

Počítačové sítě

Jan Outrata



KATEDRA INFORMATIKY
UNIVERZITA PALACKÉHO V OLMOUCI

přednášky

Tyto slajdy byly jako výukové a studijní materiály vytvořeny za podpory grantu FRVŠ 1358/2010/F1a.



Transportní vrstva



“Proč dva protokoly?”

- síťové protokoly přepravují data mezi libovolnými **uzly** (počítači) v síti, adresují síťová rozhraní uzlu
- přepravují data mezi dvěma **aplikacemi** (programy, realizujícími síťové služby) běžícími na uzlech, adresují aplikace na uzlu
- zprostředkovávají **transparentní spojení** s požadovanou kvalitou mezi více aplikacemi v rámci síťových zařízení (uzlů)

Služby

■ **spojovaná (connection oriented):**

- mezi aplikacemi navázáno **spojení** (vytvořen virtuální okruh daných parametrů), s **plně duplexní** výměnou dat
- (typicky) ztracená nebo poškozená data znovu vyžádána – „**spolehlivá**“ služba
- integrita dat zabezpečena kontrolním součtem
- zpracovává **souvislý proud/tok (uspořádaných) dat** od vyšší vrstvy (**stream**)

■ **nespojovaná (connectionless):**

- nenavazuje spojení
- data odeslána, (typicky) nezaručuje se doručení ani znovuzasílání ztracených nebo poškozených dat (ponecháno na vyšším protokolu) – “**nespolehlivá**” (**datagramová**) služba
- integrita dat zabezpečena kontrolním součtem
- zpracovává (**nesouvislé**) **části dat** od vyšší vrstvy (datagramy), rozdělení toku dat na datagramy řeší vyšší vrstva

Port

- = identifikátor aplikace (aplikace jich může používat víc), transportní adresa
- = číslo délky 2 B, 0 až 65535
- porty 0 – 1023 jsou tzv. **privilegované** (může je použít pouze privilegovaná aplikace, např. systémová služba nebo privilegovaného uživatele), ostatní **neprivilegované** (může použít kdokoliv, pokud je volný)
- pro běžné služby (aplikační protokoly) Internetu všeobecně známá „standardní“ (**well-known**) čísla portů přidělována IANA, privilegovaných i neprivilegovaných

CVIČENÍ: zjištění čísel portů nejznámějších služeb Internetu, např. jmenné (aplikační protokol DNS), vzdáleného přihlášení (Telnet, SSH), přenosu dat (FTP(S), SMB), poštovní (SMTP, POP3(S), IMAP(S)), webové (HTTP(S)), a LAN (DHCP, SNMP)

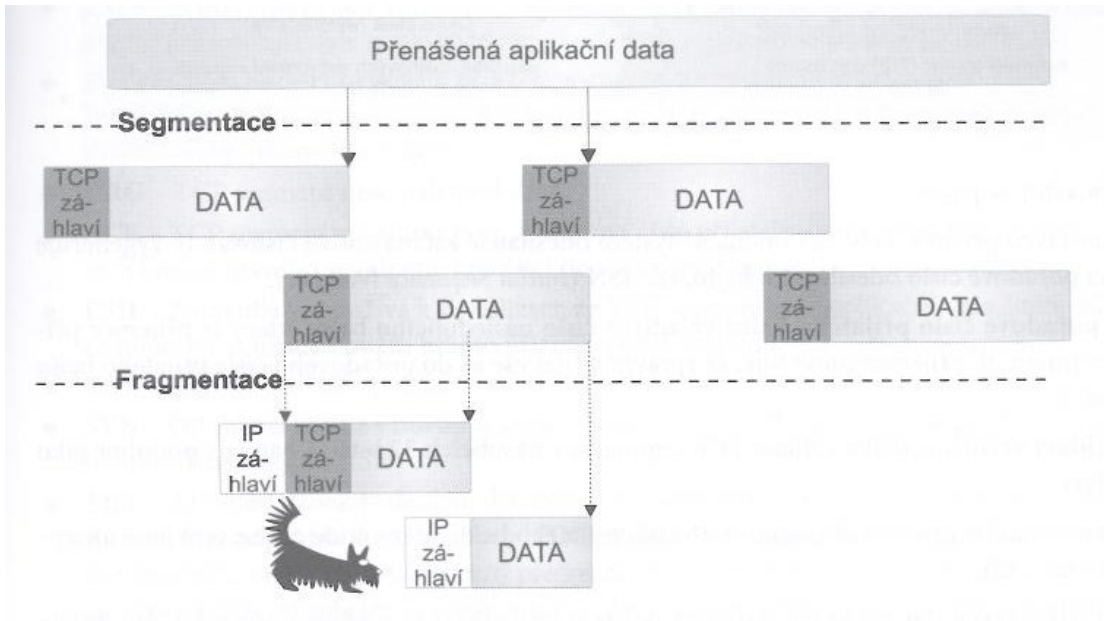
- aplikace jednoznačně určena: síťovou (IP) adresou, číslem portu a transportním protokolem (TCP/UDP), tzv. adresa **socketu** (síťového rozhraní **Socket API**)

Datagram/Segment

- = základní jednotka přenosu, transportní paket/datagram/segment, vkládán do síťového paketu
- obsahuje část (tok) dat od odesílatele k příjemci od vyšší vrstvy
- **segmentace** = rozdělení toku dat na části zapouzdřené do segmentů

Obrázek: Obrázek průvodce 231

- max. délka = max. délka síťového paketu (64 kB u IP) minus délka jeho záhlaví
- obsahuje záhlaví s porty příjemce a odesílatele + data

