínkou je souvislost grafu
      - "katenet" je "řetězec" – ten je jakousi minimální podmínkou pro souvislost celé soustavy sítí
    - možné je i redundantní propojení

představa katenetu



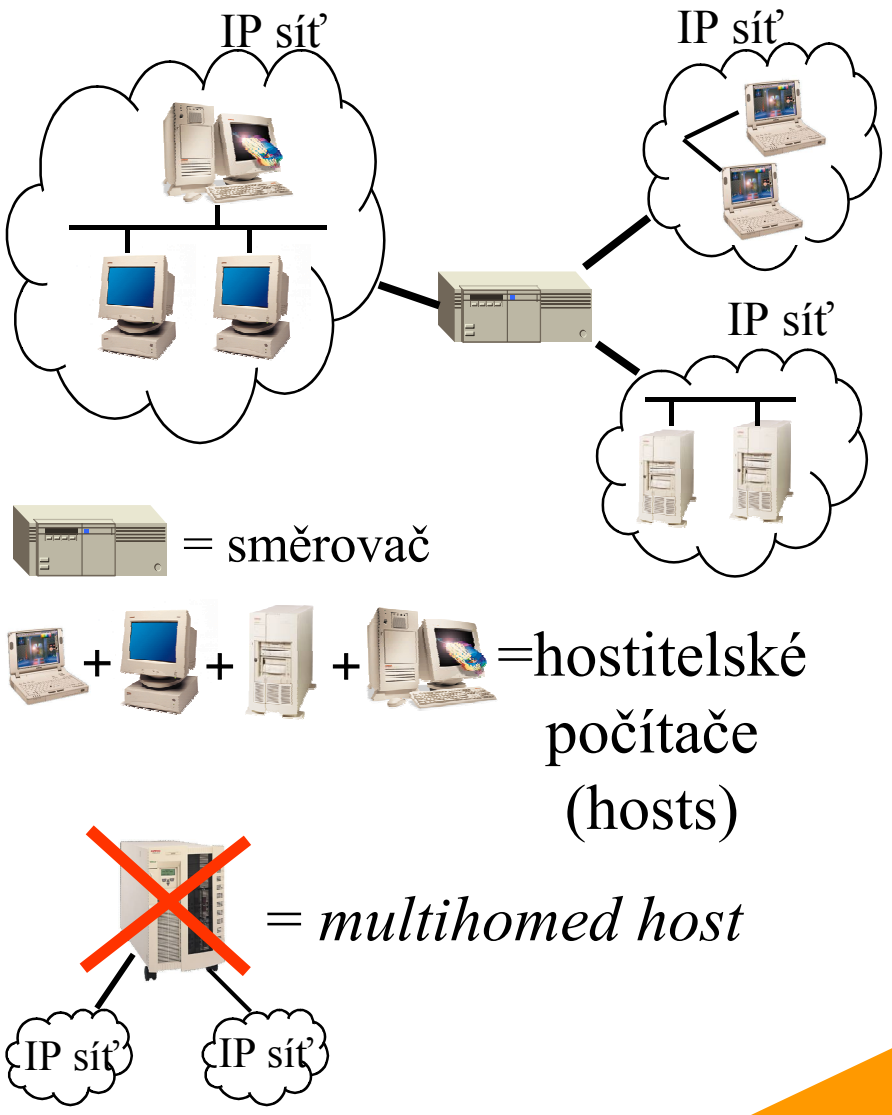
skutečná síť

= IP síť

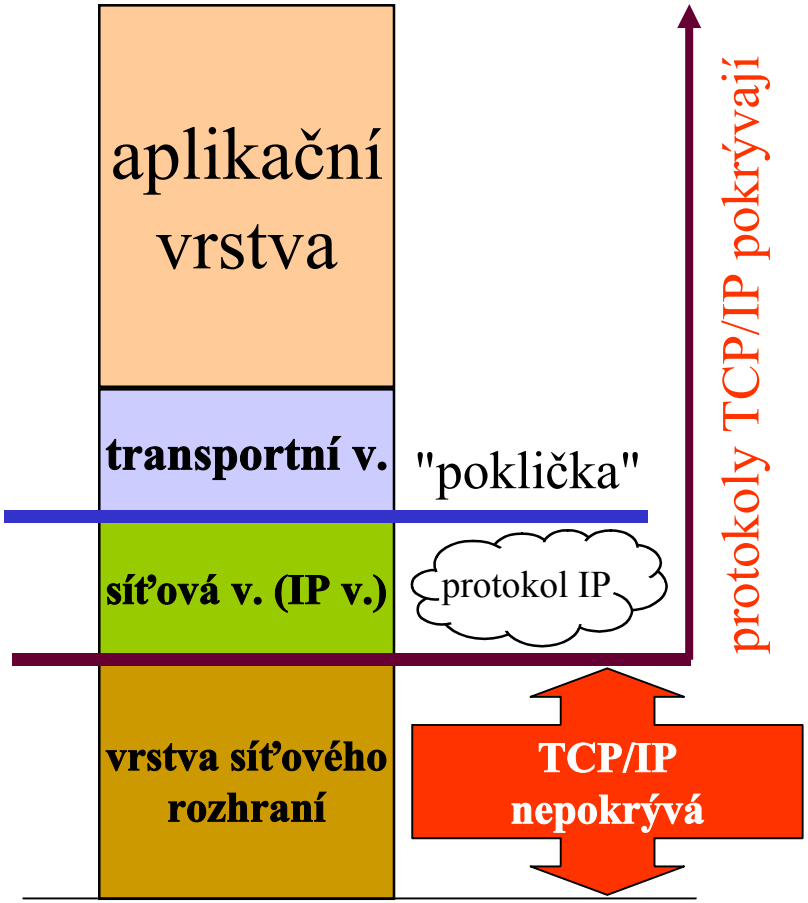
= IP Router

Hostitelské počítače vs. směrovače

- TCP/IP předpokládá, dva typy uzlů v síti:
  - hostitelské počítače (host computers)
    - tj. koncové uzly, např. servery, pracovní stanice, PC, různá zařízení (tiskárny, …)
    - jsou připojeny jen do jedné IP sítě (mají jen jednu síťovou adresu)
  - směrovače (IP Routers, dříve nesprávně IP Gateways)
    - jsou připojeny nejméně do dvou IP sítí
    - zajišťují "přestup" (směrování)
- teze:
  - oba typy uzlů by se neměly prolínat
    - směrovače by neměly plnit další funkce
    - hostitelské počítače by neměly fungovat jako směrovače
      - v podobě tzv. multihomed-hosts, kdy jsou připojeny do více sítí současně



Vrstva síťového rozhraní



- TCP/IP se nezabývá tím, co je pod úrovní síťové vrstvy
