

Informační a řídicí systémy I.

Úvod do počítačových sítí

Pavel Balda
ZČU v Plzni, FAV, KKY

Osnova přednášky

- n Motivace a historie
- n Vrstvení protokolů – referenční model ISO-OSI
- n Rodina protokolů TCP/IP
- n Formáty „paketů“ IP, TCP a UDP
- n Berkeley sockets
- n Síťové komunikace v C#
 - n Příklad: TCP server a klient

2

Doporučená literatura

- n Anglicky:
 - n Comer D.: **Internetworking with TCP/IP**. 4. vydání, Prentice Hall, 2000. ISBN 0-13-018380-6 *)
 - n RFC (Requests For Comments) – Detailní informace o konvencích a síťových protokolech v Internetu. Volně dostupné na webu.
- n Česky:
 - n Pužmanová R., Šmrha P.: **Propojování sítí s TCP/IP**. Kopp, České Budějovice, 1999. ISBN 80-7232-080-7 *)
 - n Osterloh H.: **TCP/IP kompletní průvodce**. Softpress s.r.o., 2003. ISBN 80-86497-34-8 *)
 - n Dostálek, Kabelová: **Velký průvodce protokoly TCP/IP a systémem DNS**. Computer Press, 1999. ISBN 80-7226-193-2 *)
 - n Jiří Peterka: Přednášky na MFF UK. Online: www.earchiv.cz

*) tituly jsou k dispozici v univerzitní knihovně

3

Motivace

- n V současnosti jsou sítě nezbytnou a neustále rostoucí infrastrukturou podniků i institucí
- n Rozvoj internetu urychluje vývoj nových technických prostředků (hardware) s cílem
 - n Zvyšování rychlosti přenosu kvůli zvyšování posílaného objemu dat
 - n Zvyšování komplexnosti sítí a jejich propojení
- n Oblast řídicích a informačních systémů sleduje trend v oblasti obecné výpočetní techniky
- n V současné době dochází k velmi rychlému zavádění síťových technologií do přímého řízení technologických procesů
 - n Různé verze deterministického ethernetu, Ethernet-Powerlink apod.
- n => Existují dobré důvody pro úvod do počítačových sítí v tomto předmětu

4

Z historie sítí a Internetu

- n 1969 – Defense Advanced Research Project Agency zřizuje projekt "packet switching network" – **ARPANET**
- n 1972 – vznik nejvýznamnější aplikace internetu – **e-mailu** (elektronické pošty)
- n 1973-79 – Základy otevřené architektury propojování sítí TCP/IP
- n 1980 – Experimentální provoz TCP/IP v ARPANETu, implementace TCP/IP pod **BSD** (Berkeley Software Distribution) **UNIX**. Používání IP adres – **IPv4**
- n 1983 – zavedení **TCP/IP** v ARPANETu, přenesení TCP/IP do komerční sféry (SUN Microsystems)
- n 1985-95 – program NSFNET (National Science Foundation NET) sponzoroval rozvoj sítě \$200 mil., první komerční služby na Internetu
- n 1992 – vznik programu **Mosaic** – předchůdce dnešních www prohlížečů
- n 1998 – ICANN (Internet Corporation for Assigned Names and Numbers) přebírá zodpovědnost za registraci doménových jmen
- n Po r. 2000 – vznik deterministických verzí Ethernetu pro řídicí systémy
- n současnost – boom Internetu