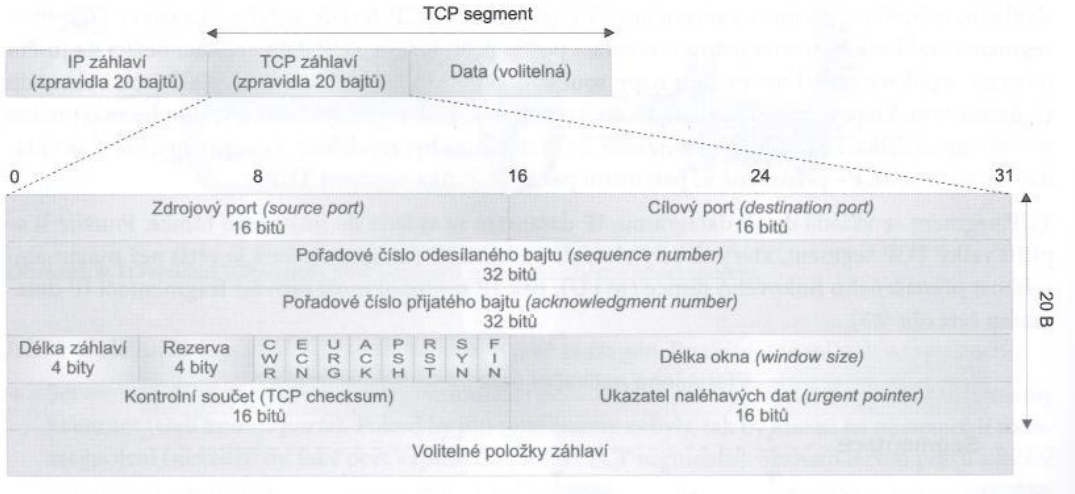


- RFC 962
- IP protokol poskytuje datagramovou (nespojovanou) „nespolehlivou“ službu, bez vyžadování opakování přenosu paketů, nanejvýš signalizace nemožnosti doručení (ICMP, nepovinná, potlačovaná)
- poskytuje spojovanou „spolehlivou“ službu, řeší:
 - navázání, udržování a ukončení **plně duplexního spojení**
 - adaptivní přizpůsobení parametrů protokolu podle stavu spojení
 - zaručení správného **pořadí dat**
 - potvrzování přijetí dat (tzv. **pozitivní potvrzování**)
 - vyžádání **opakování přenosu** ztracených nebo poškozených dat
 - **řízení toku dat** a **předcházení zahlcení sítě** pomocí časových prodlev, opakovaného odeslání a potvrzení přijetí dat, bufferů a **posuvného okna** a **okna zahlcení**
- nezávislý rozsah portů pro TCP a UDP, TCP porty označeny **číslo/tcp**

Obrázek: Obrázek průvodce 232

- záhlaví 20 B povinných položek + volitelné položky, data
- **identifikace spojení** (v Internetu) = zdrojový a cílový port, zdrojová a cílová IP adresa, transportní protokol (TCP)
- **pořadové číslo odesílaného bytu**: pořadové číslo 1. bytu dat z odesílaného toku dat (spojení) v segmentu, číslování při navázání spojení začíná od náhodného čísla (ISN, Initial Sequence Number), po dosažení $2^{32} - 1$ opět od 0 – pro zajištění správného **pořadí dat**
- **pořadové číslo přijatého bytu**: pořadové číslo následujícího bytu, který má být přijat – pro zajištění **pozitivního potvrzování** a opakování přenosu dat
- **délka záhlaví**: v jednotkách 4 B, max. 60 B

TCP segment



- příznaky:
 - **CWR, ECN** – pro (volitelné) oznámení zahlcení sítě, viz dále, bez zahazování dat, tzv. ECN (Explicit Congestion Notification), v kombinaci s IP (2 bity u položky TOS záhlaví IP paketu)
 - **URG** – segment nese naléhavá data, která má příjemce zpracovat přednostně (out of band data, použití výjimečně, např. u Telnetu pro příkazy)
 - **ACK** – signalizace správného pořadového čísla přijatého bytu, tj. potvrzení správného přijetí bytů segmentu až do tohoto čísla - 1 = **pozitivní potvrzení**
 - **PSH** – segment obsahuje aplikační data, použití není ustáleno
 - **RST** – **odmítnutí** navazovaného TCP **spojení**
 - **SYN** – nová sekvence číslování odesílaných bytů, pořadové číslo odesílaného bytu je číslo 1. bytu toku dat (ISN), nastaven u 1. segmentu při **navazování spojení**
 - **FIN** – ukončení odesílání dat (dalších, tj. s výjimkou opakování přenosu dat), **ukončení spojení pro daný směr přenosu dat**
- **délka okna**: počet bytů, které je příjemce schopen přijmout – předcházení zahlcení přijímače v rámci **řízení toku dat**

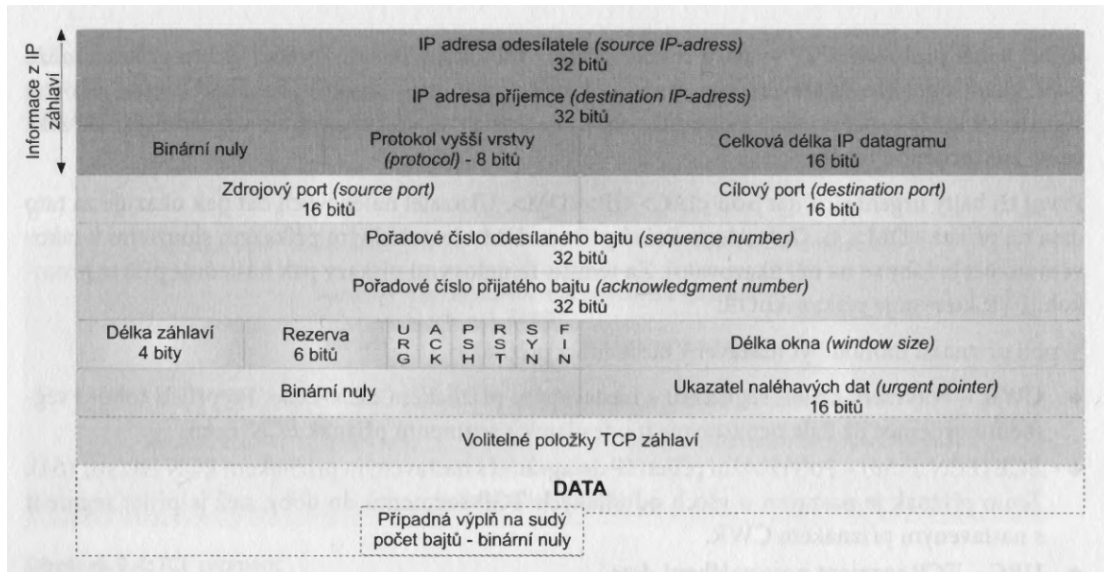
- **kontrolní součet**: počítaný z některých položek IP záhlaví (IP adresy odesílatele a příjemce, 1 B bin. nul, protokol vyšší vrstvy, celková délka IP paketu) a záhlaví TCP segmentu, tzv. **pseudozáhlaví**, a **dat segmentu** (plus případně 1 B bin. nul výplně na sudý počet bytů) – zajištění integrity dat