  - přesněji: sám nedefinuje protokoly které fungují "pod" síťovou vrstvou (na úrovni vrstvy síťového rozhraní)
    - jde např. o Ethernet, ATM, Token Ring, FDDI, Frame Relay …
  - zaměřuje se pouze na to, jak propojit síťovou vrstvu s vrstvou síťového rozhraní
    - např. jak provozovat IP nad Ethernetem, nad ATM ...
- výjimka: protokoly SLIP a PPP
  - definují způsob přenosu po dvoubodových spojích
    - zasahují až do vrstvy síťového rozhraní
- důsledek:
  - **nezávislost na fyzické (linkové) přenosové technologii**

Síťová vrstva: dilema pokličky

Autoři TCP/IP se museli rozhodnout, zda:

– vytvoří jednotnou nadstavbu nad soustavou vzájemně propojených sítí
  - přenosový protokol na úrovni síťové vrstvy (IP protokol), který bude mít všude stejné vlastnosti a poskytovat stejné služby
  - stejné adresování
  - ….

ano

vyšší vrstvy mohou být jednotné, nemusí se zabývat odlišnostmi

ne

– nebo zda nadstavba nebude všude stejná
  - tj. protokol IP bude mít v různých sítích různé vlastnosti, resp. nabízet různé služby
  - ….

