* Základní vlastnosti sítě:
  + Doručení dat – správnému příjemci
  + Správnost doručení – nepoškozená data
  + Včasnost doručení – doručení včas
* Základní součásti komunikačního systému:
  + Odesílatel
  + Příjemce
  + Zpráva
  + Přenosové médium – fyzické médium přenosu (optický kabel, kroucená dvoulinka, bezdrát, …)
  + Protokol – sada pravidel řídících komunikaci
* Základní parametry síťových toků:
  + Propustnost (bandwidth) – kapacita přenosového kanálu (v b/s)
  + Ztrátovost packetů (packet loss) – průměrný počet ztracených packetů za určitý čas v % vzhledem k celkovému počtu přenesených packetů
  + Zpoždění přenosu (delay, latency) – čas od odeslání, po přijetí u příjemce (v ms)
    - Uvádí se i RTT (Round Trip Time delay) – čas od odeslání k cíli, přijetí u cíle, odeslání zpět původnímu odesilateli a přijetí u původního odesilatele
  + Rozptyl/kolísání zpoždění (jitter) – variabilita v doručování packetů cíli (v ms)

**Ideální sítě**

* Transparentní pro uživatele/aplikace
  + Pouze end-to-end vlastnosti
* Neomezená propustnost
* Žádné ztráty
* Žádné zpoždění a rozptyl zpoždění
* Zachovává pořadí packetů
* Data nemohou být poškozena

**Skutečné sítě**

* Vnitřní struktura ovlivňuje doručení dat
* Omezená propustnost
* Dochází ke ztrátám
* Poskytuje variabilní zpoždění a rozptyl zpoždění
* Nezachovává pořadí packetů
* Data mohou být poškozena

**Požadované vlastnosti sítí**

* Efektivita
* Spravedlivost – stejný přístup ke všem datovým tokům všech uživatelů se stejnou prioritou
* Decentralizovaná správa
* Rychlá konvergence při adaptaci na nový stav
* Multiplexing/demultiplexing
* Spolehlivost
* Řízení toku dat – ochrana proti zahlcení sítě a přijímajícího uzlu

**Spojované sítě** (přepínání okruhů)

* Před začátkem přenosu je ustaveno spojení, které je udržováno během celé komunikace
* Informace o spojení udržovány sítí – síť uchovává stav
* Okruh může být pevný, nebo vytvářen na žádost
* Jednoduchá implementace kvality služby
* Např. analogové telefonní sítě

**Nespojované sítě** (přepínání packetů)

* Pro přenos není využita definovaná cesta – data jsou rozděleny do packetů a vyslány do sítě
  + Packety mohou být směrovány různými cestami, slučovány a fragmentovány
* U příjemce jsou z packetů vytáhnuty data, které jsou poté složeny do původní podoby
* Není potřeba uchovávat stav v síti
* Problematická implementace QoS (best-effort služba, neinformuje o doručení zprávy)
* Např. internet

**Implementace funkcionality**

* End-to-End
  + Aplikací požadovanou funkcionalitu lze zajistit jen se znalostí a prostřednictvím samotné aplikace
    - Pokud možno, operace komunikačního protokolu by měly být definovány tak, aby byly prováděny v koncových bodech komunikačního systému, nebo co nejblíže k nim
    - V nižších vrstvách systému mají být funkce protokolu implementovány jen tehdy, když to zlepšuje výkon
  + Vhodný pro aplikace vyžadující vysoký stupeň věrnosti přenesených dat a současně tolerují zpoždění (např. email)
* Hop-by-Hop
  + Opakováním určité funkcionality na úrovní každého dvoubodového přenosu lze dosáhnout výrazného zvýšení výkonu
  + Vyžaduje se uchovávání stavových informací na vnitřních prvcích sítě -> limitovaná škálovatelnost
  + Vhodný pro aplikace, kde minimalizace zpoždění je důležitější, než věrnost přenesených dat (real-time aplikace)

**ISO/OSI model**

* 7-vrstvý model navržený za účelem zajištění kompability komunikačních systémů různých výrobců
* Každá vrstva je zodpovědná za určitou definovanou funkcionalitu
  + Aby mohla funkcionalitu zařídit, přidává do přenášených dat své řídící informace
* Každá vrstva komunikuje jen se svými přímo sousedícími vrstvami
  + Využívá služeb nižší vrstvy, poskytuje služby vyšší vrstvě
  + Funkcionalita je izolována v rámci vrstvy – pokud dojde ke změně vrstvy, musí se upravit jen sousedící vrstvy
* Z logického pohledu se komunikace odehrává pouze mezi stejnými vrstvami
* 7 vrstev nebylo široce akceptováno -> TCP/IP model
* Vrstvy:
  + 1. fyzická (bity) – přenosová média, signály, bitová reprezentace
  + 2. datového spoje (rámce) – MAC a LLC (fyzický adresace), kromě hlavičky datům přidává i tail
  + 3. síťová (packety) – síťové (logické) adresování, směrování
  + 4. transportní (segmenty (TCP), datagramy (UDP)) – komunikace procesů, spolehlivost
  + 5. relační (data) – relace, správa relací
  + 6. prezentační (data) – datová reprezentace
  + 7. aplikační (data) – síťové aplikace
* Aplikační vrstva
  + Představuje rozhraní mezi uživatelem a sítí
  + Zahrnuje síťové aplikace a síťové protokoly
* Prezentační vrstva
  + Zajišťuje jednotnou reprezentaci dat na obou stranách
  + V rámci TCP/IP se předpokládá, že je zajištěno samotnou aplikací
* Relační vrstva
  + Spravuje ustavená spojení (relace) mezi komunikujícími aplikacemi
  + V rámci TCP/IP se předpokládá, že je zajištěno samotnou aplikací, resp. Aplikačním protokolem
* Transportní vrstva
  + Zajišťuje identifikaci (adresaci) a doručení dat (segmentů, datagramů) mezi procesy
* Síťová vrstva
  + Zajišťuje identifikaci (adresaci) a doručení dat (packetů) mezi uzly
  + Součástí je nalezení vhodné cesty mezi uzly – směrování
* Vrstva datového spoje
  + Zajišťuje přenos dat (rámců) mezi uzly propojenými přenosovým médiem
* Fyzická vrstva
  + Řídí děje v přenosovém médiu
  + Rozhoduje o vysílání/příjmu přenášených dat (bitů), kódování dat do signálů, atd.

**TCP/IP**

* Aplikační vrstva = aplikační + prezentační + relační
* Transportní vrstva = transportní
* Síťová vrstva = síťová
* Vrstva přístupu k síti = datového spoje + fyzická
* Nezávisle na využitých protokolech a technologiích přenosu je pro identifikací uzlů vždy využit IP protokol

**Síťový protokol**

* Síťový protokol definuje formát a pořadí zpráv zasílaných mezi entitami, stejně jako akce vykonané při odeslání/příjmu zpráv
* Určuje co je předmětem komunikace, jak komunikace probíhá a kdy probíhá
* Definuje
  + Syntaxi = strukturu/formát dat
  + Sémantiku = význam každé sekce bitů
  + Časování = kdy je potřeba zaslat kterou zprávu
* Příklady síťových protokolů
  + UDP, TCP, IP, IPv6, SSL, TLS, HTTP, FTP, SSH, Aloha, …

**Standardizace**

* Stanovení norem/standardů popisujících různé akce, činnosti, formy či způsoby komunikace, atd.
* Hlavní cíle
  + Kvalita
  + Bezpečnost
  + Kompatibilita
  + Interoperabilita
  + Portabilita
* Nejznámější standardizační instituce
  + ISO, IEEE, ANSI, ITU-T, IEC

**Fyzická vrstva L1**

* Point-to-point spoje, bez možnosti adresace stanic
* Poskytuje funkcionalitu pro práci s přenosovým médiem
* Poskytuje služby pro vrstvu datového spoje
  + Předávají si data vyjádřená posloupností 0 a 1, seskupená do rámců
  + Fyzická vrstva transformuje bity do signálů
* Řídí děje v přenosovém médiu, rozhoduje o:
  + Vysílání/příjmu dat
  + Kódování dat do signálů
  + Počtu logických kanálů přenášejících data z různých zdrojů souběžně
* Hlavní cíl = zajistit přenos bitů mezi odesílatelem a příjemcem
* Standardy (RS-232-C, CCITT V.24, IEEE 802.x, …) definují elektrické, mechanické, funkční a procedurální vlastnosti rozhraní pro připojení různých přenosových prostředků a zařízení
* Časování bitů
* Multiplexing – dělení fyzického média do logických kanálů pro vyšší efektivitu
* Switchování okruhů
* Signály šířené přenosovým médiem – analogové nebo digitální
* Defekty signálů
  + Útlum (slábnutí) – ztráta energie, např odporem média
  + Zkreslení – ztráta tvaru, způsobeno rozdílnou rychlostí šíření signálů na různých frekvencích
  + Šum – vliv venkovní energie
* Analogový signál se moduluje, digitální signál se transformuje
* Analogový přenos – modulace signálu
  + Využívá se modem (MOdulátor/DEModulátor)
  + Amplitudová digitální modulace – výška vln značí 0/1
  + Frekvenční digitální modulace – frekvence vln značí 0/1 1 | 0 | 0
  + Fázová digitální modulace – změna 0/1, podle změny směru vlny /\/\ \/\/\/\/
* Digitální přenos – kódování
  + Konverze binárních dat do digitálního signálu
  + Problém synchronizace vysílače a přijímače – hodiny mohou ztrati synchronizaci na dlouhých posloupnostech 0/1
  + Přímé kódování – 1 = kladná hodnota amplitudy, 0 = nulová hodnota amplitudy
  + NRZ-L – 1 = záporná, 0 = kladná
  + NRZ-I – 1 = změna polarizace amplitudy, 0 = žádná změna
  + Manchester – 1 = S, 0 = Z
  + 4B/5B – možnost detekce chyb