Obrázek: Obrázek průvodce 234

- **ukazatel naléhavých dat**: počet bytů odesílaných naléhavých dat od začátku dat v segmentu nebo offset začátku naléhavých dat (závisí na aplikaci, např. příkazy Telnetu), pouze při příznaku URG

CVIČENÍ: zachytávání a inspekce TCP segmentů

TCP segment




- max. 40 B za povinnými položkami TCP segmentu

Obrázek: Obrázek průvodce 235

- typ 0 pro poslední položku, 1 pro výplň záhlaví na násobek 4 B
- **max. délka segmentu (MSS)**, typ 2: max. délka dat přijímaných segmentů, dohodnutá stranami při navazování spojení, jen s příznakem SYN
- **zvětšení okna**, typ 3: délka bitového posunu doleva délky okna
- **povolení SACK a SACK (Selective ACK)**, typy 4 a 5: pro selektivní potvrzování segmentů mimo pořadí (vzhledem k pořadí dat), ukládání segmentů na straně příjemce do bufferu
- **časové razítko a echo časového razítka**, typ 8: echo je zopakování razítka z posledního přijatého segmentu, pro detekci starého zatoulaného segmentu při dlouhých oknech (stovky MB)
- a další, např. pro čítač spojení

Volitelné položky TCP záhlaví



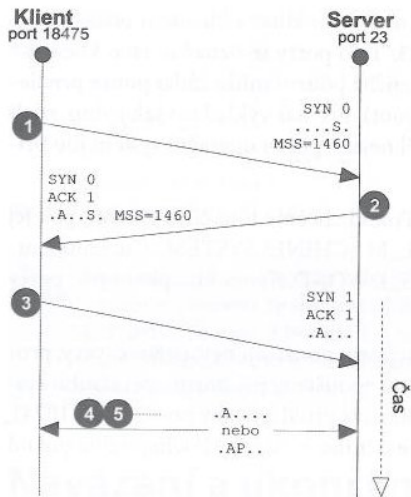
Typ (kind) 1 byte	Délka 1 byte	Hodnota	
0		Poslední (ukončující) volba End of option list - EOL	
1		Prázdňá volba (výplň) No operation - NOP	
2	4	max. délka segmentu - 2B (max. segment size - MSS)	
3	3	Zvětšení okna (Shift count) 1B	
4	2	SACK	
8	10	Časové razítko (Timestamp value) 4B	
11	6	Čítač spojení (connection count) 4B	
12	6	Nový čítač spojení (new connection count) 4B	
13	6	Echo čítače spojení (connection count echo) 4B	

- jedna strana spojení navazuje, druhá jej přijme nebo odmítne
 - model **klient/server** (z hlediska aplikační vrstvy) – klient navazuje, server očekává a případně přijímá
 - protokol TCP umožňuje navazovat spojení současně v obou směrech (v praxi ne příliš využívané) – POZOR!, neplést s **obousměrným přenosem dat** v rámci jednosměrně navázaného spojení!
- obě strany **otevrou port** (pomocí **socketu**), klient v tzv. **aktivním režimu** (navázání spojení), server v tzv. **pasivní režimu** (očekávání spojení)
- cílový port (na serveru) je daný aplikací
 - zdrojový port (na klientu) typicky náhodně vybrán OS z volných neprivilegovaných (≥ 1024)

Třífázový (Three-Way) handshake

Obrázek: Obrázek průvodce 238

- 1** klient odešle segment (bez dat) s příznakem **SYN**, náhodně vygenerovaným pořadovým číslem odesílaného bytu jako startovacím číslem (**fiktivního**) 1. bytu dat od klienta (ISN) a navrhovanou max. délkou přijímaných segmentů (MSS)
- 2** server odešle segment (bez dat) s příznaky **SYN** a **ACK**, náhodně vygenerovaným ISN a navrhovanou MSS pro směr od serveru, pořadové číslo přijatého bytu je klientovo **ISN + 1** (potvrzuje přijetí předchozího segmentu, fiktivního 1 byte dat, od klienta)
- 3** klient odešle segment (bez dat) s příznakem **ACK**, pořadové číslo odesílaného bytu je klientovo **ISN + 1** (další fiktivní byte dat, který server očekává), pořadové číslo přijatého bytu je serverovo **ISN + 1** (potvrzuje přijetí předchozího segmentu, fiktivního 1 byte dat, od serveru)





- po navázání spojení, tj. příjmu segmentu s příznakem ACK oběma stranami, lze zasílat oběma směry data (datové segmenty s příznaky ACK a PSH) nebo jen **potvrzovací segmenty** (s příznakem ACK)
- první segment s příznakem SYN nepotvrzuje žádná přijatá data, tj. neobsahuje příznak ACK a pole pořadové číslo přijatého bytu není platné (bývá vyplněno bin. nulami)
- navrhované **MSS** je \leq **MTU**, aby se zamezilo IP fragmentaci, pro Ethernet II 1460, Ethernet 802.3 1452

CVIČENÍ: zachytávání a inspekce TCP segmentů při navazování spojení, rozbor třífázového handshake