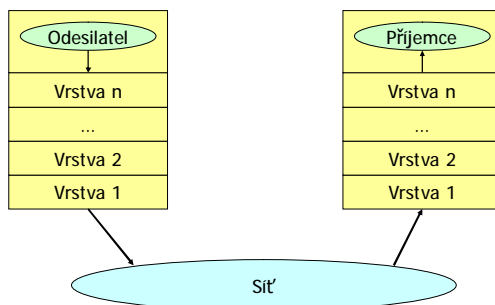
5

Architektura sítě – principy

- n **Komunikační protokol** (nebo jen protokol)
 - n Souhrn pravidel, formátů a procedur určující výměnu dat mezi dvěma komunikujícími prvky
- n **Síťová architektura**
 - n Struktura řízení komunikace, tj. souhrn řídicích činností
 - n Komunikace je složitý problém skládající se z celé řady na sebe navazujících dílčích úloh, tzv. **vrstev**
 - n Proto se hovoří o vrstevných architekturách
- n **Vrstva síťové architektury**
 - n Je definována **službou**, kterou poskytuje sousední vyšší vrstvě a **funkcí**, vykonávanou v rámci protokolu
 - n Rozčlenění komunikace do vrstev umožňuje vyměnit protokol v rámci jedné vrstvy bez změn v ostatních vrstvách
- n Síťová architektura je popsána systémem vrstev, služeb, funkcí a protokolů

6

Základní myšlenka organizace do vrstev



7

Referenční model ISO-OSI (1/2)

- n Otevřenost síťové architektury
 - n Možnost připojení koncových zařízení k síti s jednotnou architekturou
 - n Důraz na otevřenost kladen od 2. Poloviny 70. Let
- n Propojení otevřených systémů – Open Systems Interconnection (OSI)
 - n Bylo standardizováno v ISO (International Organization for Standardization), vznik normy IS 7498 v r. 1984
 - n Vznikl tzv. **Referenční model ISO-OSI** (nebo jen OSI) – abstraktní model reálného otevřeného systému
- n Norma uvádí všeobecné principy sedmivrstvé síťové architektury
 - n Účel vrstev, funkce každé vrstvy
 - n služby poskytované vyšší vrstvě, služby požadované od nižší vrstvy
 - n Norma nespecifikuje komunikační protokoly (vyžadují mnoho detailů); jsou specifikovány v navazujících normách a doporučeních

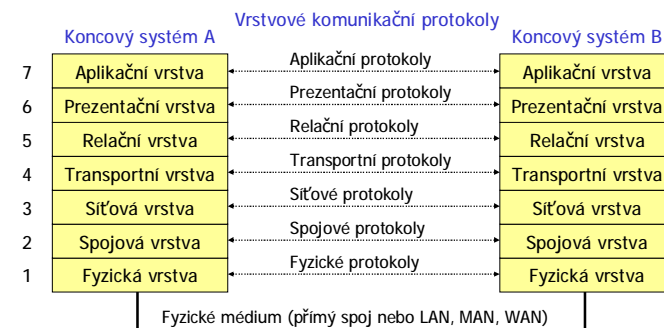
8

Referenční model ISO-OSI (2/2)

7	Aplikační vrstva (Application layer)	Zabezpečuje vlastní komunikaci mezi programy, zprostředkovává služby uživatelům. Příklad: FTP
6	Prezentační vrstva (Presentation layer)	Sjednocuje datový formát pro aplikační vrstvu, tj. provádí konverze formátu (např. znakových sad), kompresi, kódování atd.
5	Relační vrstva (Session layer)	Zabezpečuje navázání spojení se vzdáleným OS a naopak jeho ochranu před neoprávněným přístupem. Příklad: RPC
4	Transportní vrstva (Transport layer)	Přenos dat po logickém spoji, tj. segmentuje data do paketů, opakuje data a naopak odstraňuje redundanci atd. Příklad: TCP
3	Síťová vrstva (Network layer)	Správné směrování paketů po síti, tj. nalezení a indikaci cílového počítače pro spojovou vrstvu. Příklad: protokol IP
2	Spojová vrstva (Data link layer)	Přenos informace po fyzickém spoji, pracuje s vnějším obalem paketu s adresními a kontrolními informacemi
1	Fyzická vrstva (Physical layer)	Přenos signálů po fyzickém médiu (elektrický, optický), zprostředkovává vyšším vrstvám rámce (frames, pakety). Příklad: RS232C

