

## 1. Jaké jsou úkoly transportní vrstvy?

- Zajišťovat end-to-end komunikaci
- rozlišovat mezi více uživateli a příjemci v rámci daného uzlu
- vyrovnávat rozdíly mezi schopnostmi nižších vrstev a požadavky vyšších: měnit spojovaný/nespojovaný charakter, zajišťovat spolehlivost, zajišťovat kvalitu služeb
- předcházet/řídit zahlcování a řídit tok



## 2. Jaká je koncepce transportní vrstvy v RM ISO/OSI?

Prenosová část sítě (do síťové vrstvy) funguje skoro maximalisticky: spojované, spolehlivo.

Počítá s existencí 3 různých kvalitních služeb na úrovni síťové vrstvy:

- kategorie A: žádné ztráty paketů, žádné výpadky spojení
- kategorie B: žádné ztráty, občas výpadky (veřejné datové sítě)
- kategorie C: občas ztráty, občas výpadky (velké sítě)

Transportní vrstva je od toho, aby pokračovala v rovnakém duchu, až dodatečně se do ISO/OSI přesadili i nespolehlivé a nespojované služby.

Definuje 5 tříd transportních protokolů:

- TP0: jednoduchý obal nad síťovými službami kat. A
- TP1: nad B, řeší výpadky
- TP2: nad A, dokáže využít jedno síťové spojení pro více transportních
- TP3: nad B, více transportních spojení nad jedním síťovým
- TP4: transportní služba nad síťovými službami kat. C (nespolehlivá s výpadky), zajišťuje spolehlivost

## 3. Jaká je koncepce transportní vrstvy v TCP/IP?

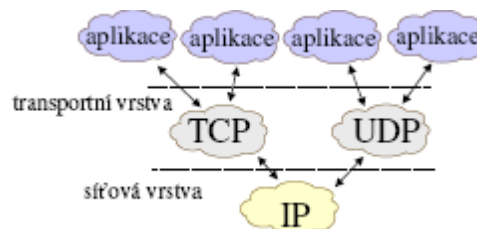
Prenosová část (do síťové vrstvy) funguje minimalisticky: zaměřena na rychlost a robustnost, funguje nespolehlivě a nespojované.

Transportní vrstva je od toho, aby měla charakter služeb síťové vrstvy, ale i když si to vyšší vrstvy želají.

Obsahuje dva alternativní transportní protokoly:

- TCP (Transmission Control Protocol): mění charakter přenosových služeb, zajišťuje spolehlivost a funguje spojované
- UDP (User Datagram Protocol): nemá charakter přenosových služeb, nespolehlivě, nespojované

Vyšší vrstvy si mohou vybrat, který protokol použít.



## 4. K čemu slouží body SAP (RM ISO/OSI) a porty (TCP/IP)?

**SAP (Service Access Point):** přechodový bod.

**Port:** přechodový port.

Problém: jak identifikovat odesílatele/příjemce.

Idea: neidentifikovat přímo entity vyšších vrstev ale přechodové body mezi transportní a vyšší vrstvou.

**Sposob fungovania:** prechodove body (porty) maju staticky charakter (lahsie sa identifikuju), existuju nezávisle na entitych vyssich vrstiev.

### 5. Jaký je vzťah medzi porty a sockety (v TCP/IP)?

Porty su logicka zalezitost, na vsetkych platformach su rovnake.

Aplikacie pracuju s portami cez API, co moze byt sucast operacneho systemu alebo moze mat formu kniznic linkovanych v aplikacii.

Styl prace s portami: prevazuje zavedeny BSD Unixom zalozeny na socketoych.

**Socket:** abstrakcia suboru v BSD pre potreby prace so subormi, styl „open-read-write-close“, boli pouzite i pre potreby sietovania, je to analogia brany veduca k sietovym sluzbam.