**Linková vrstva L2**

* Proč nestačí L1:
  + Nezajišťuje opakování chybně přenesené informace
  + Nepodporuje určení entity mající právo vysílat do média
  + Nepodporuje ovládání toku dat ze zdroje do média
  + Nepodporuje komunikaci mezi definovanými partnery
* Lokální sítě – LAN
* Přenosové médium sdíleno více stanicemi
* Node-to-node delivery
* Přijímá packety od síťové vrstvy, které transformuje do rámců
* Spolu s fyzickou vrstvou zajišťuje přenos rámců mezi uzly propojenými přenosovým médiem
* Zaručuje spolehlivost přenosu
* Zajišťuje, aby cílový uzel nebyl zahlcován proudícím tokem dat
* Řídí přístup uzlů ke sdílenému přenosovému médiu
* Tvorba rámců
* Adresování – fyzická/MAC adresa, rámce obsahují zdrojovou a cílovou fyzickou adresu entit
* Chybové řízení – chyby ve fyzické vrstvě nelze zcela eliminovat, L2 zajišťuje detekci a korekci chyb
* Řízení toku – zabraňuje zahlcení příjemce, mechanismy stop-and-wait, sliding-window
* Řízení přístupu k médiu (Medium Access Control – MAC) – eliminuje kolize způsobené násobným vysíláním
* Chyby: optická vlákna 10-12, wireless 10-5
* Linková vrstva provádí detekci/korekci chyb
  + Vysílač přidá bity, jejichž hodnota je funkcí přenášených dat
  + Přijímač spočte stejnou funkci a při porovnání detekuje případnou chybu
* Automatic Request for Retransmission (ARQ)
  + Detekce chyby a zajištění opakování přenosu
  + Vhodné pro málo chybující přenosová média
* Forward Error Correction (FEC)
  + Detekce chyb a snaha o jejich korekci
  + Vhodné pro často chybující přenosová média či média s velkou latencí
  + Např. Hammingův kód
* Kódy pro detekci chyb
  + Sudá/lichá parita – sudý/lichý počet 1, detekuje pouze 1 chybu, slabý nástroj pro detekci chyb
  + Cyclic Redundancy Check (CRC) – garantuje silnou kontrolu (možná detekce všech jednobitových, dvoubitových a většiny dávkových chyb)
* Řízení přistupu k médiu (MAC)
  + Funkcionalita odpovědná za koordinaci přístupu více stanic ke sdílenému přenosovému médiu
  + Cíl: eliminace kolizí při vysílání
  + Protokoly řízení přístupu
    - Protokoly neřízeného přístupu
    - Protokoly řízeného přístupu
    - Protokoly multiplexově-orientovaného přístupu
      * Zpřístupnění fyzické vrstvy vrstvě L2
      * Frequency-Division Media Access, Time-Division Media Access
* Protokoly neřízeného přístupu
  + Aloha
    - Stanice vysílá kdykoliv má připravený rámec
    - Kolize detekovány nepřijetím potvrzení o přijetí v definovaném čase
    - Po kolizi náhodnou dobu vyčká a zkusí vysílat znovu
    - Neefektivní
  + CSMA/CD
    - Upravená Aloha – stanice vysílá jen když zjistí klid v médiu
    - Současně na médiu naslouchá pro detekci případné kolize (CD = Collision Detection)
    - Aplikace v klasickém LAN Ethernetu, nepoužitelné v nevoděném médiu
  + CSMA/CA
    - Obcházení kolizí
    - Použitelné v nevoděném médiu
* Řízený přístup
  + Stanice smí vysílat jen tehdy, když k tomu získá právo od řídící/jiné stanice
  + Rezervace – vysílání v předem domluvených vyhrazených intervalech
  + Vyzývání – centrální stanice vyzývá stanici, která bude vysílat
  + Předávání příznaku – předávání peška indikujícího právo k vysílání
* Lokální počítačové sítě
  + Systematická topologie pro jednoduché sítě
  + Rozlehlejší sítě tvořeny vzájemným propojováním jednoduchých topologií
* Kolizní doména
  + Určena stanicemi sdílejícími přenosové médium
  + Kdykoliv začne v kolizní doméně více stanic vysílat, dojde ke kolizi -> nutnost opakování přenosu
* Sběrnicová topologie
  + Jednoduše instalovatelná
  + Kolizní doména tvořena všemi stanicemi
  + CSMA/CD
  + Náchylná k defektům
* Kruhová topologie
  + Všechny zprávy putují v jednom směru
  + Kolizní doména tvořena všema stanicemi
  + Právo vysílat určuje metoda peška
  + Náchylná k defektům
* Hvězdicová topologie
  + Centrální propojovací bod (hub, bridge, switch)
  + Hůře instalovatelná
  + Kolizní doména v závislosti na propojovacím bodu
    - Hub – L1 – kolizní doména tvořena všemi stanicemi
    - Bridge, switch – L2 – kolizní doména vždy tvořena dvěma sousedícími stanicemi
  + Nepříliš náchylná k defektům
* Můstek (bridge)
  + Transparentní propojení sítí
  + Všechen provoz prochází můstkem
  + Odděluje sdílená média (kolize se nepřenáší)
  + Může mít více jak dvě připojení
  + Přepínač (switch) – víceportový můstek
* Založeno na MAC adresách
  + Backward Learning Algorithm – můstek se učí umístění stanic nasloucháním na médiu
  + Rámce se směrují dle cílové adresy
* Lze vytvořit sítě s cykly – distribuovaný Spanning Tree Algorithm pro výpočet kostry
* Nevhodné pro velké sítě – rostoucí přepínací tabulky
* Distribuovaný Spanning Tree Algorithm
  + Cíl algoritmu: nepoužívat některé porty můstků (zabránit cyklům)
  + Každý můstek posílá periodické zprávy
  + Když dostane zprávu od souseda, upraví definici nejlepší cesty
    - Preferuje kořen s menší adresou
    - Preferuje menší vzdálenosti
    - Při stejných vzdálenostech preferuje nižší adresu
  + Mechanismus
    - Volba kořenového vrcholu stromu (nejnižší adresa)
    - Postupný růst stromu – nejkratší vzdálenost od kořene (preferuje uzly s nižší adresou)
    - Nalezené nejlepší cesty definují aktivní porty můstků
    - Všechny ostatní porty vypnout
  + Fáze výběru kořenového můstku
    - Po zapnutí všechny můstky prohlásí, že jsou kořenem
    - Každý z nich zašle konfigurační informaci na všechny porty
    - Na základě těchto informací je zvolen kořenový můstek s nejnižší ID
  + Fáze výběru kořenových portů
    - Každý můstek si za svůj kořenový port zvolí ten s nejnižší cenou cesty k Root Bridge
    - Mají-li dva porty stejnou cenu, je zvolen ten s nižším Port ID, druhý se vypne
  + Fáze výběru (ne)aktivních portů
    - Root Bridge nastaví všechny svoje porty jako aktivní
    - Na všech spojích, na kterých nejsou Root Porty, si přepínače vyměňují informace a zjišťují, kdo z nich má nižší Bridge ID, ten potom nastaví svůj port jako aktivní, druhý s vyšším Bridge ID svůj port vypne