9

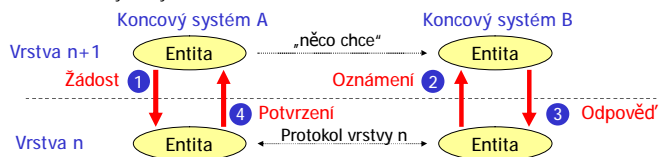
Komunikace mezi systémy v modelu ISO-OSI



10

Služby mezi jednotlivými vrstvami

- Entita je dále nedělitelný aktivní prvek (např. proces, úloha, apod.)
 - Vrstvy mohou obsahovat několik entit
 - Entity ve stejné vrstvě mohou plnit jak podobné tak i různé funkce
 - Protokol definuje pravidla komunikace mezi entitami stejnohlých vrstev
- Služby jsou vykonávány entitami pomocí primitiv:
 - Žádost** (request) – generována entitou vrstvy n+1 (žadatel)
 - Oznámení** (indication) – generuje protilehlá entita vrstvy n, upozorňuje svou entitu vrstvy n+1 na požadavek
 - Odpověď** (response) – generuje protilehlá entita vrstvy n+1
 - Potvrzení** (confirmation) – generuje entita vrstvy n; potvrzení původního požadavku entitou vrstvy n+1



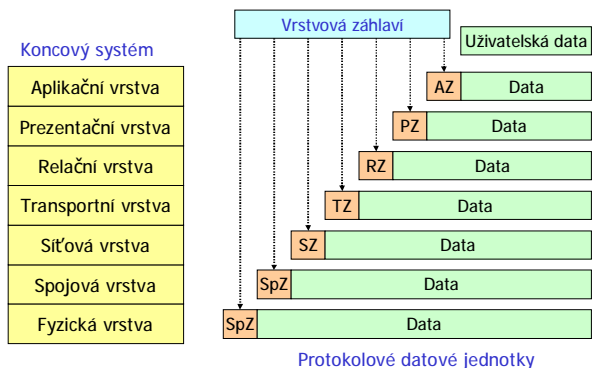
11

Protokolové datové jednotky

- Protokolová datová jednotka** (Protocol Data Unit, **PDU**) – informace přenášená jako celek mezi komunikujícími entitami. PDU se skládá z:
 - Záhlaví** (hlavičky) s protokolovou řídicí informací (Protocol control information, PCI), která může obsahovat adresu
 - Vlastních dat** (nepovinná)
- Protokolové datové jednotky vyšší vrstvy jsou zapouzdřeny do protokolových datových jednotek sousední nižší vrstvy
 - Viz obrázek na další straně
- V ISO-OSI modelu se pro první čtyři vrstvy nazývají:
 - Streams** („proudy“) – fyzická vrstva
 - Rámce** (frames) – spojová vrstva
 - Pakety** (packets) – síťová vrstva
 - Segmenty** (segments) – transportní vrstva (pozor na mnohoznačnost pojmu segment!)

12

Zapouzdření PDU



13

Rodina protokolů TCP/IP

- n Rodina protokolů TCP/IP (Transmission Control Protocol / Internet Protocol)
 - n Určena se pro komunikaci v heterogenní síti
 - n Používá ji celá řada světových sítí, souhrnně označovaná názvem Internet
 - n Základ pro celou řadu užitečných služeb, např. elektronická pošta, přenos souborů, rozsáhlé informační systémy
- n Architekturu tvoří 4 vrstvy (některé ze 7 vrstev ISO-OSI jsou sloučené):
 - n **Vrstva rozhraní sítě** – obvykle Ethernet Protocol s různým typem médií (koaxiální kabel, kroucená dvoulinka)
 - n **Vrstva mezisítě** – používá Internet Protocol (IP) a definuje způsob doručování dat mezi počítači. Zdrojový a cílový počítač nemusí být přímo spojené, proto IP zajišťuje i nalezení cesty (směrování) k cílovému počítači
 - n **Transportní vrstva** – nejčastěji používá dva základní protokoly TCP a UDP (User Datagram Protocol)
 - n **Aplikační vrstva** – obsahuje řadu známých protokolů, např. HTTP, FTP, TELNET, SNMP, DHCP, apod.