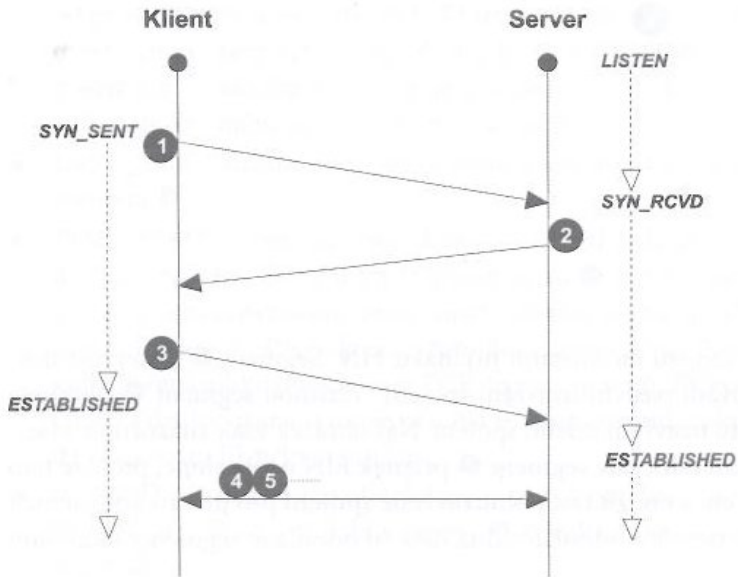
Stavy spojení při jeho navazování:

Obrázek: Obrázek průvodce 239

- LISTEN – stav serveru, čekání na navázání spojení ze strany klienta
- SYN_SENT – na straně klienta, po odeslání prvního segmentu (s příznakem SYN), tj. navazování spojení
- SYN_RCVD – na straně serveru, po obdržení prvního segmentu (s příznakem SYN), tj. obdržena žádost o spojení
- ESTABLISHED – na obou stranách, po obdržení prvního segmentu s příznakem ACK, tj. spojení navázáno (pro přenos dat ve směru od strany, která segment **obdržela**)

Všechna spojení a jejich stavy lze zobrazit např. programem `netstat`.

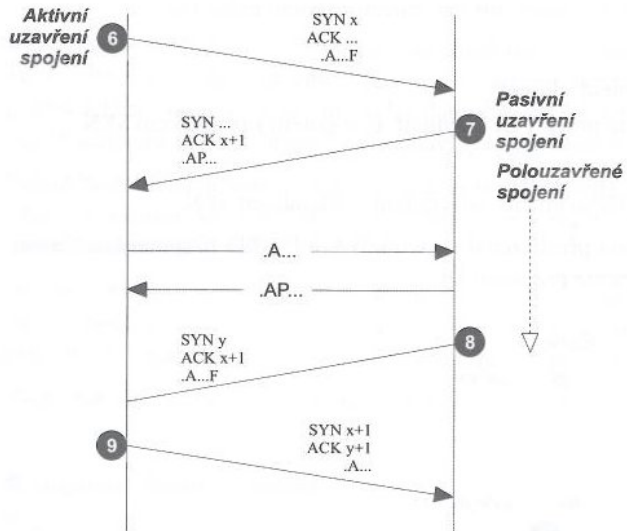
CVIČENÍ: výpis všech spojení na z/do počítače, identifikace IP adres a portů (aplikací) stran a stavů spojení, např. pomocí programu `netstat`



- ukončit/uzavřít spojení může libovolná strana, klient i server

Obrázek: Obrázek průvodce 240

1. strana odešle segment (možno i s daty) s příznakem **FIN** (vedle ACK), tzv. **aktivní uzavření spojení**, pak již nemůže odesílat datové segmenty (s příznakem PSH)
2. strana odešle segment (potvrzovací, možno i s daty) bez příznaku FIN (jen s **ACK**), tzv. **pasivní uzavření spojení**, může dál odesílat datové segmenty 1. straně tzv. **polouzavřeným spojením**
3. 2. strana odešle segment (možno i s daty) s příznakem **FIN** (vedle ACK), tzv. **úplné uzavření spojení**
4. 1. strana odešle potvrzovací segment (bez dat, s příznakem **ACK**)



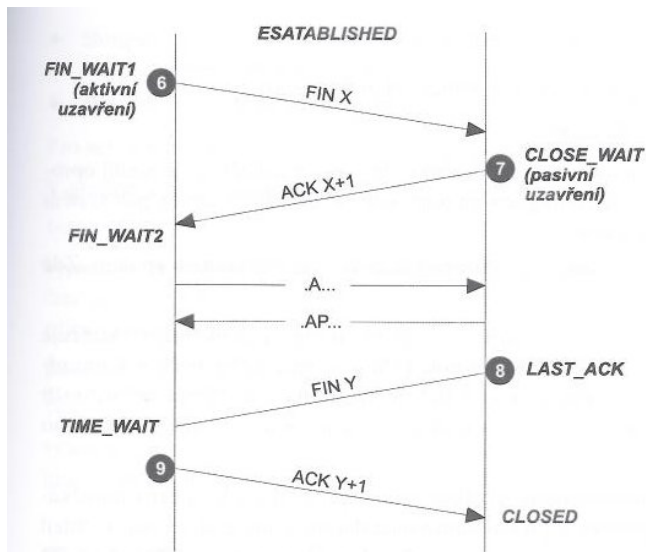
- 2. krok je možné vynechat, při oboustranném uzavření spojení
- segment s příznakem FIN bez dat se potvrzuje (segmentem s ACK) jakoby měl 1 byte (fiktivních) dat

CVIČENÍ: zachytávání a inspekce TCP segmentů při ukončování spojení, rozbor sekvence segmentů ukončujících spojení

Stavy spojení při jeho ukončování:

Obrázek: Obrázek průvodce 241

- FIN_WAIT1 – na 1. straně, po odeslání segmentu s příznakem FIN, tj. aktivní uzavření spojení
- CLOSE_WAIT – na 2. straně, po obdržení segmentu s příznakem FIN a odeslání segmentu jen s příznakem ACK (bez FIN), tj. pasivní uzavření spojení