**Síťová vrstva L3**

* Proč nestačí L2
  + Nemožnost vybudování geograficky libovolně rozlehlé sítě
  + Neuniformní prostředí
* Propojování lokálních sítí do větších, komplexních sítí (např. internet)
* Možnost ustavení komunikačního kanálu mezi libovolnými stanicemi v Internetu skrze více samostatných fyzických sítí (LANs), tzv. host-to-host delivery
* Poskytuje služby pro transportní vrstvu
  + Přijímá segmenty od transportní vrstvy, které transformuje do packetů
  + Ve spolupráci s vrstvou datového spoje zajišťuje přenos packetů mezi uzly (i mezi různými LAN sítěmi)
* Logicky spojuje samostatné heterogenní LAN sítě
  + Vyšším vrstvám poskytuje iluzi uniformního prostředí jediné velké sítě
* Poskytuje možnost jednoznačné identifikace (adresace) každého zařízení na internetu
* Zajišťuje směrování procházejících packetů
* Spolu s L2 mapuje adresy síťové vrstvy na MAC adresy
* Propojování fyzických sítí (internetworking) – iluze uniformního prostředí jediné velké sítě
* Tvorba packetů – přijaté segmenty transformovány na packety (IP protokol)
* Fragmentace packetů – rozdělování segmentů na packety s délkou závislou na vlastnostech/schopnostech sítě
* Adresace
  + Adresy entit síťové vrstvy – IP adresy, jedinečné skrze celou síť
  + Packety obsahují zdrojovou a cílovou IP adresu entit
* Mapování IP adres na/z fyzické adresy – ARP, RARP protokoly
* Směrování (routing) – nalezení nejvhodnější cesty mezi entitami, reakce na chyby
* Metody základního monitoringu stavu sítě – základní informace o nedoručitelnosti packetů, stavu sítě, uzlů, atp. – ICMP protokol
* Propojování sítí (internetworking)
  + Vzájemné propojování celých sítí i jednotlivých kabelových segmentů (hierarchie)
  + Propojením vzniká internetwork, zkráceně internet
  + Důvody pro internetworking
    - Překonání technických omezení/překážek – např. omezený dosah kabelových segmentů
    - Optimalizace fungování sítě – snaha regulovat tok dat, zamezení zbytečného šíření provozu
    - Zpřístupnění vzdálených zdrojů – přístup ke vzdáleným serverům
    - Zvětšení dosahu poskytovaných služeb – email, internetové volání, …
* Adresace
  + Požadavek jednoznačné identifikace každého zařízení připojeného k internetu
  + Nutnost systematického přidělování adres za účelem snadnějšího směrování
  + Každému zařízení/rozhraní přiřazena IP adresa (IPv4 32 bitů vs IPv6 128 bitů)
* IPv4 - typy adres
  + Individuální (unicast) adresy – identifikace jednoho síťového rozhraní
  + Broadcast adresy – slouží pro zasílání dat všem možným příjemcům na dané LAN, zdrojová adresa datagramu (identifikace odesílatele) je unicastová
  + Skupinové (multicast) adresy – slouží pro adresování skupiny příjemců, kteří o data projevili zájem, zdrojová adresa datagramu je unicastová
* Přidělování adres – Classful Addressing
  + Zcela první metoda přidělování adres
  + Adresní prostor rozdělen do 5 tříd A-E
  + Adresa sítě (NetID)
    - Identifikuje danou síť (nemůže být přidělena uzlu/rozhraní)
    - Lze využít pro směrování
  + Adresa uzlu/rozhraní (HostID)
    - Identifikuje jedinečný uzel v síti NetID
  + Problémy
    - Nedostatečná granularita – každá třída rozdělena na pevný počet sítí s pevným počtem adres
    - Přidělování více síťových adres menší třídy generuje nárůst směrovacích tabulek
  + Řešení
    - Dělení sítě do menších podsítí – subnetting
    - Využití toho, že organizace má přidělen souvislý blok adres a snižování velikosti směrovacích tabulek – supernetting
  + Subnetting
    - Standardní IP poskytuje dvouúrovňovou hierarchii
    - Subnetting zavádí možnost tříúrovňové hierarchie
      * Adresa sítě, adresa podsítě, adresa uzlu
    - Princip uzavřenosti – zvenčí se jeví jako síť, podsítě se rozlišují až na hraničním směrovači
  + Supernetting
    - Posouvá pomyslnou dělící čáru mezi oběma slozkami IP adresy směrem k vyšším bitům
    - Spojuje (agreguje) několik původně samostatných síťových IP adres v jednu výslednou
    - Musí jít o sousední síťové adresy – musí se shodovat v určitém počtu vyšších bitů své síťové části a vyčerpávat všechny bitové kombinace v příslušném počtu nižších bitů své síťové části
  + Maska sítě/podsítě
    - Oba způsoby -nettingu vyžadují mechanismus pro identifikaci bitů, které identifikují síť
      * U subnettingu nezbytné jen na hraničních směrovačích
      * U supernettingu nezbytné na všech směrovačích
    - Využitý mechanismus – maska sítě
      * 32-bitový řetězec (v rámci IPv4)
      * Obsahuje 1 v těch bitech, které odpovídají síťové části adresy, 0 u relativní adresy uzlu
      * IP uzlu && maska sítě = adresa sítě
* Přidělování adres – Classless Addressing
  + Zobecnění a rozšíření sub/super-nettingu
  + Zavádí zcela variabilní délku bloku adresy sítě
  + Adresy se přidělují hierarchicky
  + Opodstatnění subnettingu zůstává
  + Classless Inter-Domain Routing (CIDR)
    - Konvence popisující pravidla – použití IP adres, významu masek, supernetting a subnetting
    - Nahrazuje původní třídní charakter IP adres, IP přidělovány po CIDR blocích
    - Velikost CIDR bloku dána příslušnou maskou – možno velmi pružně přizpůsobovat
    - Snížení tempa vyčerpávání adresového prostoru
    - Důsledek CIDRu – adresy závislé na poskytovateli
    - Původně IP nezávislé na způsobu jejich připojení
    - Zavedení závislosti
      * Poskytovatel získává CIDR blok, který rozděluje dle svého uvážení
      * Při změně poskytovatele je potřeba síť přeadresovat
* Network Address Translation (NAT)
  + Mechanismus pro snížení tempa vyčerpávání adresového prostoru
  + Určeno zejména pro domácí uživatele
  + Původně připojováni modemy -> možnost dynamického přidělování adres
  + Nyní ADSL, kabelová připojení – (většinou) trvalá alokace adres
  + Časté požadavky na přidělení více IP adres
  + Řešení – NAT
    - Skrývání vnitřní sítě za jednu/několik externích adres
    - V rámci vnitřní sítě možnost využít mnoho interních adres
  + Překlad adres odchozích packetů je triviální
  + Překlad adres příchozích packetů vyžaduje dodatečné informace
    - Překladové tabulky
* IPv6
  + Řešení nedostatku IP adres
  + Lze zkracovat zápis vynecháváním úvodních nul skupin
  + Lze vynechat jednu sekvenci po sobě jdoucích nulových skupin
  + Cílem usnadnění směrování
  + IPv6 adresace je pouze classless, třídy neexistují
  + Sítě popisovány s využitím notace CIDR
* IPv6 typy adres
  + Individuální (unicast) adresy – totéž co v IPv4, identifikace jednoho síťového rozhraní
  + Skupinové (multicast) adresy – totéž co v IPv4, adresování skupin zařízení
  + Výběrové (anycast) adresy – také označují skupinu příjemců, data se doručí jen jedinému členovi (nejbližšímu)
* Interakce L3 s L2
  + Mechanismus doručení dat v IP sítích – hop-by-hop
  + Doručení zprávy na základě MAC adres
  + Příjemce na stejné LAN jako odesílatel
    - IP datagram obsahuje IP příjemce, rámec L2, MAC adresu příjemce
  + Příjemce na jiné LAN než odesílatel
    - IP datagram obsahuje IP příjemce, rámec L2, MAC adresu směrovače
    - Směrovač po přijetí datagramu jej vloží do nového rámce s MAC adresou dalšího směrovače ve snaze přiblížit se cíli
  + Nutnost mapování IP adres na MAC adresy
    - Statické mapování
      * Vytvoření statické tabulky párů (IP, MAC)
      * Obtížně spravovatelné
    - Dynamické mapování
      * Address Resolution Protocol (ARP)
* ARP protokol
  + Protokol pro zjištění MAC adresy uzlu/směrovače na základě IP adresy
  + Mechanismus
    - Zaslání ARP request packetu všem uzlům na dané LAN (broadcast)
      * Packet obsahuje IP a MAC odesílatele a IP hledaného uzlu
    - Packet zpracován všemi uzly, odpoví jen ten, jehož IP se shoduje s hledanou
  + ARP packety baleny přímo do rámců L2
  + Protokol RARP (Reverse Address Resolution Protocol)
    - Zpětný překlad MAC adres na IP, již se nevyužívá
* IP protokol
  + Nejrozšířenější protokol síťové vrstvy
  + Doprava dat na místo jejich určení i přes mezilehlé uzly (směrovače) – host-to-host delivery
  + Uzly v rámci IP protokolu jednoznačně identifikovány IP adresami
  + Poskytuje nespolehlivou (best-effort) službu
  + Doplněn dalšími podpůrnými protokoly (ICMP, ARP, RARP, IGMP)
    - Ošetření nestandardních situací, šíření informací potřebných ke korektnímu směrování, identifikace rozhraní na LAN, atd.
  + Navržen a standardizován ve dvou verzích
    - IPv4 – 1981
    - IPv6 – 1998
  + IPv4 datagram obsahuje
    - Verzi IP protokolu
    - Délku hlavičky IP datagramu
    - Typ služby
    - Celkovou délku
    - Identifikaci, flagy, offset
    - Time to live (TTL) – řízení macimálního počtu skoků navštívených datagramem
    - Identifikace protokolu vyšší vrstvy využívajícího služeb IP vrstvy
    - Kontrolní součet hlavičky IP datagramu
    - Zdrojová a cílová IP adresa
    - Options
    - Data
  + IPv4 fragmentace datagramů
    - Ne všechny sítě mohou přenášet data stejné velikosti
    - Maximum Transfer Unit (MTU) – maximální velikost dat, které lze přenést využitým L2 protokolem, určuje maximální velikost přenositelného IP datagramu
    - Je-li datagram větší, než MTU výstupní linky, je rozdělen na několik menších datagramů – fragmentů
    - Každý fragment má svou IP hlavičku
    - Složení fragmentů do původního datagramu vyžaduje
      * Identifikaci datagramu, kterému fragmenty náleží
      * Znalost počtu fragmentů
      * Znalost pozice každého fragmentu v původním datagramu
    - Fragmentace se provádí na zdrojovém uzlu, nebo na směrovačích
    - Skládání fragmentů probíhá pouze na cílovém uzlu
    - Ztráta fragmentu = ztráta datagramu
    - Lze provádět vícenásobnou fragmentaci
  + Internet Control Message Protocol (ICMP)