ne

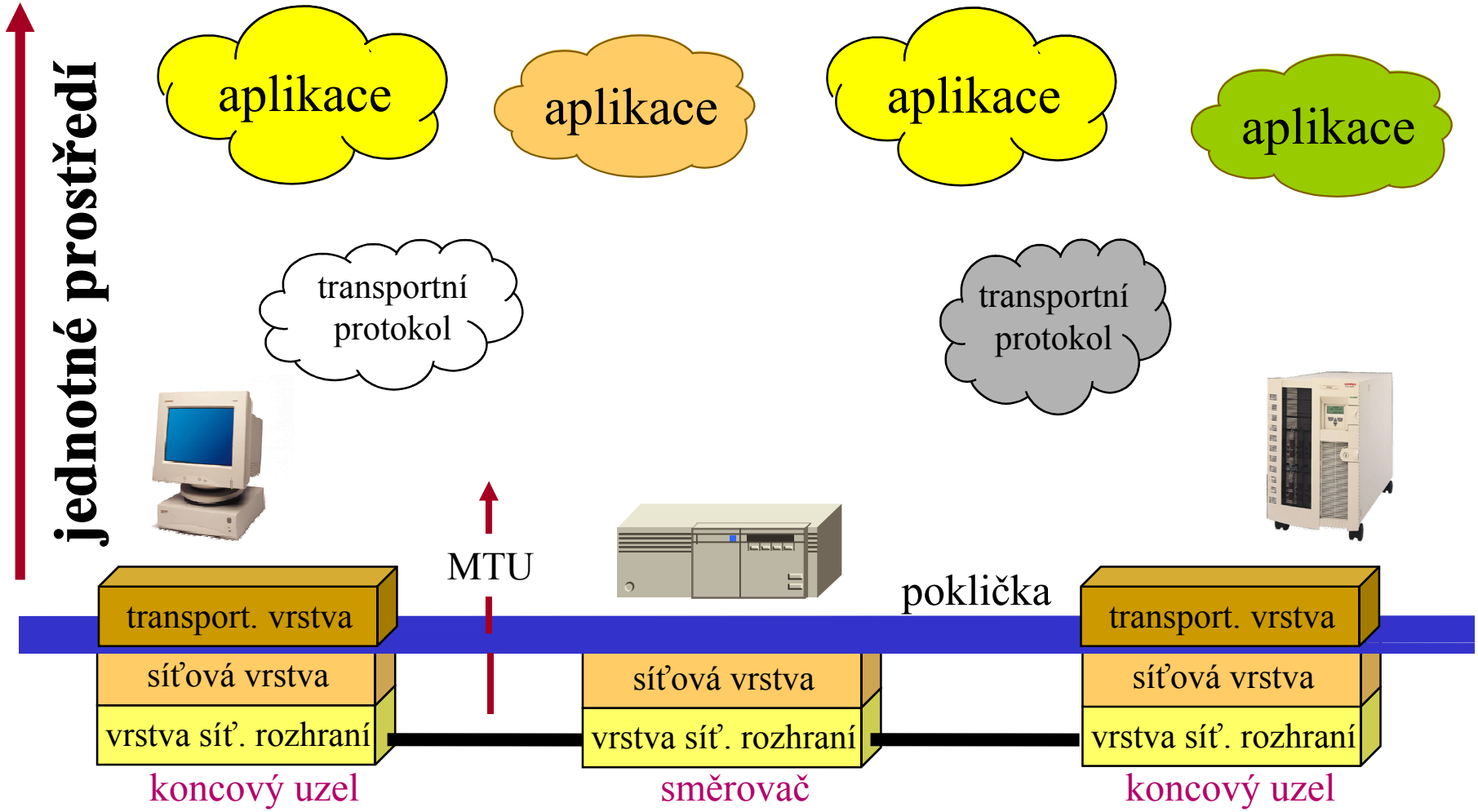
ano

umožňuje to dosahovat maximální možné efektivnosti, přizpůsobením se specifickým vlastnostem přenosových mechanismů

Výsledek – koncepce síťové vrstvy

- autoři TCP/IP se rozhodli pro "jednotnou pokličku", která zastírá konkrétní specifika jednotlivých IP sítí
- fakticky jde o jednotnou nadstavbu, kterou tvoří:
  - **přenosový protokol IP**, který má všude stejné vlastnosti a všude poskytuje stejné služby
    - je **nespojovaný, nespolehlivý, funguje na principu maximální snahy**
  - jednotné adresování
    - virtuální 32-bitové adresy (nemají žádný reálný vzor), tzv. IP adresy
      - tyto adresy by měly vyhovovat "pohledu na svět", který má TCP/IP – že svět je tvořen dílčími sítěmi a hostitelskými počítači (a směrovači)
      - IP adresy mají "síťovou část", identifikující síť jako celek, a dále "uzlovou část", identifikující uzel v rámci sítě
- existuje ale jedna výjimka:
  - IP protokol i vyšší vrstvy "vidí" maximální velikost linkového rámce (skrz parametr MTU, Maximum Transfer Unit) a měli by jej respektovat
    - tak aby nedocházelo ke zbytečné fragmentaci při přenosech