- **FIN_WAIT2** – na 1. straně, po obdržení (potvrzovacího) segmentu bez příznaku FIN, po 11,25 min. nečinnosti polouzavřeného spojení (tj. bez přijetí segmentu) přechází do stavu CLOSED
- **LAST_ACK** – na 2. straně, po odeslání segmentu s příznakem FIN, tj. úplné uzavření spojení
- **TIME_WAIT** – na 1. straně, po obdržení segmentu s příznakem FIN a odeslání potvrzovacího segmentu, protože potvrzovací segment není potvrzován, po 30 s – 2 min. přechází do stavu CLOSED, kvůli možnosti opakování potvrzovacího segmentu po jeho vyžádání 2. stranou (při neobdržení)
- **CLOSED** – na obou stranách, na 2. straně po obdržení potvrzovacího segmentu

CVIČENÍ: výpis všech spojení na z/do počítače, identifikace IP adres a portů (aplikací) stran a stavů spojení, např. pomocí programu `netstat`



- pokud cílový port na straně příjemce není otevřen (např. neběží aplikace serveru, nebo jsou segmenty zahazovány firewallem), klient, bez odpovědi serveru, po vypršení časového intervalu **opakuje** požadavek na **navázání spojení** (1. segment s příznakem SYN) do vypršení celkového času nebo počtu pokusů → časová prodleva
- = kdykoliv zaslání segmentu s příznakem **RST** (bez dat) → **okamžité uzavření spojení** (v obou směrech) a přechod do stavu CLOSED na obou stranách
- použití např. u neúspěšného vytvoření šifrovaného kanálu u SSL/TLS
- použití také pro **rychlejší ukončení spojení**: nastavení příznaku RST místo FIN v 3. (nebo i 1.) segmentu při ukončování spojení, nebo po 4. segmentu ještě 2. strana odešle potvrzovací segment s příznakem RST, pro ušetření 1. straně čekání ve stavu TIME_WAIT

Odesílatel:

- má definovaný časový interval pro příjem potvrzovacího segmentu od příjemce (retransmission timeout)
- při ztrátě nebo poškození segmentu (odeslaného nebo potvrzovacího) po vypršení intervalu nebo příjmu tří opakovaných stejných potvrzení od příjemce (viz dále) **opakuje odeslání segmentu**
- hodnota intervalu se dynamicky mění podle stavu sítě (linky) – na základě předpokládané doby odezvy (vypočítané z RTT), Karn-Jacobsonův algoritmus

Příjemce:

- má definovaný časový interval pro příjem následujícího segmentu s dalšími daty v toku dat (dle pořadových čísel)
- při neobdržení následujícího segmentu po vypršení intervalu nebo obdržení segmentu s dalšími daty mimo pořadí **opakuje potvrzení přijetí** předchozích dat
- ukládá si i data mimo pořadí do vstupního bufferu, po obdržení chybějícího segmentu **potvrdí příjem všech dat**



CVIČENÍ: simulace ztráty segmentu (přerušením linky) a pozorování chování protokolu TCP při opakování odesílání a potvrzování dat

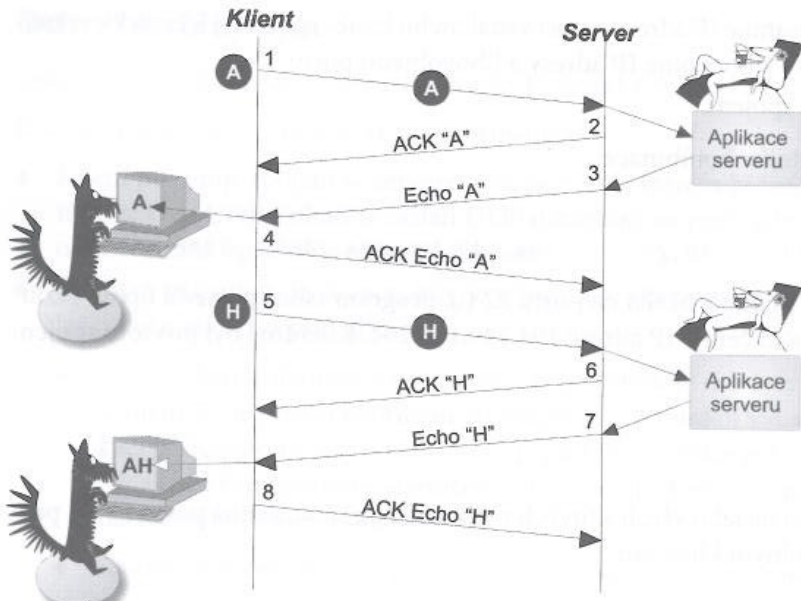
- výhodné u **interaktivních (konzolových) aplikací**, např. Telnet, FTP (příkazový kanál), SSH apod., vyměňujících **malé segmenty** (např. 1 B dat)

Obrázek: Obrázek průvodce 244

- klasický průběh: uživatel stiskne klávesu, klient odešle znak serveru (v segmentu v IP paketu v linkovém rámci), server potvrdí příjem, zpracuje znak, odešle znak klientovi pro jeho zobrazení (interaktivita), klient potvrdí příjem a zobrazí, tj. min. 117 bytů (pro Ethernet) v každém směru – **velká reže**
- snaha zmenšit objem přenášených dat a nebezpečí zahlcení sítě

CVIČENÍ: pozorování zpoždění odpovědi u aplikace Telnet (viz dále)

Zpoždění odpovědi

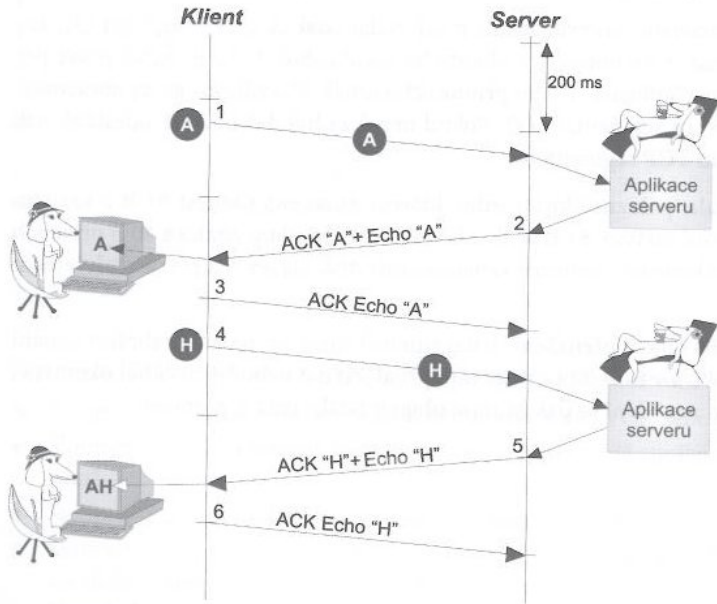


= **potvrzování příjmu dat** ne hned, ale **se zpožděním**, během kterého se mohou nahromadit data k odeslání:

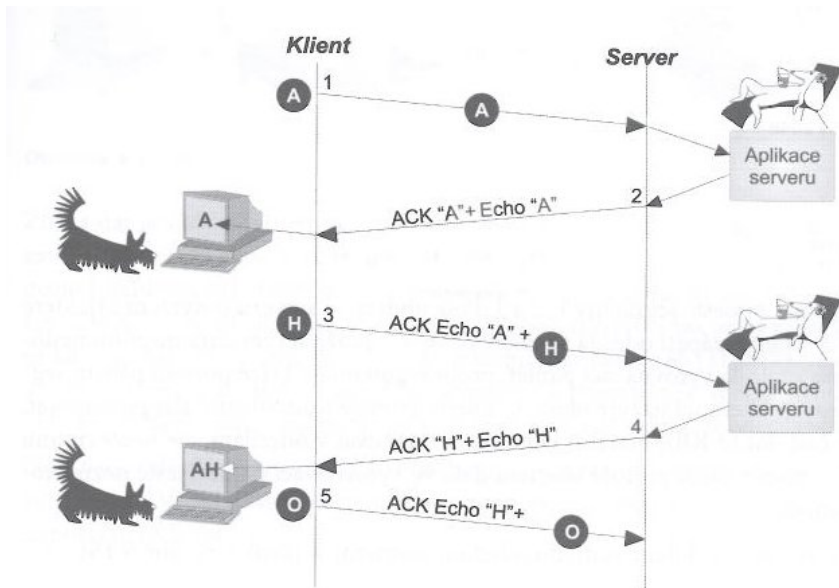
Obrázek: Obrázky průvodce 244,245

- **„delayed ACK“**: odesílání dat včetně potvrzení **v intervalech** např. 200 ms (≤ 500 ms)
- **Nagleův algoritmus**: odesílání dat včetně potvrzení až **po obdržení dalších dat** od druhé strany nebo až je objem dat k odeslání \geq MSS – vyrovnává dobu odezvy vůči kapacitě přenosové cesty v síti
- kombinace způsobuje konstantní zpoždění potvrzování (“ACK delay”) → zakázání Nagleova algoritmu pomocí volby **TCP_NODELAY** síťového API OS

Zpoždění odpovědi



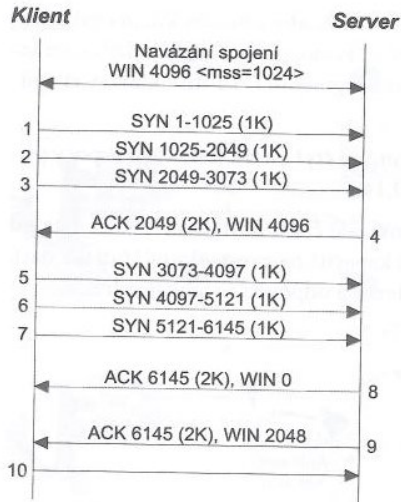
Zpoždění odpovědi



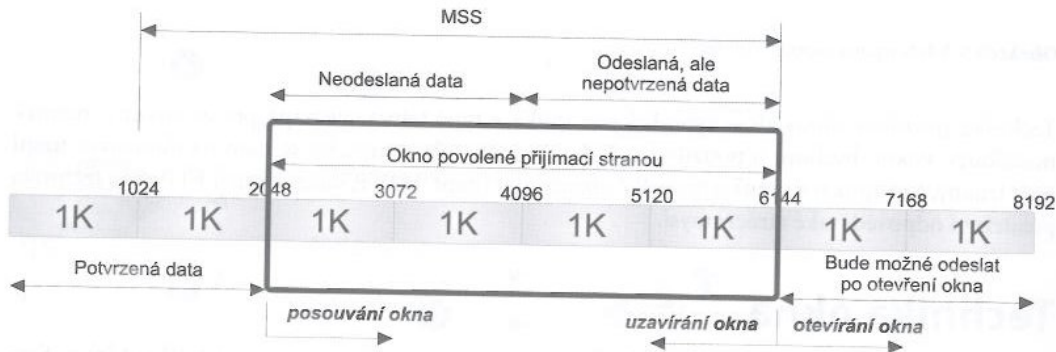
Obrázek: Obrázek průvodce 246

- využití při odesílání většího množství dat, **zamezení zahlcení příjemce**
- **segmenty se odesílají bez potvrzení každého zvlášť** až do počtu odeslaných bytů rovno délce **posuvného okna** (v položce délka okna v TCP segmentu, pak se ukládají do výstupního bufferu)
- délka okna vyjadřuje počet bytů, které je příjemce schopen přijmout (má plný vstupní buffer) či (v určitém čase) zpracovat
- při navazování spojení **příjemce** navrhne počáteční délku (spolu s MSS, typicky 6–8 MSS) a pak ji může **v potvrzovacích segmentech měnit (inzerovat)** nebo i vynulovat (okno „uzavřít“), tj. zakázat odesílateli odesílat další data (když „nestíhá“)

Posuvné okno (sliding window)



Posuvné okno (sliding window)





- položka délka okna má 2 B, tzn. okno může být dlouhé max. 64 kB – malé u rychlých sítí → volitelná položka **zvětšení okna**, $n = 0$ až 14, délka okna je potom násobena 2^n (posun o n bitů doleva), tj. až 1 GB, možno použít jen u segmentů s příznakem SYN při navazování spojení, nastavováno parametrem OS
- potvrzováním příjmu dat se okno po datech k odeslání „posouvá“ a mění velikost = řízení toku dat (**flow control**)