14

Model TCP/IP a ISO-OSI

OSI	Vrstva	Protokoly v TCP/IP (příklady)
7	Aplikační	NFS
6		FTP
5		RPC
4	Transportní	TCP, UDP, ICMP
3	Internet	IP, ARP
2	Network interface	Ethernet, FDDI, ATM, SLIP, X.25
1	Hardware	–

15

Hlavní protokoly architektury TCP/IP

HTTP	FTP	TELNET	SMTP	DNS	TFTP	NTP	RPC	DHCP	SNMP
RFC2616 port 80	RFC959 p.20/21	RFC854 port 23	RFC821 port 25	RFC1035 port 53	RFC1350 port 69	RFC1305 port 123	RFC1831 port 111	RFC2131 p.546/547	RFC1157 p.161/162
Border Gateway Protocol (BGP) RFC1771, číslo portu 179					Routing Information Protocol (RIP) RFC1058/1723, číslo portu 520				
Transmission Control Protocol (TCP) RFC793, číslo protokolu 6					User Datagram Protocol (UDP) RFC768, číslo protokolu 17				
Open Shortest Path First (OSPF) RFC2328, číslo protokolu 89		Internet Control Message Protocol (ICMP) RFC792, číslo protokolu 1			Internet Group Management Protocol (IGMP) RFC2236, číslo protokolu 2				
Internet Protocol (IP) RFC791									
Address Resolution Protocol (ARP) RFC826					Reverse Resolution Protocol (RARP) RFC903				
Ethernet RFC894, Token Ring RFC 1042, FDDI RFC 1188, Novell RFC1088, Arcnet RFC1201, X.25 RFC1356									

16

Seznam zkratk v předchozím obrázku

- n HTTP – HyperText Transfer Protocol
- n FTP – File Transfer Protocol
- n TELNET – standardní protokol pro vzdálené terminálové služby
- n SMTP – Simple Mail Transfer Protocol
- n DNS – Domain Name System
- n TFTP – Trivial File Transfer Protocol
- n NTP – Network Time Protocol
- n RPC – Remote Procedure Call
- n DHCP – Dynamic Host Configuration Protocol
- n SNMP – Simple Network Management Protocol
- n FDDI – Fiber Distributed Data Interface – síťová technologie založená na optických vláknech

- n Podrobnosti k jednotlivým protokolům viz doporučená literatura

17

Spojované a nespojované služby

- n **Spojované služby (connection-oriented)**
 - n Obdoba telefonního spojení
 - n Zaručeno doručení datagramů ve správném pořadí (*stream*)
 - n Využívá se protokol **TCP**
 - n + Aplikace je jednodušší, ale nemůže řídit komunikaci
 - n – Vnitřně komplikovaný protokol, proto má větší režii
- n **Nespojované služby (connectionless)**
 - n Obdoba poštovního spojení
 - n Není zaručeno pořadí ani doručení datagramů
 - n Využívá se protokol **UDP**
 - n + Aplikace může lépe řídit komunikaci
 - n + Malá režie i při výpadku některého počítače
 - n – Kontrolu konzistence musí provádět aplikace