    - Doprovodný protokol IP protokolu
    - Poskytuje informace o chybách při přenosu IP datagramů
    - Poskytuje základní informace o stavu sítě
    - Oznamy o chybách
      * Destination unreachable – destination může být protokol, port, uzel nebo celá síť
      * Time exceeded – informace o vypršení TTL či o vypršení času pro znovusložení fragmentů datagramu
    - Dotazy na stav sítě/uzlu
    - Zprávy obsahují část packetu, který způsobil chybu/váže se na něj odpověď
    - Využití ICMP v aplikacích
      * Ping – využití ICMP Echo request/reply
      * Traceroute – využití ICMP Time exceeded
    - ICMP packet není generován jako reakce na
      * ICMP chybu
      * Broadcast nebo multicast zprávu
      * Poškozenou IP hlavičku
      * Chybu fragmentu (kromě prvního)
  + IPv6
    - Další důvody zavedení
      * Slabá podpora přenosů aplikací reálného času
      * Žádná podpora zabezpečení komunikace na úrovni IP
      * Žádná podpora autokonfigurace zařízení
      * Žádná podpora mobility
    - Rozšířený adresní prostor
    - Jednodušší formát hlavičky
    - Možnosti dalšího rozšíření hlavičky
    - Podpora přenosů reálného času
    - Podpora zabezpečení přenosu
    - Podpora mobility
    - Podpora autokonfigurace zařízení
  + IPv6 datagram obsahuje
    - Verze IP protokolu
    - Priorita datagramu
    - Flow label
    - Celková délka datagramu bez základní hlavičky
    - Next header
    - Hop limit (TTL)
    - Zdrojo a cílová adresa
  + Podpora zabezpečených přenosů
    - Implementace zabezpečené komunikace na síťové vrstvě
    - Označováno jako IPSec
    - V IPv6 povinná
    - Autentizace dat
      * ověřuje, že data odeslal ten, kdo to o sobě tvrdí
      * zaručuje, že obsah datagramu nebyl po cestě sítí změněn
    - šifrování dat
      * umožňuje utajit obsah korespondence
    - dvě rozšiřující hlavičky
      * Authetication Header (AH) – autentizace datagramu, možnost ochrany před opakovaným vysíláním téhož
      * Encapsulating Security Payload (ESP) – autentizace datagramu + možnost šifrování obsahu, ale ne současně
    - Dva režimy ochrany
      * Transportní režim – bezpečnostní hlavičky se vkládají přímo jako součást datagramu mezi rozšiřující hlavičky
      * Tunelovací režim – celý diagram se zabalí jako data do nového datagramu, opatřeného novými hlavičkami, včetně bezpečnostních
    - Bezpečnostní asociace (Security Association, SA)
      * Virtuální spojení dvou počítačů, zajišťující zabezpečený přenos dat
      * Součástí jsou všechny potřebné informace
      * Jsou jednosměrné
  + Podpora mobility
    - Domácí adresa – neměnná adresa, pod níž je stroj trvale dostupný
    - Dočasná adresa – měnící se adresa, závislá na síti, kde se zařízení nachází
    - Domácí agent
      * směrovač v domácí síti, jehož prostřednictvím je zařízení trvale dosažitelné
      * stahuje na sebe datagramy směřující k mobilnímu uzlu a předává je tunelem
    - optimalizace cesty
      * seznámení vzdálené strany s aktuální dočasnou adresou mobilního uzlu
      * cílem je zefektivnění komunikace
      * není nezbytná
  + podpora autokonfigurace
    - stavová konfigurace – základem server spravující konfigurační parametry, které na požádání sděluje klientům
    - bezstavová konfigurace
      * zcela nový způsob konfigurace IPv6 klientů
      * v síti sídlí směrovače, které vědí vše potřebné
      * občas sdělí, jaká je situace v síti
      * nově příchozí klient čeká na ohlášení, nebo si o něj vyžádá
      * na základě ohlášení si vypočte vlastní IPv6 (prefix + L2 adresa)
      * nutné doplnit mechanismem pro oznamy lokálních DNS serverů
  + fragmentace datagramů
    - stejný mechanismus jako u IPv4
    - rozdíl: vnitřní uzly nesmí fragmentovat -> nutnost zjištění maximální velikosti packetů
    - problém s dynamickými cestami
  + podpůrné protokoly
    - ICMPv6
      * Založen na stejných principech jako ICMPv4
      * Navíc zahrnuje funkcionalitu ARP a IGMP
  + Přechod IPv4 -> IPv6
    - Dvojí zásobník – zařízení podporuje IPv4 i IPv6
    - Tunelování – IPv6 datagramy zabaleny jako data do IPv4 datagramu
    - Translátory – zařízení pro překlad IPv6 datagramů do IPv4 datagramů a naopak
* Směrování (routing)
  + Proces nalezení cesty mezi dvěma uzly
  + Ovlivňující faktor
    - Statické – topologie sítě
    - Dynamické – zátěž sítě
  + Na směrování lze nahlížet jako na problém teorie grafů
  + Linky můžou být ohodnoceny podle
    - Rovnoměrně
    - Hodnoty kapacity
    - Zpoždění
    - Využití (zátěž)
    - Reálné ceny za použití
  + Zpravidla se nezabývá celou cestou
    - Směrovač řeší jen jeden krok – kam packet předat dál
  + Směrování
    - Společná činnost směrovačů
    - Proces nalezení/vytváření a údržby směrovacích tabulek
  + Zasílání (forwarding)
    - Lokální proces – každý směrovač samostatně
    - Vyžaduje přístup ke směrovací tabulce
  + Ve směrovací tabulce se vyhledává nejdelší prefix, který vyhovuje požadavku
  + Globální znalost topologie celé sítě je problematické
    - Složité získat
    - Neaktuální
  + Statické
    - Adminem editované záznamy
    - Směrovač neumí vytvářet alternativní cesty
    - Jednodušší, málo flexibilní
    - Vhodné pro statickou topologii
  + Dynamické
    - Reaguje na změny v síti
    - Složité algoritmy
    - Pravidelné aktualizace směrovacích tabulek
    - Nezaručuje pořadí doručení
* distribuované směrování
  + distance vector (DV)
    - Bellman-Ford algoritmus
    - Sousední směrovače si v pravidelných intervalech či při topologické změně vyměňují kompletní kopie svých směrovacích tabulek
    - Podle toho si doplňují nové informace a inkrementují své distance vector číslo (udává počet hopů k dané síti)
    - Všechny informace jen svým sousedům
    - Směrovač nesděluje cestu zpět uzlu, od kterého se o ní dozvěděl
    - Využívá protokol RIP – používá CIDR, metrika je počet hopů, směrovače zasílají informaci každých 30 sekund
  + Link state (LS)
    - Dijkstrův algoritmus
    - Směrovače si zasílají pouze informace o stavu linek, na něž jsou bezprostředně připojeny
    - Udržují si kompletní informace o topologii dané sítě
    - Pak se počítá nejkratší cesta
    - Informace o svých sousedech všem
    - Využívá protokol OSPF (Open Shortest Path First) – metrika = cena, autentizace zpráv
* LS v DV
  + LS
    - Složitost vyšší
    - Rychlost konvergence nižší, trpí na oscilace
    - Vhodné i pro rozsáhlé sítě
  + DV
    - Složitost nižší
    - Může konvergovat pomaleji než LS, problém s cykly, count-to-infinity problém
    - Vhodné jen pro menší sítě
* Autonomní systémy (AS)
  + Rozšíření hierarchického členění sítě
  + V rámci AS může každý zajistit přenos a aktualizaci směrovacích údajů podle svého, ale navenek musí všichni postupovat jednotně
  + Cílem rozdělení Internetu na AS je
    - Snížení směrovací režie
    - Zjednodušení správy sítě
  + AS = domény
    - Každému AS přiřazen 16bitový identifikátor (Autonomous System Number – ASN)
    - Odpovídají administrativním doménám – CESNET, PASNET, …
  + Stub AS
    - Připojen pouze k jednomu dalšímu AS
    - Směrovač stub AS je v rámci tohoto AS výchozí pro všechny sítě ležící mimo
  + Multihomed AS
    - Připojen k nejméně dvěma dalším AS, mezi kterými neumožňuje přenášení provozu
  + Transit AS
    - Připojen k nejméně dvěma dalším AS, mezi kterými umožňuje přenášení provozu
  + Interior routing - směrování uvnitř AS
  + Exterior routing – směrování mezi AS
  + Nutná spolupráce interior a exterior routing protokolů
  + Interní směrovače znají cestu do všech podsítí uvnitř AS
  + Hraniční směrovače sumarizují a zveřejňují interní cesty, aplikují směrovací pravidla
  + Směrování uvnitř AS hraje hlavní roli výkon
  + Směrování mezi AS hrají hlavní roli politiky a škálovatelnost
  + Kombinace nejlepších lokálních pravidel nemusí představovat globální optimum
  + Asymetrie cest (AS nemusí umožňovat přenos oběma směry)
  + Exterior Routing Protocol (EGP)
    - Protokol mezidoménového směrování
    - Využívá DV přístup
    - Cílem dosažitelnost, ne efektivita
    - Navržen pro stromovou strukturu Internetu – nepodporuje redundanci, neumí pracovat s cykly
  + Border Gateway Protocol (BGP)
    - Využívá Path Vector směrování – popis celých cest zahrnující všechny skoky
    - Podporuje redundantní topologie, umí pracovat s cykly
    - Pracuje nad spolehlivým protokolem TCP
    - Používá CIDR
* Unicast
  + Zasílání dat každému uzlu zvlášť
* Multicast
  + Zaslání dat jen jednou skupině uzlů, směrovače data replikují do potřebných směrů
  + Každým spojem jen jedna kopie dat
  + Hop-by-hop služba
  + Nezaručené doručení (best-effort, UDP, skupinová adresa)
  + Rozsah šíření omezen TTL
  + Skupina se identifikuje prefixem IP
  + Dva přístupy
    - Source Based Tree
    - Core Based Tree (Shared Tree)
  + Source Based Tree
    - Vysílající zašle data všem
    - Ti, co je nechtějí, odpoví zamítací zprávou
    - Posílá se jen ostatním
    - Omezení šířky – TTL
    - Vysoká režie, záplava broadcasty
  + Core Based Tree
    - O data si zažádají jen uzly, které je chcou
    - Až poté se vysílá
    - Redukce broadcastu – lepší škálování
    - Závislost na dostupnosti jádra
  + Nekonečná škálovatelnost
  + Nezatěžuje síť zbytečnými kopiemi
  + Problematické účtování
  + Problém se zajíštěným doručením
  + Snadný terč útoku DDoS
  + Absence kontroly členství
  + Správa skupin – Internet Group Management Protocol (IGMP)
    - Správa členství ve skupině
    - Lokální působnost
    - Typy zpráv
      * Přihlášení ke skupině
      * Odhlášení ze skupiny
      * Monitoring skupiny – dotazy směrovače na zájem uzlů setrvat ve skupině, řeší odstranění náhle vypadlých uzlů