Představa pokličky



Součásti síťové vrstvy

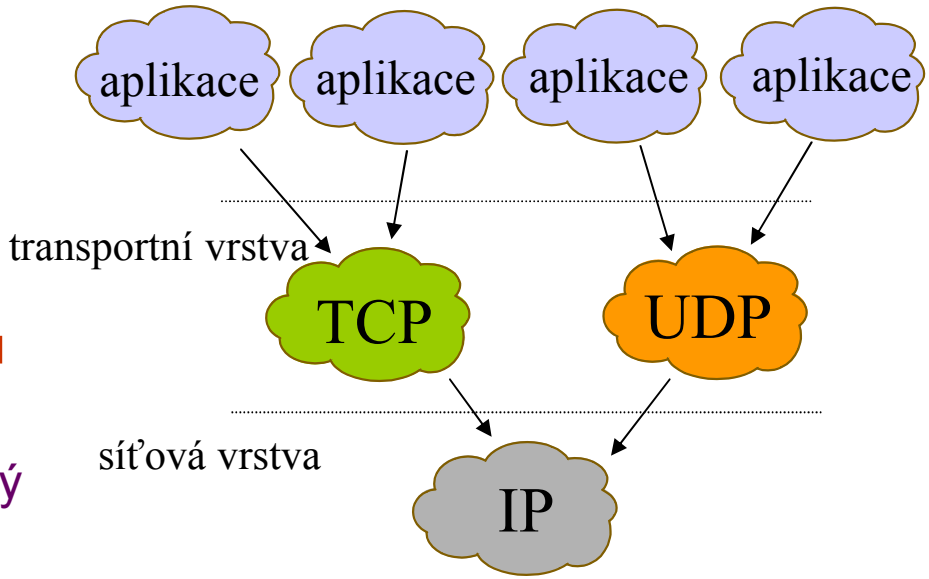
- v síťové vrstvě jsou "zabudovány":
  - síťové adresy
    - 32-bitové abstraktní adresy
      - nevychází z linkových adres
  - převodní mechanismy, které překládají mezi fyzickými (linkovými) adresami a virtuálními IP adresami
    - protokoly ARP, RARP, ….
  - mechanismy fragmentace
    - vazba na MTU
  - protokoly na podporu fungování síťové vrstvy
    - protokol ICMP
      - "posel špatných zpráv"
      - zajišťuje informování o nestandardních situacích
- se síťovou vrstvou úzce souvisí:
  - protokoly podporující směrování a výměnu aktualizačních informací o stavu sítě
    - RIP, OSPF, IGP, EGP, …
  - mechanismy přidělování IP adres
  - mechanismy překladu mezi symbolickými doménovými jmény a IP adresami
- do síťové vrstvy byly nově přidány také
  - mechanismy překladu adres
    - NAT
  - koncept privátních IP adres
  - mechanismy dělení adres a sdružování adres
    - subnetting, supernetting, CIDR
  - bezpečnostní mechanismy
    - IPSec
  - podpora mobility
    - Mobile IP

Koncepce transportní vrstvy

- realizuje "end-to-end" komunikaci
  - nabízí dva transportní protokoly
- TCP (Transmission Control Protocol)
  - funguje spojovaně
    - vyžaduje navázání/ukončení spojení ..
  - od aplikace přebírá data po bytech
    - jako "bytový proud"
    - ale sám data přenáší po blocích, jako tzv. TCP segmenty
  - funguje spolehlivě
    - zajišťuje spolehlivý přenos
      - používá kontinuální potvrzování a selektivní opakování
  - je velmi adaptivní
    - dokáže se průběžně přizpůsobovat různým podmínkám přenosu
      - přenosové zpoždění, rozptyl zpoždění atd.
  - je velmi komplikovaný
    - velký a složitý kód, …
- UDP (User Datagram Protocol)
  - je pouze jednoduchou nadstavbou nad síťovým protokolem IP
    - jeho kód je malý a jednoduchý
  - funguje nespojovaně
    - nenavazuje spojení
  - funguje nespolehlivě
  - od aplikace přebírá data po blocích
    - a vkládá je do svých "datagramů"
      - UDP datagramů, User datagramů
- transportní vrstva zajišťuje multiplex/demultiplex
  - adresuje entity v rámci jednotlivých uzlů
  - pomocí čísel portů

Koncepce transportní vrstvy

- "realizuje demokracii":
  - přenosové mechanismy do úrovně síťové vrstvy fungují nespolehlivě
  - na úrovni transportní vrstvy jsou dva alternativní protokoly
    - UDP, nespojovaný, nespolehlivý
    - TCP, spojovaný, spolehlivý
  - **aplikace si mohou samy vybrat, zda budou používat TCP nebo UDP**



