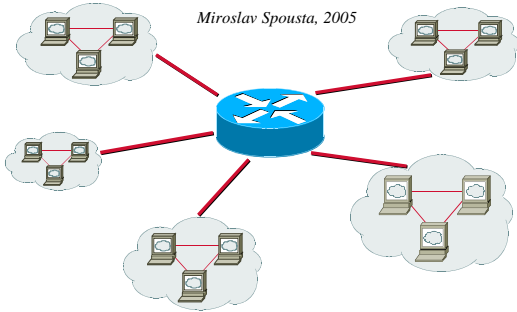


Počítačová síť II

14. Transportní vrstva: TCP a UDP



1

Transportní vrstva

- přítomná v ISO/OSI i TCP/IP
- zodpovídá za rozšíření vlastností, které požadují vyšší vrstvy (aplikace)
 - spolehlivost
 - spojitost
- další možnosti zlepšení
 - řízení toku
 - bufferované vysílání a příjem
 - rozlišení mezi více adresáty v rámci uzlu
 - plnduplexní spojení
 - transparentní přenos libovolných dat
- pozorování: ne všechny aplikace vyžadují všechny tyto vlastnosti

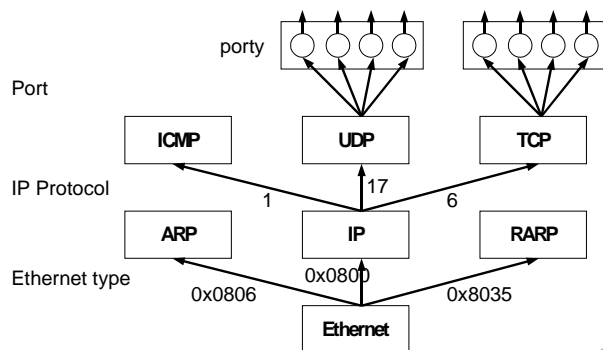
2

TCP a UDP

- aplikace mohou využívat dva způsoby komunikace
- UDP (User Datagram Protocol)
 - má obdobné vlastnosti jako protokol IP
 - přidává navíc možnost adresovat více procesů v rámci jednoho uzlu (pomocí portů)
 - přidává kontrolní součet k celému datagramu (IP nemá!)
- TCP (Transmission Control Protocol)
 - podstatně rozšiřuje vlastnosti IP
 - spolehlivý (ochrání datagram před ztrátou)
 - spojovaný - před komunikací je nutné navázat spojení, také se spojení ukončuje
 - libovolně velká data pro přenos
 - řízení toku dat (ochrana proti zahlcení příjemce i sítě)
 - stejně jako UDP nabízí jemnější adresaci než IP protokol pomocí portů

3

TCP/IP



4

Porty

- porty (doslova přístavy) jsou reprezentovány celým číslem (1..65535)
 - je to jakási relativní adresa v rámci uzlu
- porty existují „od začátku“, nevznikají, ani nezanikají
- aplikace se připojují k portům, pokud chtějí komunikovat pomocí transportních protokolů
 - mohou využívat více portů pro různou komunikaci
 - ale k jednomu portu patří nejvíce jedna aplikace
- některé porty jsou tzv. dobře známé (well-known)

5

Well-known porty

- některé porty, mají definovaný význam
 - udávají, jaká aplikace (služba) na daném portu pracuje
 - umožňují komunikaci s aplikací na vzdáleném serveru (pokud by byly porty náhodně, nevíme, kam se připojit)
- well-known (dobře známé) porty: mají čísla 0-1023
 - přiděluje IANA, původně jako RFC, dnes <http://www.iana.org/assignments/port-numbers>
 - 21 - FTP, 22 - SSH (secure shell), 23 - telnet, 25 - SMTP (pošta), 53 - DNS
 - 80 - HTTP (WWW), 110 - POP3 (vybírání pošty), 143 - IMAP
 - 443 - HTTPS (šifrovaný web), ...
- registrované porty: mají čísla 1024 - 49151
 - IANA nepřiděluje, pouze registruje používání (3306 - MySQL, 1433 - MSSQL)
- ostatní porty: 49152 - 65535
 - volně k použití (např. pro klienty)

6

Sockety

- *spojení* je identifikováno pomocí dvou socket
 - každý na jedné straně komunikujícího páru uzlů
- socket je dvojice IP adresy – číslo portu
 - např. 192.168.33.1:3306
- na jednom portu tedy může probíhat několik spojení
 - rozeznávají se podle adresy/portu protistrany
 - např. web server: spojení pracuje na portu 80, ale liší se socketem protistrany (klienta)
- sockety postupně vznikly v UNIXu (kde je všechno soubor) – pracuje se s nimi jako se soubory