### 6. K čemu slouží konvence o době známých portech a jakou má podobu?

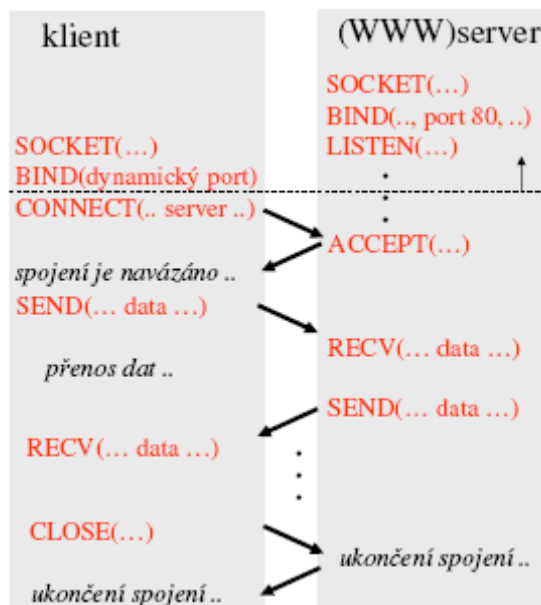
Port – adresa, na ktorej je poskytovana urcita sluzba.

Su to porty 0-1023 a na tychto portoch su poskytovane obvykle sluzby.

Vyznam tychto portov je apriorne dany, prideluje IANA (ICANN).

Port #	Popis
21	FTP
23	Telnet
25	SMTP
69	TFTP
70	Gopher
80	HTTP
88	Kerberos
110	POP3
119	NNTP
143	IMAP
161	SNMP

### 7. Naznačte příklad práce se sockety při spojované komunikaci (v TCP/IP)



## 8. Naznačte příklad práce se sockety při nespojované komunikaci (v TCP/IP)

- se sockety lze provádět další "primitivní operace"
  - **SENDTO(socket, data, ..., adresa ....)**
    - pošle data zadanému příjemci
    - určeno pro nespojovaný způsob komunikace, bez navazování spojení
  - **RECVFROM(socket, ..., adresa, ...)**
    - přijme data nespojovaným způsobem

```
příklad: činnost serveru při nespojované komunikaci:  
  
newsock=SOCKET(...);  
BIND(newsock, číslo portu);  
  
repeat  
    RECVFROM(newsock, adresa.);  
    zpracování požadavku z "adresa"  
until ....
```

## 9. Jakým způsobem (podle čeho) volí protokol TCP velikost časového limitu (timeout-u)

TCP monitoruje správanie siete a podľa neho sa mení dĺžka časového intervalu, ktorý čaká na potvrdenie.

Monitoruje dobu obrátky: nepozná prenosové oneskorenia, ale sleduje, ako dlho dostáva odpoveď. Vyhodnocuje (robí vážený prímer doby obrátky, rozptyl obrátky), čakajúcu dobu vypočíta ako funkciu váženého priemeru a rozptylu.

Efekt: čakacia doba vychádza tesne nad strednou dobou obrátky, dobre reaguje na predĺžovanie doby obrátky pri dávkach paketov, skrátenie doby obrátky.

## 10. Jakým způsobem pracuje protokol TCP s bytovým proudem (při zajišťování spolehlivého přenosu)?

Protokol TCP prijíma/vydáva data po bytoch (pracuje s byte streamom): data bufferuje (ukladá do bufferu, veľkosť je určená podľa parametru MTU).

Obsah bufferu je odosielaný až po jeho naplnení.

Potrebuje označovať jednotlivé byty v rámci prúdu (nepracuje s blokmi), potrebuje to na potvrdzovanie. Využíva k tomu 32-bitovú pozíciu v bytovom prúde.