SMTP SNMP TFTP BOOTP DHCP

RPC rlogin FTP Telnet HTTP DNS RPC NFS XDR

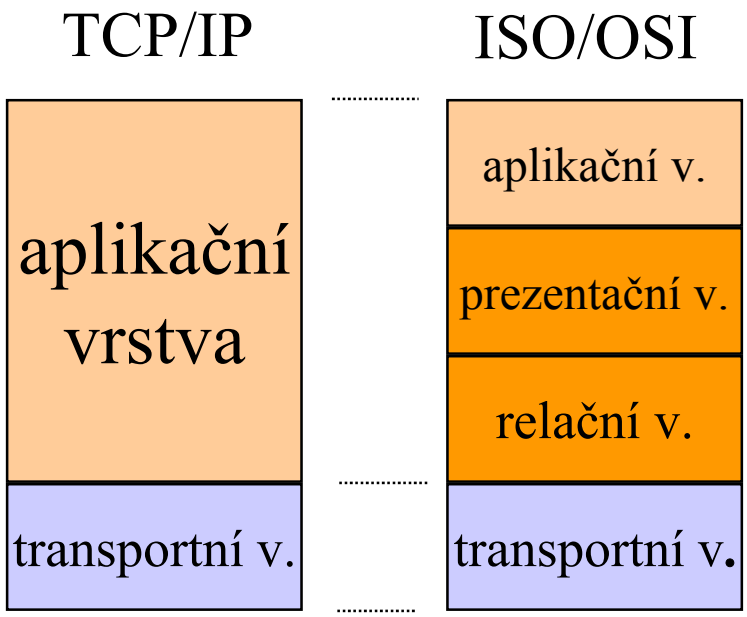
aplikační vrstva

TCP **UDP**

transportní vrstva

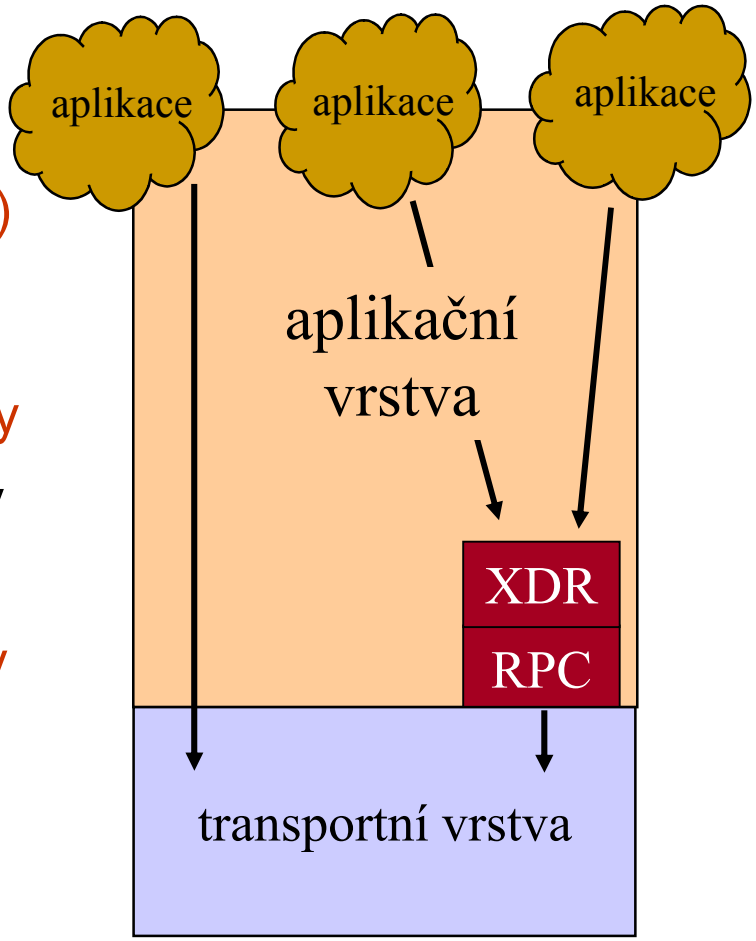
Prezentační a relační služby v TCP/IP

- ISO/OSI má samostatnou prezentační a relační vrstvu
  - vychází z předpokladu že prezentační a relační služby budou potřebovat všechny aplikace
    - pak mají samostatné vrstvy smyl
- TCP/IP nemá samostatné vrstvy
  - vychází z předpokladu, že prezentační a relační služby budou potřebovat jen některé aplikace
    - pak nemá smysl dělat samostatné vrstvy
    - aplikace, které tyto služby potřebují, si je musí realizovat samy



Výjimka: RPC a XDR

- aplikační protokol NFS používá ke svému fungování prezentační a relační služby
  - protokol RPC (Remote Procedure Call) pro relační služby)
  - protokol XDR (eXternal Data Representation) pro prezentační služby
- tyto protokoly jsou implementovány jako vícenásobně využitelné
  - jako samostatné moduly, jejichž služby může využívat každá aplikace která chce
    - a naopak nemusí ta aplikace, která nechce (a v tom případě nenese jejich režii !!!!)