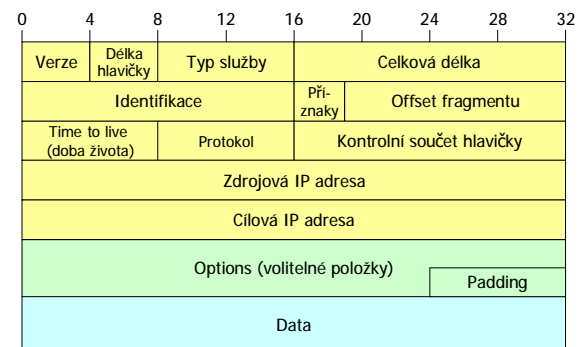
18

Typy adresování

- n **MAC adresa (Media Access Control)**, též Ethernetová adresa
 - n Šestibajtová hardwarová adresa používaná ve spojivé vrstvě, např. 00-0B-6A-9E-07-85 někdy též zapisovaná jako 00:0B:6A:9E:07:85
 - n Měla by být jedinečná na celém světě, každý výrobce má svůj 3 bajtový prefix
 - n Speciální adresa pro posílání zpráv všem (broadcast) je FF-FF-FF-FF-FF-FF
- n **IP adresa (IPv4)**
 - n Čtyřbajtová adresa přidělována poskytovatelem sítě v závislosti na topologii sítě, např. 147.228.47.16
 - n Dá se rozdělit na adresu sítě (network address) a adresu počítače (host address)
 - n Adresa sítě určuje směrování paketů mezi jednotlivými sítěmi, celá adresa určuje cílový počítač
- n **Doménová adresa**
 - n Hierarchická adresa pro „lidskou komunikaci“, např. cybernet.zcu.cz
 - n Je mnemonická, přiděluje se podle organizační struktury

19

Datagram v protokolu IPv4 (1/3)



20

Datagram v protokolu IPv4 (2/3)

- n **Verze** – Verze IP protokolu, která vygenerovala datagram. Pro IPv4 je 4
- n **Délka hlavičky** – v násobcích 32 bitů (včetně délky options a jejich doplnění na násobky 32 bitů). Nejsou-li použity optiony je rovna 5 (5*4 je 20 bajtů)
- n **Typ služby** – může určovat prioritu pro doručení
- n **Celková délka** – celková délka datagramu v bajtech (max. 65535, obvykle je však mnohem menší)
- n **Identifikace** – 16 bitová hodnota společná pro všechny fragmenty dané zprávy. Slouží ke správnému „poskládání“ zprávy u příjemce
- n **Příznaky** – řídí fragmentování dané zprávy
- n **Offset fragmentu** – dojde-li k fragmentování, udává offset daného fragmentu od začátku zprávy v násobcích 8 bajtů
- n **Time to live** – udává, jak dlouho může datagram „žít“ v počtů průchodů routery. Každý router snižuje hodnotu o 1. Když dosáhne nuly je datagram odstraněn

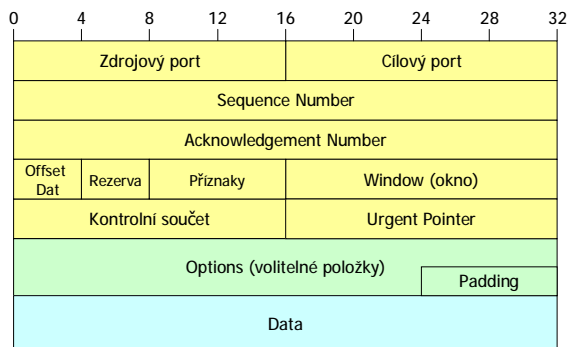
21

Datagram v protokolu IPv4 (3/3)

- n **Protokol** – kód protokolu z vyšší vrstvy. Např. 1 ... ICMP, 2 ... IGMP, 6 ... TCP, 17 ... UDP (kódy jsou uvedeny dekadicky).
- n **Kontrolní součet hlavičky** – obyčejný kontrolní součet hlavičky (ne dat!), která se rozdělí na 16 bitová slova a ta se sečtou. Pokud při předávání datagramu kontrolní součet hlavičky nesouhlasí, je datagram zahozen jako poškozený
- n **Zdrojová IP adresa** – 4 bajtová IP adresa tvůrce datagramu. Routery, kterými datagram prochází obvykle tuto adresu nemění
- n **Cílová IP adresa** – 4 bajtová IP adresa příjemce datagramu. Opět se nemění až do doručení na koncovou adresu
- n **Options** – v hlavičce datagramu mohou být přidavné informace. Jsou-li využity, je jejich délka vždy doplněna (**Padding**) na násobek 4 bajtů
- n **Data** – data přenášená v datagramu. Obsahují buď celou zprávu z vyšší vrstvy nebo její fragment