**Transportní vrstva L4**

* Komunikace konkrétních aplikací na konkrétních uzlech sítě
* Možnost zajištění spolehlivého přenosu nad nespolehlivou (best-effort) IP sítí
* Data od odesílací aplikace transformuje do segmentů, které pak předává cílové aplikaci
* Ve spolupráci se síťovou vrstvou zajišťuje doručení dat mezi komunikujícími procesy – poskytuje jim logický komunikační kanál (iluze fyzického propojení)
* Nejnižší vrstva poskytující end-to-end služby
  + Hlavičky generované na straně odesílatele jsou interpretovány jen na straně příjemce
* Služby
  + Tvorba packetů
  + Řízení spojení
    - Spojované a nespojované služby
  + Adresace
    - Adresy služeb – porty
  + Zajištění spolehlivosti přenosu
    - Řízení toku a řízení chyb – end-to-end
  + Řízení zahlcení sítě a zajištění kvality služby
* Porty rozděleny do 3 tříd
  + Well-known
  + Registrované
  + Dynamické
* Adresace na L4 představuje formu (de)multiplexingu
* Spojované služby
  + Na začátku přenosu ustaveno spojení – udržováno po celou dobu přenosu dat
  + Packety jsou číslovány, doručení je potvrzováno
* Nespojované služby
  + Packety zasílány cílové aplikaci bez ustaveného spojení
  + Packety nejsou číslovány, ani potvrzovány – mohou se ztratit, zpozdit, dorazit mimo pořadí, atd.
* User Datagram Protocol (UDP)
  + Nejjednodušší transportní protokol poskytující nespojovanou a nespolehlivou službu
  + Ke službám IP přidává pouze process-to-process komunikaci a jednoduchou kontrolu chyb
  + Jednoduchost, minimální režie
  + Malá hlavička
    - Zdrojový port
    - Cílový port
    - Délka UDP packetu
    - Kontrolní součet
  + Procesy vyžadující jednoduchou komunikaci dotaz – odpověď
    - Např. služba DNS (Domain Name Service)
  + Procesy/protokoly s interním řízením toku a kontrolou chyb
    - Např. protokol TFTP (Trivial File Transport Protocol)
  + Real-time přenosy
  + Multicastové přenosy
  + Aktualizace směrovacích tabulek pomocí RIP
* Mechanismy zajištění spolehlivého přenosu
  + Spolehlivost přenosu zajištěna mechanismem pozitivního/negativního potvrzování
  + V případě výskytu chyby jsou data opětovně přeposílána – mechanismy ARQ (Automatic Repeat reQuest)
    - Stop-and-Wait ARQ
    - Go-Back-N ARQ
    - Selective-Repeat ARQ
  + Nutnost vypořádat se s duplicitami
  + Stop-and-Wait ARQ
    - Nejjednodušší mechanismus řízení toku a řízení chyb
    - Odesílací strana po odeslání packetu vyčkává na jeho potvrzení, aniž by odesílala další
    - Po uplynutí timeoutu je packet pokládán za ztracený a znovuposlán
    - V případě poškození packetu jej příjemce zahazuje a vyčkává na znovuposlání – žádné negativní potvrzení
    - Packety číslovány 0 a 1, potvrzení = ACK s číslem následujícího packetu
    - V případě obousměrného přenosu lze využít mechanismus Piggybacking
      * Kombinace datového packetu s potvrzením, místo dvou samostatných packetů se zasílá jen jeden
  + Go-Back-N ARQ
    - Stop-and-wait může do sítě vždy vyslat jen jeden packet – degradace výkonu
    - Zasílání více packetů bez vyčkávání na potvrzení
    - Packety číslovány zvyšujícími se čísly
    - Potvrzení packetu = ACK s číslem následujícího packetu
    - Možno využít piggybacking
    - Informace o odeslaných/přijatých packetech uchovávána pomocí sliding window
    - Protokol využit v TCP
  + Selective-Repeat ARQ
    - Go-back-n neefektivní pro vysoce ztrátové linky (packety mimo pořadí zahazuje)
    - Packety číslovány zvyšujícími se čísly
    - Out-of-order packety na straně příjemce bufferovány
    - Potvrzení packetu = ACK s číslem následujícího packetu
    - Možnost využít piggybacking
    - Negativní potvrzení zasílány v případě detekce ztráty/poškození packetu
* TCP protokol (Transmission Control Protocol)
  + Protokol poskytující spojovanou a spolehlivou službu
  + Pokud možno, odeslaná data budou příjemci doručena kompletní a ve správném pořadí
  + Oproti UDP orientován na přenos proudu bytů (UDP orientováno na přenos bloků dat)
  + Před začátkem přenosu nutnost ustavení spojení mezi odesílací a přijímací stranou
    - Tzv. handshake pro výměnu všech potřebných parametrů
    - Spojení rozeznatelné jen na koncových uzlech – end-to-end
    - Spojení pouze dvoubodové – point-to-point
  + Multiplexing a detekce chyb stejné jako v UDP
  + Přenos dat v UDP
    - Aplikace předává bloky dat, které UDP opatřuje hlavičkou a předává síťovému protokolu
  + Přenos dat v TCP
    - Aplikace předává TCP proud bytů, které TCP segmentuje, opatřuje hlavičkou a předává síťovému protokolu
    - Aplikacím poskytuje iluzi roury, která přenáší jejich data
  + Aplikací předaná data nutno uchovávat v bufferech
    - Nutnost vyrovnání rychlostí komunikujících stran
    - Buffery navíc využity pro řízení toku a chyb
  + Segmentace dat
    - TCP získává od aplikace proud bytů, síťova vrstva očekává bloky dat
    - Nutnost tvorby segmentů
    - Velikost segmentů omezena Maximum Segment Size (MSS)
    - Segmenty opatřeny TCP hlavičkou a předány síťovému protokolu
    - Číslovány nejsou bloky dat, ale jednotlivé přenášené bajty
  + Hlavička segmentů
    - Zdrojový/cílový port
    - Sekvenční číslo segmentu
    - Číslo potvrzovaného segmentu – piggybacking
    - Délka hlavičky
    - Řídící data
    - Velikost okna
    - Kontrolní součet
    - Urgentní data
    - Volby
  + Full-duplex přenos – obě strany musí iniciovat spojení
  + Three-way handshake
  + Ukončení spojení iniciováno jednou ze stran, spojení musí být uzavřeno oběma stranami
  + Řízení chyb (Error Control)
    - Během přenosu nutné detekovat poškozené, ztracené, duplikované a out-of-order segmenty
    - Mechanismy pro zajištění spolehlivého přenosu
      * Kontrolní součty
      * Potvrzování segmentů
      * Timeouty
    - Mechanismus přeposílání založen na Go-back-n ARQ, na rozdíl od něj je u příjemce buffer pro out-of-order segmenty
    - Timeouty založeny na RTT, typicky timeout = 2RTT
  + TCP řídí množství zasílaných dat, aby
    - Zabránilo zahlcení příjemce – řízení toku (Flow Control)
    - Zabránilo zahlcení sítě – řízení zahlcení (Congestion Control)
    - Množství dat zaslatelných do sítě je definováno velikostí okna příjemce a velikostí okna zahlcení – vybírá se menší hodnota
  + Řízení toku (Flow Control)
    - Zpětná vazba od příjemce
    - Příjemce informuje odesílatele o stavu svého přijímacího bufferu – v rámci ACK packetů, nebo pomoci piggybackingu
  + Řízení zahlcení (Congestion Control)
    - Snaha o přizpůsobení rychlosti vysílání dostupné kapacitě sítě
    - Proaktivní přístup (snaha zahlcení předcházet) vs reaktivní přístup (až dojde k zahlcení, je detekováno a řešeno)
    - V průběhu přenosu odhadována velikost okna zahlcení
    - Tři základní algoritmy pro odhad okna zahlcení (přístup AIMD – Additive Increase, Multiplicative Decrease)
      * Fáze Slow Start
        + Snaha o rychlé navýšení rychlosti odesílání až do určité hranice
      * Fáze Additive Increase
        + Zpomalení rychlosti růstu
        + Snaha o udržení vysoké rychlosti přenosu co nejdelší dobu
      * Fáze Multiplicative Decrease
        + Zahlcení sítě – snížení rychlosti přenosu
        + Umožňuje zajištění férovosti mezi TCP proudy
  + Víceproudové TCP
    - Zlepšuje chování TCP jen když nastávají izolované výpadky packetů
    - Komplikovanější než TCP (obvykle více vláken)
    - Nastartování zrychleno nejvýše lineárně
    - Synchronní přetěžování front a vyrovnávacích pamětí na směrovačích