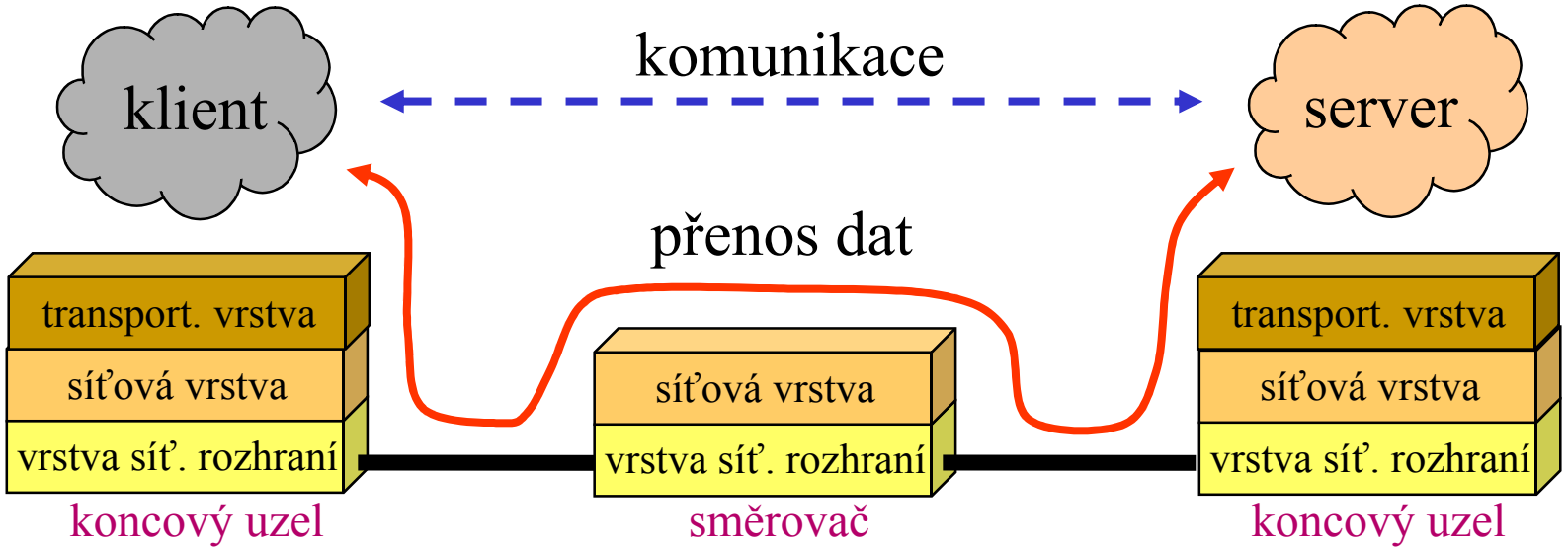


Aplikace v TCP/IP

- původně:
  - elektronická pošta (SMTP, RFC 822)
  - přenos souborů (FTP)
  - vzdálené přihlašování(TELNET, rlogin)
    - těmto aplikacím dobře vyhovovalo fungování sítě "na principu maximální snahy, ale nezaručeného výsledku"
- později se objevily a prosadily nové aplikace:
  - news
  - sdílení souborů (NFS)
  - WWW (HTML, HTTP, ….)
  - on-line komunikace (chat, IRC, ICQ, messengery, …)
    - princip maximální snahy je pro ně stále ještě akceptovatelný
      - byť ne ideální
- později se objevují "multimediální" aplikace
  - "audio over IP"
    - rozhlasové vysílání
  - VOIP
    - Voice over IP, IP telefonie
  - IPTV
    - TV over IP, TV na žádost po IP
  - pro tyto aplikace princip "maximální snahy" není optimální, ale ještě postačuje, důležitá je hlavně disponibilní přenosová kapacita
- dochází k "platformizaci" aplikací
  - původně samostatné aplikace se přesouvají do role nadstavby na platformě jiné aplikace
    - nejčastěji WWW

Aplikace v TCP/IP

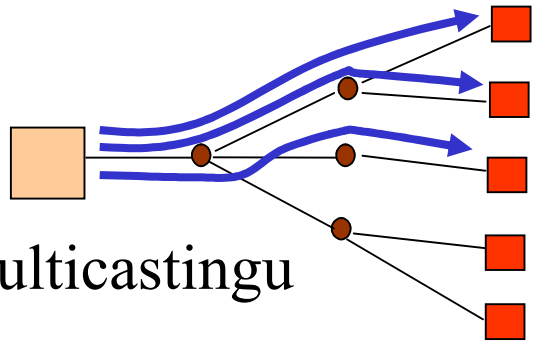
- prakticky všechny aplikace v rámci TCP/IP jsou založeny na architektuře client/server
  - servery poskytující "veřejné" služby jsou dostupné na tzv. dobře známých portech (well-known ports)
  - přenosové mechanismy TCP/IP jsou uzpůsobeny komunikaci stylem 1:1 (mezi 1 serverem a 1 klientem)