## 11. Jakým způsobem předchází protokol TCP zahlcení?

Každú stratu dát chápe ako dôsledok zahlčenia. Po strate paketu ho pošle znovu, ale neposiela ďalšie, ale čaká na potvrdenie, t.j. Prechádza od kontinuálneho k jednotlivému potvrdzovaniu. Ak príde potvrdenie včas, odosle dvojnásobok a čaká na potvrdenie. Takto postupuje, kým nenarazí na obmedzenie dane aktuálnou veľkosťou okienka.

## 12. Jakým způsobem lze zajistit podporu QoS (nejen na úrovni transportní vrstvy)?

- **Priamo na úrovni sietovej vrstvy:** kde vzniká „best effort“
- **Predpoklady na sietovej vrstve, hlavná časť riešenia na transportnej:** na úrovni sietovej vrstvy sa rezervujú určité kapacity, ktoré sa potom využívajú na úrovni transportnej vrstvy.
- **Na transportnej alebo aplikacnej vrstve:** bez predpokladov na sietovej vrstve.

Možnosti:

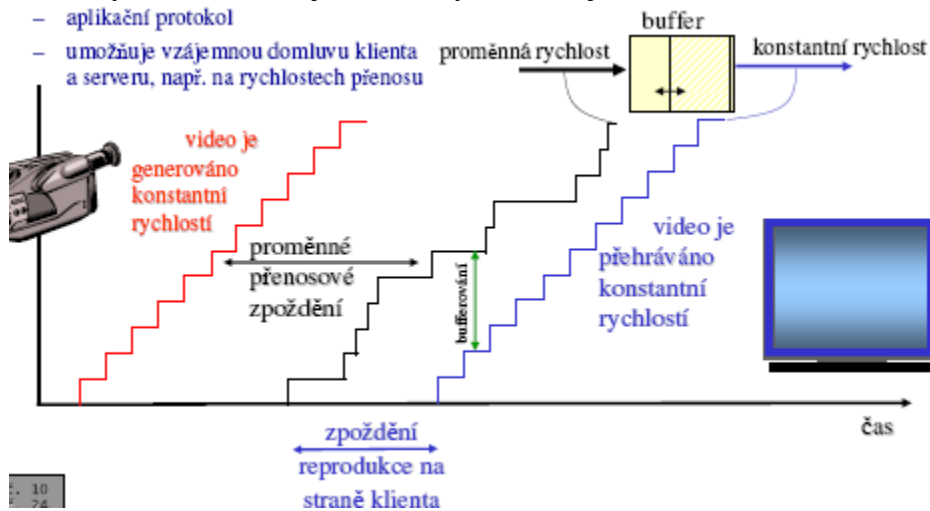
- traffic conditioning: úpravy dátového provozu, aby lepšie presiel
- Integrated Services: zásadnejšie zmeny v prenosovej časti siete tak, aby bolo možné rezervovať potrebné zdroje

- Differentiated Services: menšie zásahy do prenosovej časti siete, snaha diferencovať provoz a poskytnúť aspoň záruku rozdielu

### 13. Jaké jsou požadavky aplikací na QoS? Rozdělte alespoň na "datové" a "multimediální" aplikace.

### 14. Jak funguje technika "client buffering" (pro podporu QoS)?

U jednosměrných multimedií je možné vyrovnávat jitter až u klienta bufferováním.



### 15. Co přináší a jak funguje protokol RTP (a RTCP)?

**RTP (Real Time Protocol):** balí jednotlivé části multimediálních dat do vlastních bloků a tie vklada do UDP paketov, pripaja informacie o type multimediálneho obsahu, poradí paketov (jednotlivé pakety čísluje), o case vzniku dat, o konkrétnom streame (v rámci jedného RTP prenosu môže byť prenasaných viac prúdov, podporuje multicast).

**RTCP (Real Time Control Protocol):** sprostredkuvava vzajomne informovanie zdroja a prijemcu, napr o percente stratených paketov a ich spozdení...

### 16. Jaký je princip QoS Integrated Services? Jak souvisí s protokolem RSVP?

Snaha je každému poskytnout to, co potřebuje.