**Kvalita služby (Quality of Service)**

* Všechny síťové toky jsou v TCP/IP obhospodařovány ekvivalentně
* Někdy je nezbytné některé toky upřednostnit, poskytnout jim definovanou kvalitu služby (QoS)
* Omezit maximální možnou ztrátovost, zpoždění, garantovat požadovanou přenosovou rychlost, …
* Nezbytné pro real-time přenosy
* Mechanismy zajištění kvality služby
  + Plánování
  + Formování/omezování toků
  + Prevence zahlcení
* Nutno zajistit na více vrstvách (L4, L3, L2)
* Plánování
  + Souvisí s obsluhou vstupních/výstupních front
  + Základní přístupy
    - FIFO (FCFS)
    - Priority Queuing
    - Weighted Fair Queuing
  + FIFO
    - Nejjednodušší uspořádání, využívá pouze jednu frontu
    - Žádná podpora priority, agresivní proudy zvýhodněny
  + Priority Queuing
    - Příchozí packety zařazeny do prioritních tříd
    - Každá prioritní třída má svou FIFO frontu
    - Vyšší třídy jsou obsluhovány dříve – nižší třídy se neobslouží, dokud nejsou vysokoprioritní fronty prázdné
    - Síťové toky s vyšší prioritou mají garantovánu přednostní obsluhu
    - Pokud existuje kontinuální tok vyšší priority, packety nižší priority nebudou nikdo odbaveny
  + Weighted Fair Queuing
    - Packety řazeny do prioritních front
    - Fronty obsluhovány střídavě dle přiřazené váhy (vyšší váha = více odbavených packetů)
    - Řeší problém vyhladovění předchozího mechanismu
* Formování/omezování toků (Traffic Shaping)
  + Mechanismus pro řízení množství a rychlosti odesílaných packetů
  + Průměrná rychlost odesílanách packetů
  + Špičková rychlost odesílanách packetů – max počet packetů, které lze do sítě zaslat za krátky časový interval
  + Maximální počet packetu, které lze do sítě odeslat najednou
  + Dva mechanismy
    - Leaky Bucket
    - Token Bucket
  + Leaky Bucket
    - Nezávisle na rychlosti příjmu jsou data zpracovávány konstantní rychlostí
    - Pokud je bucket plný, nové packety jsou zahozeny
  + Token Bucket
    - Umožňuje shromažďování tokenů v okamžiku nečinnosti uzlu
    - Za každou přenesenou datovou buňku je z bucketu odebrán token
    - Velikost bucketu ovlivňuje velikost krátkodobých špiček
* Prevence zahlcení (Congestion Avoidance)
  + Fronta se plní, žádné packety nejsou zahazovány
  + Jakmile je fronta plná, nové packety zahazovány
  + Nemožnost pružné reakce na blížící se zaplnění fronty
  + Může dojít k synchronizaci zahlcení mezi více směrovači a vytváření vln
  + Random Early Detection (RED)
    - Přesáhne-li zaplnění fronty určitou mez, začne směrovač zahazovat packety náhodně vybraných toků
    - Odesílatel toku sníží rychlost odesílání – reakce na ztrátu
    - Pravděpodobnost zahození se zvyšuje se zvyšujícím se zaplněním fronty
    - Odstraňuje problém globální synchronizace
  + Weighted Random Early Detection (WRED)
    - Totéž, co RED, ale zahazování se řídí i dle priority
* Kvalita služby v Internetu
  + Dva mechanismy zajištění kvality služby
    - Integrované služby
    - Rozlišované služby
  + Integrované služby
    - Aplikace oznámí síti své kvalitativní požadavky na přenos
    - Síť ověří dostupnost prostředků a rozhodne, zda požadavkům vyhoví
    - Pokud nelze požadavkům vyhovět, spojení je zamítnuto
    - Pokud lze požadavkům vyhovět, síť informuje všechny komponenty po cestě k příjemci, ať rezervují dostatečný objem prostředků – např. pomocí RSVP (Resource reSerVation Protocol)
    - Nutnost udržování stavu na vnitřních prvcích sítě
  + Rozlišované služby
    - Síti se neoznamují žádné požadavky na kvalitu přenosu
    - Každý packet vstupující do sítě je označen značkou, určující kvalitativní třídu
    - Umístěno do Type of Service (IPv4) nebo Traffic Class (IPv6)
    - Jednoduché
    - Žádné stavové informace na vnitřních prvcích sítě
    - Žádné úvodní zpoždění přenosu kvůli rezervaci zdrojů

**Relační vrstva L5**

* Správa relací – tzv. dialogů
* L1-L4 orientovány spíše na vlastní přenos dat mezi počítači
* Vyšší vrstvy se orientují na potřeby síťových aplikací
* Každé spojení obvykle zajišťováno prostřednictvím jednoho transportního spojení – ne nutně, lze i 1:M a M:1
* Řízení dialogu – která aplikace smí vysílat
  + 3 možnosti
    - Plně duplexní (obě strany naráz)
    - Poloduplexní (two-way-alternate)
    - Simplexní (one-way)
* Synchronizace (checkpointing)
  + Mechanismus kontrolních bodů
    - Příjemce si může na vysílajícím vyžádat návrat k zadanému kontrolnímu bodu (nové zaslání dat)
    - Dva druhy kontrolních bodů – hlavní a vedlejší
    - Lze se vracet nanejvýš k poslednímu hlavnímu checkpointu