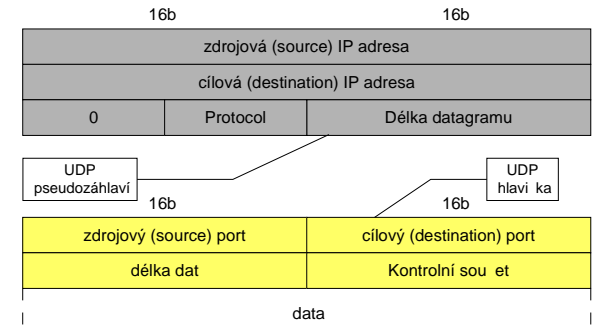
7

UDP

- velmi jednoduchá nastavení nad IP
- navíc poskytuje (de)multiplexování v rámci jednoho uzlu
 - neboli porty
- má malou režii (8 byte)
 - výhodné pro aplikace, které nepotřebují vlastnosti TCP
 - ale hodí se jim jednoduchost (a rychlost), bezstavovost
- může být použit pro broadcast/multicast
- také zahrnuje kontrolní součet pro celý datagram (0 – nepoužitý)
 - dokonce i přes pseudozáhlaví – data, která se nepřenášejí (jsou již obsažena v IP hlavičce), ale kontrolní součet se počítá, jako by byla přítomna
 - pokud uzel přijme UDP datagram se špatným kontrolním součtem – zahodí se
- velikost datagramu musí být taková, aby se vešel do IP datagramu

8

UDP



9

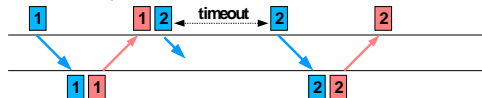
TCP

- výborně řeší problém spolehlivé komunikace – nad IP
- oproti ostatním protokolům (IP, UDP) velmi složitý
- vlastnosti:
 - spojovaný charakter (3 fáze: navázání spojení, přenos dat, ukončení spojení)
 - jedná se o dvoubodovou komunikaci (vždy komunikují právě dva uzly)
 - obousměrný (plně duplexní) přenos dat
 - implementuje řízení toku
 - předchází zahlcení
 - spolehlivost (automatické přeposílání ztracených/poškozených dat)
 - v vyšším protokolu se tváří jako bytová rouha
 - korektní navázání a ukončení spojení (obě strany souhlasí s navázáním spojení, nedojde ke ztrátě dat při navazování ani při ukončování spojení)

10

TCP

- spojovaná komunikace:
 - pouze na transportní vrstvě, neumožňuje využívat nižší vrstvy
 - neodpovídá úplně virtuálním okruhům
 - týká se pouze koncových uzlů, nikoli transportní (IP) infrastruktury
- zajištění spolehlivosti: kontinuální potvrzování a retransmise
 - generují se pozitivní potvrzení (že příjemce data obdržel)
 - odesílatel čeká na příjem potvrzení od příjemce
 - pokud ho nedostane (příjemce nedostal data, nebo se ziarilo po cestě potvrzení) do určité doby od odeslání, vyšle *segment* dat znovu
 - otázka, jak dlouho čeká?



11

Adaptivní opakování

- jak nastavit timeout
 - malá hodnota: způsobuje zbytečné retransmise
 - velká hodnota: způsobuje přenos při výpadku
- odesílatel sleduje přenosové zpoždění datagramu, podle toho si nastaví timeout
 - sleduje průměrnou round-trip (čas, za který mu přijde potvrzení)
 - timeout se dynamicky mění podle kvality linky (podle váženého průměru a rozptylu)
- timeout je něco málo nad střední hodnotou
 - pružně reaguje – je-li velký rozptyl, zvyšuje se, není-li, blíží se střední hodnotě
- potvrzování se vkládá do protisměrných paketů (není samostatné) – nemá velkou režii

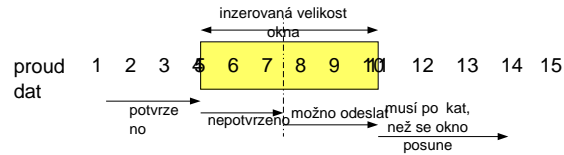
12

řízení toku dat

- TCP se na data dívá jako na posloupnost byt
 - postupně je bufferuje, až má dostatečně velký úsek (nebo po vypršení timeoutu), pak edá data ke zpracování nižší vrstvě (vytvorí tzv. segment)
- aby nedošlo k zahlcení příjemce, používá se řízení toku dat pomocí tzv. metody *klouzavého okna* (sliding window)
 - protisměrně inzeruje v každém poslaném segmentu dat, kolik má místa v bufferech
 - pokud inzeruje 0, má by uzel přestat vysílat
- je potřeba zabránit, aby příjemce inzeroval příliš malé okno (bajty)
 - při příliš vysoké rychlosti
 - potvrzení se posílá až když je zpracováno určité množství dat (zpoždění potvrzení, max. 500ms)
- stejně tak není žádoucí, aby odesílatel posílal malé segmenty
 - optimalizace rychlosti