Princip: při navazování spojení žadatel specifikuje, co bude potřebovat (kapacitu, rychlost...), síť posoudí, či to dokáže zajistit a garantovat. Ak ano, spojenie je naviazane, ak nie, žiadosť je odmietnuta.

Možnosti realizácie: nutné k tomu vyhradiť potrebný objem zdrojov, nie je možné riešiť vyhradne na úrovni transportnej vrstvy., nutná spolupráca na úrovni sietovej vrstvy.

Realizácia:

- žiadateľ o naviazanie spojenia uvedie svoje R-spec (co bude požadovať)
- uvedie svoje T-spec (ako bude vyzerat datový provoz, ktorý bude generovať)
- Protokol RSVP (ReSerVation Protokol) preda R-spec a T-spec jednotlivým smerovacom v sieti a dohodne s nimi ich akceptvanie/odmietnutie, tie potom vyhradia zdroje pre prave vznikajúci virtuálny okruh.

### 17. Jaký je princip QoS Differentiated Services?

Nesnaží sa o absolútne naplnenie požiadaviek, ale ponúknuť aspoň relatívne (rozdielové) služby, jeden druh provozu je uprednostnený na úkor druhého.

Princip: zavedie sa niekoľko tried provozu (každá trieda bude mať inú prioritu), každý prenasaný paket sa môže prihlásiť k určitej triede, musí to byť podporované v celej sieti, uz na úrovni sietovej vrstvy.

Využitie: jednotlivé pakety deklarujú svoju príslušnosť k určitej triede provozu cez vhodnú nalepku (prefix, nastavenie údajov v hlavičke), v IP4 sa využíva byte ToS (Type of Service)

### 18. Jaké jsou úkoly relační vrstvy? Které současné protokoly by se hodily pro

## **původní představu relační vrstvy?**

### **Ulohy:**

- viest relacie a riadiť dialóg
- zaistovať synchronizáciu
- podporovať transakcie
- zaistovať bezpečnosť...

Protokoly: neviem

## **19. Jaké jsou úkoly prezentační vrstvy? Jakými způsoby mohou být řešeny?**

Nízšie vrstvy sa snažia doručiť každý bit presne tak ako bol odoslaný. Postupnosť bitov môže mať iný význam pre odosielateľa a iný pre prijemcu, napríklad kvôli: kodovaniu znakov (ASCII...), formátovaniu textu, formátu čísel...

Prez. vrstva rieši konverzie, aby obe strany interpretovali prenesené údaje rovnako, prevádza údaje z/do takej formy, v akej sa prenosujú.

### **Možnosti:**

- prispôbenie štýlom každý s každým: dvojica sa dohodne na spoločnom formáte, konverzia sa prevádza len 1x (u odosielateľa/prijemcu), v praxi problematicke
- cez spoločný medzitvar: je definovaný jeden spoločný medzitvar, konverzia sa prevádza 2x (odosielateľ a prijemca), jednoduchšie pre vzájomnú koordináciu, stačí sa naučiť prevádzať z/do spoločného medzitvaru

## **20. Jaký způsob řešení konverzí předpokládá jazyk ASN.1 v ISO/OSI a jakou úlohu plní?**

Služi na abstraktný popis dát obecných dát.

Problém: ako zabezpečiť aby prijemca vedel rekonštruovať údaje prijaté v medzitvare?

U odosielateľa sa údaje popisujú, vznikne prívodka k dátam, údaje sa prevedú do prenosovej formy a k nim sa pripojí prívodka. Prijemca podľa prívodky prevedie údaje z prenosovej formy do takej, ktorú má pre neho rovnaký význam.

Je potrebný jazyk pre prívodku, túto úlohu plní jazyk ASN.1

Využíva pravidlá BER (Basic Encoding Rules), ktoré hovoria ako reprezentovať jednotlivé dátové položky, používa sa tzv. TLV kódovanie (Type, Length, Value)