**Prezentační vrstva L6**

* Konverze přenášených dat do jednotného formátu
* Na různých architekturách různé vnitřní reprezentace dat
  + EBCDIC vs ASCII
  + Jedničkový doplňkový kód vs dvojkový doplňkový kód
  + Little Endian vs Big Endian
* Nutnost jednotné interpretace dat na obou stranách
  + 2 možnosti dosažení
    - Přímé přizpůsobení stylu každý s každým
    - Převod do společného mezitvaru
* Prezentační vrstva předpokládá alternativu se společným mezitvarem
* Využit jazyk ASN.1 (Abstract Syntax Notation version 1)
* Aplikace prezentační vrstvě předává data + jejich popis v jazyce ASN.1
* Další služby
  + Šifrování a komprese dat

**Aplikační vrstva L7**

* Poskytuje služby pro uživatele
* Email, WWW, DNS, atd.
* Zahrnuje síťové aplikace a aplikační protokoly
  + Aplikační protokoly (HTTP, SMTP, atd) součástí síťových aplikací
* Aplikační protokoly definují
  + Typy zpráv, které si aplikace předávají
  + Syntaxi zpráv
  + Sémantiku zpráv
  + Pravidla, kdy a jak aplikace zprávy vysílají
* Členění aplikací
  + Dle užitého komunikačního modelu
    - Client-Server
    - Peer-to-peer
  + Dle přístupu k informacím
    - Pull model
    - Push model
  + Dle nároků na počítačovou síť
    - Aplikace s nízkými nároky
    - Aplikace s vysokými nároky
* Client-Server vs Peer-to-peer
  + Client-Server
    - Komunikace iniciována klientem
    - Po ustavení kanálu klient zasílá požadavky na server a ten mu odpovídá
    - Po ukončení komunikace je kanál uzavřen
    - Centralizace zdrojů
  + Peer-to-peer
    - Klienti spolu komunikují přímo
    - Každý uzel poskytuje své zdroje ostatním
    - Každý uzel využívá zdrojů ostatních
    - Decentralizace zdrojů
* Client-Server
  + Tenký klient
    - Aplikace na straně klienta vykonává minimum aplikační logiky
    - Jednodušší, menší nároky na HW
    - Menší škálovatelnost, vyšší objemy přenášených dat, single point of failure
  + Tlustý klient
    - Vetšina aplikační logiky na straně klienta
    - Menší nároky na server, nižší objem přenesených dat, možnost práce offline
    - Komplexní provedení i instalace, spotřeba lokálních zdrojů
* Přístup k informacím
  + Pull model
    - Přenos dat iniciován klientem
    - Např. webové prohlížeče
    - Asymetrický datový tok, rozmanité požadavky na propustnost
  + Push model
    - Přenos dat iniciován serverem automaticky na základě znalosti uživatelových požadavků
    - Např. stream multimedií
    - Jednosměrný datový tok, definované požadavky na propustnost
* DNS (Domain Name System)
  + Zpočátku řešeno pomocí host souborů, obsahující dvojice (doménové jméno, IP)
    - Neškálovatelné
  + Plochý jmenný prostor
    - Bez vnitřní struktury (např. mujRouterDomaVBrne) – nemožnost využití ve velkém systému
  + Hierarchický jmenný prostor
    - S hierarchickou vnitřní strukturou
    - Jména sestávají z několika částí s definovaným významem
    - Možnost decentralizace správy jmen
  + Jmenný prostor Internetu (Domain Name Space)
    - Hierarchické uspořádání
    - Struktura invertovaného stromu
    - Maximálně 128 úrovní
    - Každému uzlu přidělena jmenovka (label) a doménové jméno
    - Např. label = email, domain name = email.seznam.cz.
* HyperText Transfer Protocol (http)
  + Protokol pro přístup k datům na WWW
  + Součástí požadavku je Uniform Resource Locator (URL)
    - Definuje zdroj, který chce klient získat
    - Součástí URL je
      * Protokol pro přístup k odkazovanému zdroji
      * Host
      * Port
      * Cesta k informaci
      * Protokol://host:port/cesta
* Simple Mail Transfer Protocol (SMTP)
  + Struktura
    - Obálka – obsahuje adresu odeílatele/příjemce a další info
    - Zpráva – hlavičky a tělo zprávy
  + Emailové adresy se skládají z lokální\_část@doménové\_jméno
  + Doručení emailu se skládá ze tří fází
    - Doručení emailu lokálnímu mailserveru
    - Předání emailu cílovému mailserveru
    - Předání emailu cílovým poětovním klientem (POP3, IMAP4)
* File Transfer Protocol (FTP)
  + Určený pro přenos souborů mezi uzly
  + FTP klient s FTP serverem ustavuje dvě samostatná TCP spojení
    - Řídící spojení – otevřeno po celou dobu relace
    - Datové spojení – otevíráno pro každý přenášený soubor
* Kvantování
  + Diskrétní reprezentace hodnoty intenzity zvuku (zaokrouhlení na celé čísla)
* TCP
  + Zajištění bezchybnosti přenosu na úkor zvýšení end-to-end latence
  + Zajištění férovosti nedovoluje dostatečnou šířku pásma na vytížených linkách
* UDP
  + Nemá režii kolem oveřování bezchybnosti přenosu
  + Minimalistický, efektivnější, rychlejší
  + Minimálně navyšuje latenci přenosu
  + Vhodný pro přenos multimediálních dat

**Základy síťové bezpečnosti**

* Bezpečná komunikační síť by měla nabízet následující služby
  + AAA
    - Autentizace
    - Autorizace
    - Účtování (Accounting)
  + + zabezpečená komunikace
    - Důvěrnost
    - Integrita
    - Nepopiratelnost
* Jiný pohled na bezpečný systém
  + Dostupnost
  + Spolehlivost
  + Zabezpečení
  + Důvěrnost
  + Integrita
  + Udržovatelnost
* Autentizace
  + Ověření identity uživatele
  + Zahrnuje prokázání identity koncovému systému i komunikujícímu partnerovi
  + Metody zjištění identity
    - Co uživatel zná – heslo, PIN
    - Co uživatel má – token
    - Co uživatel je – biometriky
    - Co uživatel umí – správně odpovědět na náhodně vegenerovaný dotaz
* Autorizace
  + Oprávnění použít určitou službu nebo zdroj
  + Následuje po autentizaci
  + Na základě seznamů pro řízení přístupu
* Účtování (Accounting)
  + Sledování využívání síťových služeb uživateli
  + Informace mohou být využity pro správu, plánování, skutečné účtování nebo další účely
* Důvěrnost
  + Ochrana přenášených dat před neautorizovaným odhalením
  + Zajištěno šifrováním
* Integrita
  + Ochrana přenášených dat před neautorizovanou modifikací
* Nepopiratelnost
  + Nepopiratelnost odesilatele a nepopiratelnost doručení – slouží k prokázání odeslání zprávy
* Symetrická kryptografie
  + K šifrování i dešifrování využit stejný klíč
  + Nízká výpočetní náročnost, vhodné pro šifrování dlouhých zpráv
  + Nutnos sdílení tajného klíče
  + DES, 3DES, IDEA, …
* Asymetrická kryptografie
  + K šifrování je použit jiný klíč než k dešifrování
  + Keypair = public + private key
  + Není potřeba nikam posílat šifrovací klíč – bezpečnější
  + Veřejný klíč je možno dát všem
  + Pomalé, asymetrické šifry jsou vhodné pro krátké zprávy
  + RSA, DSA, …
  + Certifikát veřejného klíče
    - Informace, která váže identitu uživatele s jeho veřejným klíčem
    - Obsahuje jméno vlastníka, hodnotu veřejného klíče, dobu platnosti veřejného klíče, podpis vydavatele
    - Vydávají certifikační autority
* Autentizace stran
  + Heslem
    - Šifrováno sdíleným symetrickým klíčem
    - Negarantuje čerstvost hesla
  + S využitím náhodných čísel
    - A si od B vyžádá zaslání náhodného čísla
    - A toto číslo zašifruje symetrickým klíčem
    - Řeší čerstvost hesla
  + Vzájemná s využitím náhodných čísel
    - Stejné jako předchozí, autentizace je obousměrná
* Šifrování
  + Přenášená data šifrována nejčastěji s využitím symetrické kryptografie
  + Sdílené tajemství
    - např. Diffie-Hellman
    - asymetrická kryptografie – zvolený symetrický klíč je šifrován veřejným klíčem protistrany
* Digitální podpis
  + Mimo integrity a nepopiratelnosti zajišťuje i autentizaci stran
  + Obrácený postup asymetrické kryptografie – zpráva podepsána soukromým klíčem, ověřována veřejným klíčem
  + Podpis celého dokumentu
  + Podpis otisku dokumentu
    - Častěji využívané
    - Ze zprávy se vypočte otisk (hash), který je podepsán a odeslán s původním dokumentem
    - Řeší problém podpisu dlouhých zpráv
* Hashovací funkce
  + Jednosměrnost – po zahashování nelze získat původní dokument
  + One-to-one – nízká pravděpodobnost, že dvě různé zprávy budou mít stejný hash
  + Pro jakkoli dlouhý dokument má vždy pevnou délku
  + MD5 (prolomeno), SHA-256
* Všechny výše uvedené bezpečnostní protokoly lze realizovat na aplikační, transportní a síťové vrstvě
* IPSec
  + Kolekce protokolů pro zajištění zabezpečené komunikace na síťové vrstvě
  + Authetication Header (AH) – autentizace odesílatele a integrita zprávy
  + Encapsulating Security Payload (ESP) – autentizace odesílatele, integrita zprávy, důvěrnost přenosu
  + 2 módy
    - Transportní
    - Tunelovací
  + Zabezpečení všech datových toků mezi dvěma uzly, není potřeba upravovat aplikace
  + Žádné automatizované prostředky pro správu kryptografických klíčů
* Secure Sockets Layer / Transport Layer Security (SSL/TLS)
  + Protokoly pro zajištění zabezpečené komunikace na transportní vrstvě
  + Protokoly aplikační vrstvy je pro využití nutno upravit
    - HTTP -> HTTPS
    - FTP -> FTPS
* Principy bezpečné architektury sítě
  + Síťové protokoly a jejich pružnost
  + Princip redundance v designu sítě
  + Ochrana spojení a cest
  + Agregace spojení
  + Vícecestné směrování
  + Odolné překryvové sítě (RON)
  + Zabezpečené DNS