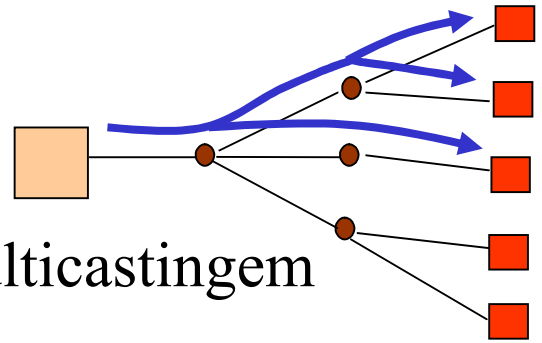
Problém distribučních aplikací

- s postupem času se objevily i takové aplikace, pro které je fungování přenosových mechanismů TCP/IP principiálně nevhodné
- "distribuční služby" = videokonference, vysílání rozhlasu a TV, ….
  - potřebují dopravovat stejná data od 1 zdroje k více příjemcům současně
    - tzv. **multicasting** (event. broadcasting)
  - přenosové mechanismy TCP/IP to neumí !!!
    - přenosové mechanismy počítají s přenosem 1:1 (od jednoho zdroje k jednomu příjemci)
    - pokus: služba MBONE (nepříliš úspěšná)
    - řeší se až v rámci IPv6 a IP Multicast Initiative

netýká se VOIP a IPTV, to jsou služby s přenosy typu 1:1



bez multicastingu



s multicastingem

Problém multimediálních aplikací

- potřebují dostávat svá data:
  - s malým zpožděním
  - s pravidelným zpožděním
    - s pravidelnými odstupy mezi sebou
- týká se to například přenosu živého obrazu či zvuku
  - aplikace VOIP, TV vysílání, rozhlas, video-on-demand
- problém je s fungováním přenosových mechanismů TCP/IP na principu "*maximální snahy, ale nezaručeného výsledku*"
  - byla by zapotřebí podpora QoS (kvality služeb)
    - QoS je v zásadě "protipólem" principu maximální snahy

- možná řešení:
  - "**kvantitativní**": zvyšování disponibilní kapacity
    - fungování na principu "maximální snahy …" zůstává
    - zlepšení je statistické
      - je menší pravděpodobnost, že bude muset dojít ke krácení požadavků
    - týká se:
      - přenosových kapacit (tj. linek)
      - "přepojovacích kapacit" (směrovačů, switchů)
  - "**kvalitativní**": zavedení podpory QoS
    - fungování na principu "maximální snahy …" je nahrazeno jiným způsobem fungování
    - zlepšení je garantované
      - ale drahé a obtížné

QoS v TCP/IP – možné přístupy

- prioritizace
  - různým druhům přenosů se přiřadí různé priority a je s nimi nakládáno odlišně
    - přenosy s vyšší prioritou dostávají "kvalitnější obsluhu" (a příděl zdrojů) na úkor přenosů s nižší prioritou
  - příklady řešení:
    - DiffServ
      - Differentiated Services
    - MPLS
      - MultiProtocol Label Switching
- rezervace
  - pro potřebu konkrétních přenosů si lze vyhradit (rezervovat) požadované zdroje a ty pak využívat
    - týká se i vyhrazení přenosové kapacity, přepojovací kapacity atd.
  - příklady řešení:
    - IntServ (Integrated Services)
    - RSVP (ReSource reserVation Protocol)
- "hrubá síla"
  - princip "best effort" se nemění, pouze se předimenzují dostupné kapacity
    - tak aby nedocházelo ke kapacitním problémům – tak často

Problém bezpečnosti

- přenosové mechanismy TCP/IP neposkytují žádné zabezpečení
  - nebylo to "v původním zadání"
    - ARPANET (budoucí Internet) byl tehdy spíše privátní sítí, jeho uživatelé byli "hlídáni" jinak
    - uživatelé byli "dobře známí"
    - spíše se aplikovala "fyzická bezpečnost"
      - ochrana budov, zařízení atd.
  - přenášená data nejsou žádným způsobem chráněna proti "odposlechu"
    - nejsou šifrována ani jinak kódována či chráněna
    - chybí tzv. důvěrnost
  - nejsou ani chráněna proti ztrátě či změně
    - při nespolehlivému přenosu
    - chybí tzv. integrita
- předpoklad:
  - pokud nějaká aplikace potřebuje určitou míru zabezpečení, musí si ji zajistit sama
- jde o stejný "kompromis" jako u spolehlivosti:
  - buďto poskytnout zabezpečení všem (i těm kteří jej nepotřebují), nebo si jej bude muset každý zájemce udělat sám
    - teze: přenosové mechanismy by měly hlavně přenášet data, ne se starat o další funkce …
- důsledek:
  - přenosová infrastruktura je jednodušší, rychlejší a také lacinější
    - oproti stavu, kdy by fungovala zabezpečeným způsobem
- praxe:
  - zabezpečení se řeší na aplikační úrovni
- IPSEC:
  - časem byl vypracován celý framework (rámec) pro zajištění bezpečnosti ještě na úrovni síťové vrstvy

IP Security (IPSec)