13

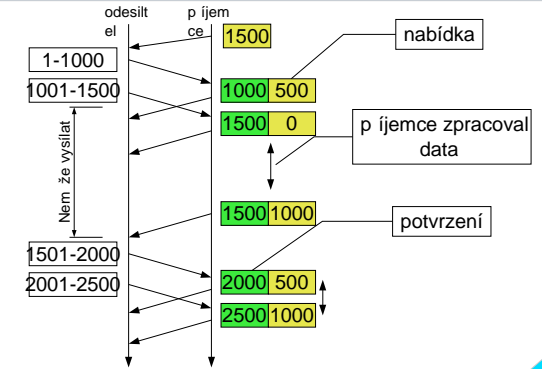
Sliding window



- v okně jsou nepotvrzená data
- při přijetí potvrzení se okno posune doprava
- pokud nemáme potvrzená data a vyčerпали jsme okno, musíme požadovat potvrzení
 - na potvrzení, případně (pokud vyprší časovač) na retransmisi

14

řízení toku



15

TCP

| | | | |
|--------------------------|---|---------------------------|------------|
| 16b | | 16b | |
| zdrojový (source) port | | cílový (destination) port | |
| pořadové číslo (seq) | | | |
| číslo potvrzení (ack) | | | |
| délka záhlaví | 0 | flags (6 bit) | šířka okna |
| kontrolní součet | | ukazatel spouščených dat | |
| optional data (+padding) | | | |
| data | | | |

16

Popis TCP segmentu

- porty: 0 – 65535 (jsou v prvních 64 bitech (kvůli ICMP))
- pořadové číslo: sequence number – označuje, kolik byte bylo odesláno
 - na počátku spojení je to náhodné číslo
 - kvůli bezpečnosti (aby se nedalo odhadnout)
- číslo potvrzení – udává, že všechny předcházející byte jsou potvrzeny
 - je to číslo, které protistrana očekává jako následující
- délka záhlaví – v násobcích 32 bit (4B)
- šířka okna: počet byte, které je možné ještě vyslat bez požadování potvrzení
 - odpovídá velikosti vyrovnávací paměti protistrany
- kontrolní součet – zabezpečení přes celý segment (včetně pseudozáhlaví)
- ukazatel spouščených dat – poslední byte urgentních dat
- volitelné možnosti – například MSS

17

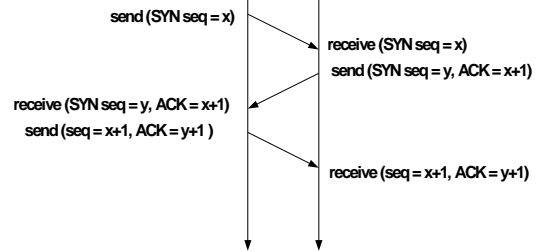
Průznaky TCP segmentu

- bitové pole (6b)
- URG – spouščená (urgentní data) v segmentu
- ACK – udává, že pole potvrzení je platné
- PSH – požaduje okamžité doručení segmentu vyšší vrstvy (pro interaktivní aplikace, ...)
- RST – požaduje okamžité ukončení spojení
- SYN – žádost o navázání spojení (segment neobsahuje data)
- FIN – žádost o přerušení spojení

18

Navázání spojení

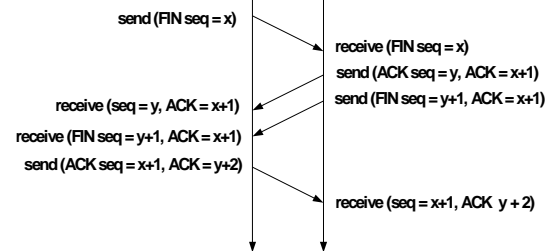
- musí fungovat spolehlivě i v případě ztráty některého IP datagramu
- jedna strana (klient) je aktivní, protistrana (server) je (zpočátku) pasivní



19

Ukončení spojení

- stejně jako navázání musí fungovat spolehlivě
 - i v případě ztráty některého IP datagramu
 - zaručit, že všechna data, která byla odeslána před uzavřením dojdou (v obou směrech!)



20