**Překryvové a P2P sítě**

* P2P síť je typicky virtuální síť utvořená nad existující síťovou infrastrukturou
* Překryvová síť je využita pro indexování a zjišťování sousedů (peerů) -> P2P systém je nezávislý na topologii základové (fyzické) sítě
* Data jsou přenášena po fyzické síti
* Nový peer musí za účelem svého připojení k P2P síti získat informaci o nejméně jedno jejím členovi (IP, port, …)
* Informace o dalších peerech mohou být získány od něj
* Překryvová síť je defacto virtuální síť vybraných uzlů vzniklá nad fyzickou sítí
* Členění P2P systémů
  + Centralizované – obsahují jeden nebo více centrálních serverů, které poskytují nejrůznější služby
  + Decentralizované – neobsahují žádné centrální servery
    - Struktura – plochá vs hierarchická
    - Topologie překryvné sítě – nestrukturovaná vs strukturovaná
  + Existují i hybridní systémy
* Centralizované P2P systémy
  + Kombinují výhody centralizovaných (klient-server) a decentralizovaných distribuovaných systémů
  + Obsahují jeden nebo více centrálních serverů, které uzlům poskytují vyhledávací služby a koordinují akce v síti
  + Pokud nějaký peer získá informaci o peerovi, který spravuje požadovaná data, komunikuje s ním napřímo
  + Citlivé na útoky, bottleneck pro sítě s velkým počtem uzlů, slabá škálovatelnost a robustnost
* Decentralizované (pure) P2P systémy
  + Všichni peerové mají stejná práva a povinnosti
  + Každý peer má pouze částečnou představu o celé síti
  + Lokalizace peerů poskytujících definovaná data/služby je v decentralizovaných P2P systémech velkou výzvou
  + Imunní vůči single point of failure
  + Poskytují vysoký výkon, škálovatelnost, robustnost, …
  + Plochá struktura – funkcionalita rovnoměrně rozložena mezi všechny uzly sítě
  + Hierarchická struktura – směrovací struktury sestávají z více vrstev
  + Nestrukturovaná topologie
    - Každý peer je zodpovědný za svá vlastní data a drží si informace o sousedech, na které může směrovat vyhledávací dotazy
    - Lokalizace dat je výzvou
    - Žádná garance na kompletnost odpovědi
    - Žádná garance na dobu potřebnou na odpověď
  + Strukturovaná topologie
    - Lokace dat je pod kontrolou určité definované strategie (distribuované hashovací tabulky – DHT)
    - Poskytují garanci doby nezbytné pro nalezení odpovědi
    - Vyšší režie
* Hybridní P2P systémy
  + Výhodou centralizovaných P2P sítí jsou rychlé a přesné odpovědi na vyhledávací dotazy, na úkor škálovatelnosti
  + Výhodou decentralizovaných P2P sítí je škálovatelnost, na úkor delšího času vyhledávání informací
  + Pro udržení škálovatelnosti neexistují žádné centrální servery
  + Některé peery jsou vybrány a prohlášeny za servery sloužící dalším peerům
  + Lokalizace dat je prováděna kombinací decentralizovaného a centralizovaného přístupu – dotazem na super peer uzly

**Bezdrátové sítě**

* Založeny na buněčné infrastruktuře
* Každá buňka pokryta jednou základovou stanicí
* Základové stanice jsou mezi sebou propojeny drátem
* Podpora mobility je dosažena přepínáním mobilního klienta mezi základovými stanicemi
* Když není žádná infrastruktura dostupná/je příliš drahé infrastrukturu vybudovat/není dostatek času na vybudování -> Wireless Ad-hoc Networks
* Wireless Ad-hoc
  + Snaží se vybudovat komunikační síť bez nutnosti vybudované infrastruktury, pouze s využitím schopností účastníků
  + Ad-hoc síť = síť konstruovaná na požádání pro specializované účely
  + Kolekce autonomních uzlů, které spolu komunikují skrze jimi zformovanou bezdrátovou multihop síť, udrýovanou decentralizovaným způsobem
  + Každý uzel je jak koncový uzel, tak síťový směrovač
  + Topologie je obecně dyamická – může se měnit v závislosti na odchodech a příchodech uzlů, je potřeba efektivních směrovacích protokolů
  + Velmi rychlé vybudování
  + Odolnost – žádný single point of failure
  + Efektivnější využívání rádiového spektra než u buněčných sítí
  + Neexistence centrální entity organizující participující uzly
  + Omezený dosah bezdrátové komunikace
  + Mobilita uzlů – kvalitu sítě lze hodnotit podle rychlosti adaptace na změny v topologii
  + Mobilní ad-hoc sítě (Mobile Ad-hoc Networks, MANETs)
  + Řízení přístupu k médiu (medium access control) – neexistuje základová stanice, která by určovala oprávnění k přístupu ke sdílenému médiu, musí být rozhodováno distribuovaným způsobem
  + Směrování – hledání cest mezi komunikujícími uzly
  + Dopravní ad-hoc sítě (Vehicular ad-hoc networks, VANETs)
    - Využívá aut jako uzlú/směrovačů pro vytvoření mobilní sítě
    - V porovnáni s MANETy se uzly VANET pohybují organizovaným způsobem
  + Bezdrátové senzorové sítě (Wireless Sensor Networks, WSNs)
    - Místo interakce s lidmi interakce s prostředím
    - Uzly sítě jsou vybaveny senzorem/ovládacím prvkem pro měření/ovládání daného prostředí
    - Uzly zpracovávají získané informace a bezdrátově spolu komunikují

**Historie komunikačních sítí**

* Modem – 1949 MIT
  + Do 60. let hlavní forma komunikace
  + Přepojování okruhů
  + 1958 AT&T vyrábí nové modemy – 110 b/s
* 1953-62 projekt SAGE
  + 23 lokalit propojených telefonními linkami, komunikující přes modemy
  + Systém vzdušné obrany USA
* Kroucená dvoulinka – 1881
  + Alexander Graham Bell
  + Varianty FTP, STP, UTP (Foiled/Shielded/Unshielded Twisted Pair)
* Koaxiální kabel – 1880
* Optická vlákna
  + Známá od 2. pol. 19. st.
  + 1966 první funkční demonstrace přenosu dat
  + V 70. letech snížení útlumu -> použitelné pro přenos dat
* ALOHA – 1971
  + První bezdrátová síť, vznikla na Hawaii
  + Cílem propojit jednotlivé ostrovy
  + Hvězdicová topologie
* Ethernet – 1973/4
  + Vyvinut v Xerox
  + Inspirace v ALOHANETu
* ARPANET – 1969
  + Decentralizovaná síť funkční i při jaderném útoku
  + Interface Message Processor (IMP)
  + Od roku 1983 TCP/IP
  + Vypnut 28.2.1990
* Internet – 1987
* WWW – 1989
  + V CERNu
  + HTML + URL + HTTP
* 1991 – ČSFR připojena k Internetu
* 13.2.1992 – oficiální zahájení provozu sítě Internet v ČSFR