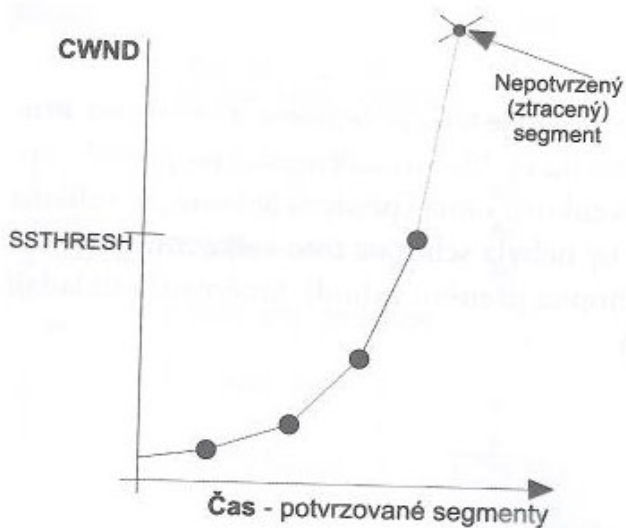
CVIČENÍ: identifikace a pozorování posuvného okna při přenosu dat

- posuvné okno udává množství dat akceptované příjemcem
- pokud je příliš velké a síť na straně příjemce plně využita nebo pomalá, odesílatel může síť zahltit a ta (směrovače) začne data zahazovat
- okno i na straně odesílatele: **okno zahlcení (congestion window)** = jaké množství **nepotvrzených dat je možné odeslat aniž by došlo k zahlcení sítě** – cíl: největší možné
- odesílatel odesílá data do velikosti menšího z posuvného okna a okna zahlcení
 - dvě fáze určování velikosti okna zahlcení: pomalý start a předcházení/vyhýbání se zahlcení

Pomalý start (slow start)

- = od navázání spojení se **velikost okna zahlcení (CWND)** počínaje MSS s každým potvrzeným segmentem **zdvojnásobuje**, až do ztráty segmentu nebo pokud by se překročila velikost posuvného okna nebo parametru **SSTHRESH** – hranice pravděpodobnosti zahlcení, první hodnota je parametr OS, typicky 64 kB
- při ztrátě segmentu:
 - po třech stejných potvrzeních předchozího segmentu se **CWND zmenší na polovinu** a na tuto hodnotu se také nastaví SSTHRESH (minimálně ale $2 \times \text{MSS}$)
 - po neobdržení potvrzení (v časovém intervalu) se **CWND nastaví na MSS** a SSTHRESH na $2 \times \text{MSS}$ a začne se **znovu**

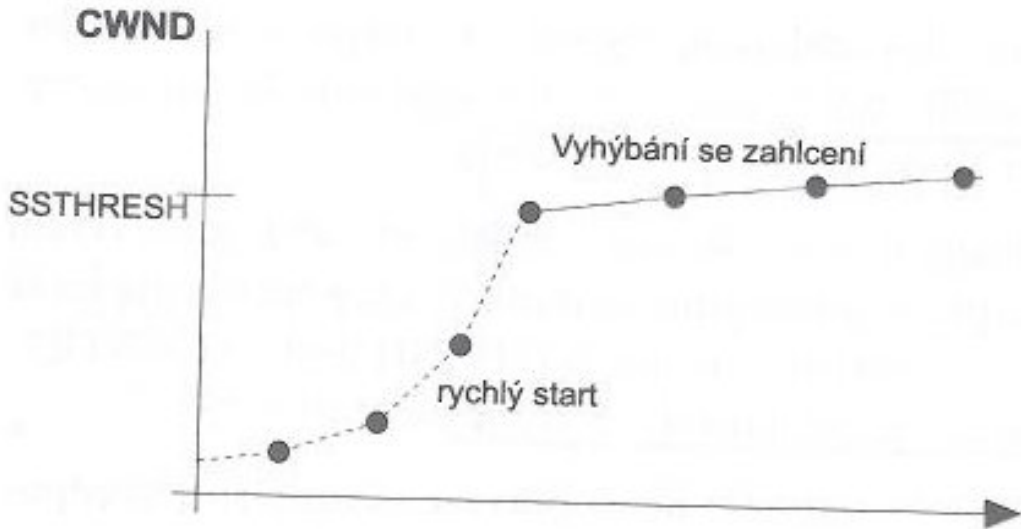
Obrázek: Obrázek průvodce 248



Předcházení/vyhýbání se zahlcení (congestion avoidance)

- následuje po pomalém startu, **pomalé zvětšování okna** s každým potvrzením, např. o MSS , $MSS^2/CWND + MSS/8$ apod.
- algoritmy vyhýbání se zahlcení (**congestion avoidance algorithms**): Tahoe (první), **Reno**, **New Reno**, Hybla (pro rádiové spoje), **BIC** (rychlejší adaptace pro rozsáhlé rychlé sítě), **CUBIC** (CWND je kubická funkce času od posledního zahlcení) aj.
- **selektivní potvrzování (selective ACK, SACK)** = potvrzování i segmentů mimo pořadí, pomocí volitelných položek záhlaví (s dohodou při navazování spojení)

Obrázek: Obrázek průvodce 248



- odesílatel udržuje pro každé spojení velikosti MSS, posuvného okna, okna zahlcení (CWND) a parametru Ssthresh
- nalezená hodnota Ssthresh pro daný směr se i po ukončení spojení použije jako výchozí u dalších spojení v tomto směru, uložena ve směrovací tabulce

Při ztrátě segmentu (během přenosu dat):