22

Struktura segmentu TCP (1/2)



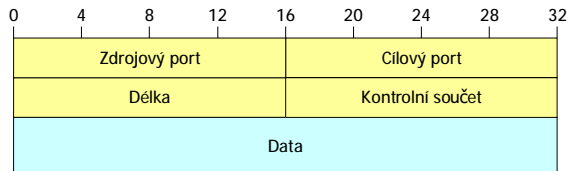
23

Struktura segmentu TCP (2/2)

- n **Port** je celé 2 bajtové číslo, které identifikuje danou aplikaci (proces) na zdrojovém nebo cílovém počítači
- n **Zdrojový port** – identifikuje vysílací port
- n **Cílový port** – identifikuje přijímací port
- n Další položky slouží pro řízení a zabezpečení bezchybného přenosu dat

24

Struktura paketu UDP



- Paket je mnohem jednodušší než u protokolu TCP, což umožňuje dosahovat s protokolem UDP větší výkon

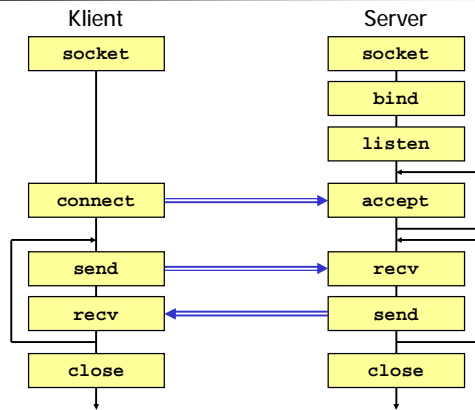
25

Programování komunikací – sockets

- **Berkeley socket API** (Application Programming Interface)
 - Též známé jako BSD (Berkeley Software Distribution) socket API
 - Toto programátorské rozhraní bylo vyvinuto na University of California v Berkeley pro usnadnění **programování síťových komunikací v jazyku C** pro systémy UNIX
 - Varianta **WinSock2** se používá i v systémech Windows

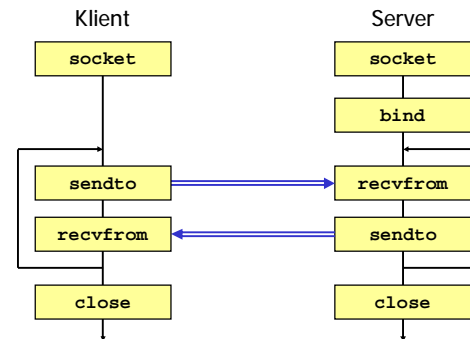
26

Architektura TCP aplikací



27

Architektura UDP aplikací



28

Síťové komunikace v C#

- n **System.Net.Sockets**
 - n Jmenný prostor poskytující implementaci Windows Sockets v řízeném kódu
 - n Podporuje i protokoly založené na IPv6
- n Základními třídami jsou:
 - n **Socket** – implementuje rozhraní Berkeley sockets
 - n Umožňuje synchronní i asynchronní přenos dat celé řady protokolů
 - n **TcpListener** – poskytuje jednoduché metody pro listen a accept, pracuje v synchronním (blokujícím) režimu
 - n **TcpClient** – poskytuje jednoduché metody pro připojení, posílání a přijímání streamu dat (TCP) do/ze sítě v synchronním (blokujícím) režimu
 - n **UdpClient** – poskytuje jednoduché metody pro posílání a přijímání datagramů UDP v synchronním (blokujícím) režimu