- je to celý rámec (framework)
  - nejde o (jeden) konkrétní protokol
    - ale o soustavu vzájemně provázaných opatření a dílčích protokolů
  - nejde o jeden internetový standard
    - je definován několika RFC
  - funguje na síťové úrovni!!!
- IPSec původně vznikl pro IPv6
  - ale začal se používat i pro stávající IPv4
- IPSec zajišťuje:
  - důvěrnost
    - šifruje přenášená data
  - integritu
    - že přenášená data nejsou při přenosu změněna
- umožňuje:
  - aby si komunikující strany dohodly algoritmy a klíče pro zabezpečení svých přenosů
- chrání
  - i proti některým druhům útoků
    - např. "replay attack"
- má dva režimy fungování:
  - transport mode
    - "zabezpečovací údaje" se vloží přímo do IP datagram
      - do jeho hlavičky a za ni
  - tunnel mode
    - IP datagram se vloží do jiného (zabezpečeného) datagramu

IP verze 6

- 197x:
  - rozhodnutí o 32-bitových IP adresách, IPv4
  - tehdejší představa:
    - ARPANET může mít až tisíce uzlů
    - dnes: Internet má milionu uzlů
- 198x/9x:
  - začíná hrozit nebezpečí vyčerpání 32-bitového adresového prostoru
  - IAB začíná problém řešit
    - vzniká samostatná oblast (area) v rámci IETF
- dočasná řešení - usilují zpomalit úbytek IP adres
  - přísnější způsob přidělování IP adres
  - subnetting
  - privátní IP adresy
  - mechanismus CIDR
- dočasná řešen významně uspěla v oddálení problému
  - nebezpečí vyčerpání se stalo méně akutní
- současně se začalo pracovat na "definitivním" řešení
  - zjištění: 32-bitové adresy jsou v protokolu IP tak hluboce "zakořeněny", že není zvětši adresový prostor
    - ještě v rámci téhož IP (IPv4)
  - je nutné vyvinout zcela nový protokol IP !!!
    - s větším adresovým prostorem, ale i dalšími změnami
- dnes používaný protokol IP je verze 4
  - IPv4
- nový protokol IP je verze 6
  - IPv6
    - IPv5 neexistuje
- IPnG (IP – The Next Generation)
  - obecné označení pro všechny návrhy, které se sešly v rámci IETF při hledání nové verze
    - IPv6 je jeden z protokolů IPnG
    - někdy se bere IPv6=IPnG

Koncepce IPv6

- používá adresy v rozsahu 128 bitů
  - celkem 340'282'366'920'938'463'463'374'607'431'768'211'456 unikátních IPv6 adres
    - každý dnes žijící člověk by mohl dostat na 4 miliardy adres,
    - každé zrnko písku na plážích světa by mohlo dostat na 2128 různých adres
    - na každý čtvereční mikrometr zemského povrchu by připadlo na 5000 adres IPv6
    - ……
  - 128- bitový adresový prostor je hierarchicky členěn
    - multicast adresy, lokální adresy pro sítě a segmenty, pro ISP, "IPv4 embedded"….
- má nový formát IP datagramu
  - "IPv6 datagramu"
- nabízí různé strategie přidělování IP adres
  - včetně možnosti, aby si uzel sám určil svou vlastní IP adresu
    - podle toho, jak je naadresováno jeho okolí
- podporuje hierarchické směrování, bezpečnost, kvalitu služeb (QoS), …
  - nemá broadcast, má unicast, multicast a nově také "anycast"

TCP/IP a mobilita

- mobilita
  - IP adresy nejsou "mobilní"
    - nelze je přenášet mezi sítěmi
    - směruje se na základě IP adres, podle jejich síťové části
      - nelze jen tak "vytrhnout" jednotlivé IP adresy z jejich "mateřské" sítě
  - protokol IP vznikal v době, kdy počítače nebyly přenosné, nebyl požadavek na mobilitu
- řešení mobility:
  - přidělení nové IP adresy v nové síti
    - BOOTP, DHCP atd.
  - skrze agenty a tunely
    - "na původním místě" zůstane agent, který vše přeposílá "skrze tunel" tam, kde se uzel právě nachází
  - jinak
- IP Mobility Support
  - "Mobile IP"
    - RFC 2002 a další (3220, 3344)
  - princip fungování:
    - metoda "agentů"
      - pakety jsou směrovány na původní místo, odkud jsou následně přeposílány na nové místo
    - pro vyšší vrstvy je to neviditelné
    - vzdálené zařízení nemusí Mobile IP podporovat
      - vše zařizuje agent, mobilní zařízení o tom neví
  - je to určeno pro "příležitostnou mobilitu"
    - např. pohyb 1x za týden
    - nikoli pro "častou mobilitu", jako např. v mobilních sítích, roaming apod.
  - mobilní zařízení musí mít staticky přiřazenou IP adresu