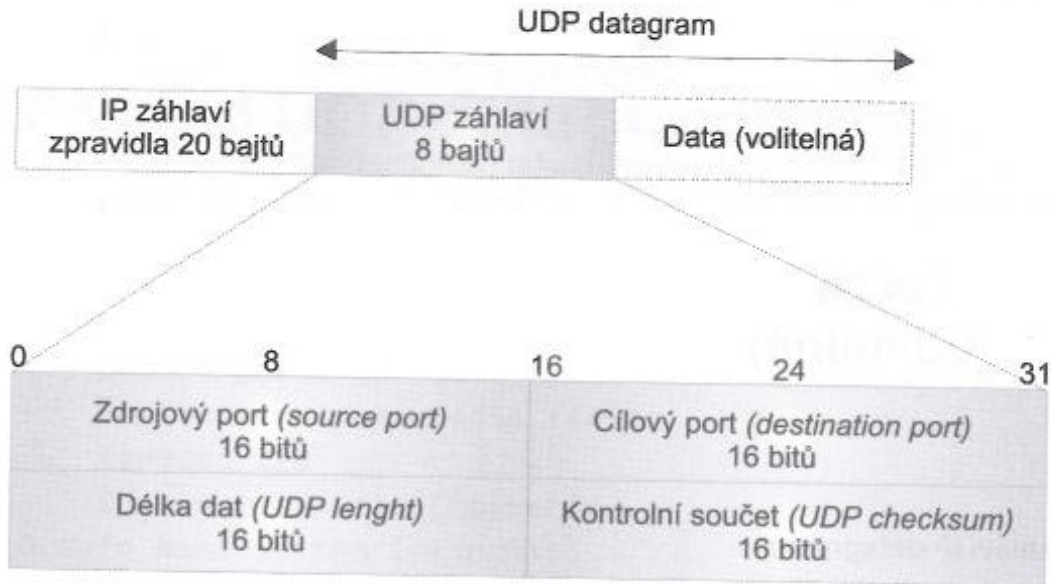
- po třetím stejném potvrzení se nastaví **Ssthresh na polovinu aktuální CWND** (minimálně $2 \times \text{MSS}$), zopakuje se segment, nastaví se **CWND na „o něco“ vyšší než Ssthresh** a při opakovaných potvrzeních se zvyšuje o MSS
- po potvrzení ztraceného segmentu (celého okna zahlcení) se nastaví **CWND na původní Ssthresh (rychlý start/zotavení)** a opět probíhá pomalé zvětšování okna (algoritmus vyhýbání se zahlcení)
- po neobdržení potvrzení (v časovém intervalu) **znovu pomalý start** (CWND = MSS, Ssthresh = $2 \times \text{MSS}$)

- RFC 768
- poskytuje **nespojovanou (datagramovou) „nespolehlivou“ službu**: data odeslána, nezaručuje se doručení ani znovuzasílání ztracených nebo poškozených dat – ponecháno na vyšším (aplikačním) protokolu
- **vyšší výkon** a rychlost přenosu dat než u TCP, za cenu „nespolehlivosti“ – využití u streamování multimediálního obsahu
- nezávislý rozsah portů pro TCP a UDP, UDP porty označeny **číslo/udp**
- snaha vyhnout se IP fragmentaci datagramů – **velikost datagramu** \leq **MTU** linky (např. u DNS delší odpověď zkrácena na 512 B a na vyžádání poslána celá pomocí TCP)
- oproti TCP může být příjemcem skupina uzlů, tj. **IP adresa příjemce** může být **všesměrová** (např. u DHCP) nebo **skupinová** (multicast, typicky u streamování multimediálního obsahu) – jak dožádat nedoručená data (např. u přenosu souborů pomocí Multicast FTP)? → od nejbližšího směrovače (protokolem pro multicast)

Obrázek: Obrázek průvodce 251

- záhlaví 8 B, data
- **délka dat:** délka datagramu, tj. záhlaví a dat
- **kontrolní součet:** stejně jako u TCP počítán z tzv. pseudozáhlaví (některé položky IP záhlaví a UDP záhlaví) a dat, nemusí být povinně vyplněný (nulový), pro zrychlení (např. u NFS), ale může být nebezpečné (např. u DNS, pak počítán jen z linkového rámce, ale např. SLIP nepočítá)

CVIČENÍ: zachytávání a inspekce UDP datagramů



TCP

- „spolehlivá služba“ – potvrzování příjmu dat a znovuzaslání ztracených a poškozených
 - pouze kontrolní součet (i když i z části IP záhlaví a dat) – lze přepočítat
 - náhodné pořadové číslo 1. bytu odesílaného toku dat (ISN) – pouze pro zaručení správného pořadí dat (a také zahazení zatoulaných segmentů z předchozího přerušeno spojení ze stejného portu)
 - **útoky**: **převzetí spojení** (**connection hijacking**, autentizovaného a dále nezabezpečeného!), **odepření služby** (**Denial of Service**, vyčerpání zdrojů systému pro spojení, maximum příznaků v záhlaví), zjišťování otevřených portů serveru (**port scanning**) a útok na aplikaci, aj.
- **řešení**: **šifrování spojení** pomocí SSL/TLS, S/MIME apod. nebo vytvořením (šifrovaných) **tunelů** na jiných portech, omezování počtu spojení za daný čas, sledování (sekvenčního) skenování portů aj.

UDP

- vyplnění kontrolního součtu je nepovinné, jinak lze přepočítat
- **musí** jej používat aplikace přenášející data na **skupinové nebo všesměrové adresy**, např. streamovaná multimedia nebo DHCP
- na směrovačích bývají povoleny porty pro **DNS** (53/udp, 53/tcp)

Firewall

- **filtrace** paketů a segmentů/datagramů na základě TCP/UDP záhlaví
- zejména „bránění“ **navázání TCP spojení nebo přenosu dat pomocí UDP na vybraných portech** (~ „blokování služeb“) = filtrování TCP segmentů s příznakem SYN (prvního při navazování spojení) a UDP datagramů na cílový port
- TCP záhlaví jen v prvním IP fragmentu – doporučené sledovat fragmenty a filtrovat i další



Překlad adres (NAT)

- překlad IP adresy odesílatele paketů z vnitřní sítě na IP adresu hraničního směrovače ve vnější síti, tzv. **maškaráda** = **překlad** adresy a **zdrojového portu** spojení/přenosu (adresy socketu) na zdrojový port nového spojení/přenosu ze směrovače (**NAPT (Network Address and Port Translation)**)
- překlad portů u transparentních proxy (typicky v DMZ nebo přímo hraniční směrovač)
 - zasahuje i do aplikační vrstvy, v případě nutnosti porozumět aplikačnímu protokolu pro překlad IP adres/portů v datech, např. FTP