29

Příklad TCP serveru a klienta

- n Vychází z příkladu „A very basic TCP server written in C#“
 - n viz <http://www.codeproject.com/cs/internet/BasicTcpServer.asp>
- n Implementuje TCP server pracující asynchronně a tedy nevyužívající třídu **TcpListener**
- n Dopsán jednoduchý klient
- n Hlavní třídy jsou:
 - n **ConnectionState** – uchovává informace o stavu připojení každého klienta a využívá třídu **Socket** pro čtení a zápis dat z/do klienta
 - n **TcpServiceProvider** – abstraktní třída od které lze odvodit vlastní třídu pro rozebírání (parse) dat od klienta, jejich zpracování a zaslání výsledků
 - n **TcpServer** – třída, která řídí proces navazování spojení (accept) a spuštění příslušných metod třídy odvozené od **TcpServiceProvider**
- n Funkce serveru je demonstrována na jednoduchém serveru, vracejícím zpět klientovi poslaná data (echo).
 - n Server data kumuluje, dokud se v datech neobjeví podřetězec "<EOF>". Pak vše odešle zpět
 - n Příklad bude uveden na internetové stránce předmětu IŘS1

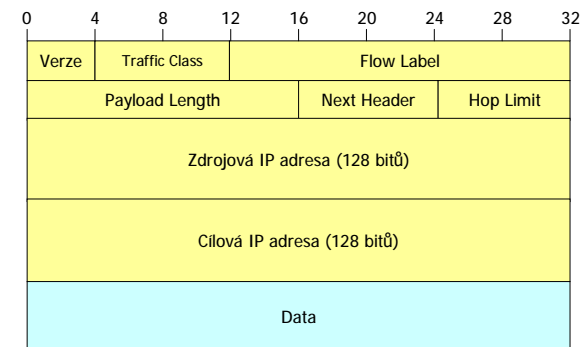
30

Adresování v protokolu IPv6

- n Teoretický počet adres v IPv4 je $2^{32} = 4\,294\,967\,296$
- n Velikost adresy v IPv6 se zvětšila na 128 bitů (16 bajtů)
- n Obdobný zápis pomocí „tečkové“ notace jako v IPv4 by byl velmi nepřehledný
- n Adresy se zapisují **hexadecimálně oddělené dvojtečkami**, např.:
 - n 805B:2D9D:DC28:0000:0000:FC57:D4C8:1FFF
 - n Úplná adresa, lze ji zkrátit (pro úsporu místa) následovně
 - n 805B:2D9D:DC28:0:0:FC57:D4C8:1FFF
 - n Je možné ještě další zkrácení vypuštěním nul a nahrazením dvěma dvojtečkami ::
 - n 805B:2D9D:DC28::FC57:D4C8:1FFF
 - n Dvojici dvojteček lze použít v zápisu jen jednou, pak se počet nul se dá dopočítat

31

Datagram v protokolu IPv6 (1/2)



32



Datagram v protokolu IPv6 (2/2)

- n **Verze** – Verze IP protokolu, která vygenerovala datagram. Pro IPv6 je 6
- n **Traffic Class** – nahrazuje Typ služby (Type of Service) v IPv4. Definuje však kvalitu služby (Quality of Service, QoS)
- n **Flow Label** – poskytuje další informace pro doručování v reálném čase, používá se např. pro video streamy
- n **Payload Length** – nahrazuje Total Length (celkovou délku) v IPv4, udává však jen délku dat v bajtech, včetně rozšiřujících (extension) hlaviček
- n **Next Header** – nahrazuje položku Protokol a používá se ve dvou významech:
 - n má-li datagram rozšiřující hlavičky, identifikuje první rozšiřující hlavičku (následuje za touto hlavičkou)
 - n Jinak specifikuje číslo protokolu podobně jako v IPv4 (pozor na rozdíl v číslování)
- n **Hop Limit** – nahrazuje položku Time to Live, název lépe odpovídá významu (počtu průchodů routery)