

IPv6

- 128b
- jednodušší formát hlavičky 40B – pouze nezbytné informace
- extension headers
- support for real-time transmissions – streams' tagging and priorities
- authentication, encryption and integrity verification support
- mobility support – using so-called home agents
- autoconfiguration – stateful and stateless
- CIDR
- header: verze, priorita – QoS tag, flow label, payload length, next header, hop limit, s. addr, d. addr, payload
- address form: 0s might be omitted, groups of zeros might be replaced by "::"
- **hierarchie adres:**
 - global routing prefix: n bits (48)
 - subnet addr: 64 – n (16)
 - interface addr: 64 bits
- **typy adres:**
 - unicast
 - multicast – ff00::/8, ! místo broadcastu je multi. adresa ff02::1 – všichni v LAN
 - anycast – data doručena pouze jednomu členu skupiny
 - broadcast
- **Path MTU Discovery** – pouze zdrojové zařízení rozhoduje o velikosti fragmentu
 - zdroj pošle datagram s konkrétní MTU, pokud nepřijde ICMPv6 Packet Too Big, je zachována velikost podle tohoto MTU

Neighbor Discovery Protocol

- autokonfigurace
- **duplicate IP address detection (DAD)**
 - během autokonfigurace
 - uzel pošle NS s jeho adresou a nastaví si svojí na :: (samé nuly), pokud přijde odpověď je již adresa zabraná
- **Neighbor Unreachability Detection** – periodická kontrola dostupnosti
- 5 ICMP zpráv:
 - Router Solicitation (RS)
 - Router Advertisement (RA)
 - Neighbor Solicitation (NS)
 - Neighbor Advertisement (NA)
 - ICMP Redirect – gateway pošle uzlu zprávu na základě níž může uzel vybrat lepší next hop router
- L2 address resolution: Neighbor Solicitation (NS) -> odpověď od uzlu NA
 - unsolicited Neighbor Advertisement – když se L2 adresa změní

- Autokonfigurace:
 - **stavová:**
 - podobná DHCP - DHCPv6
 - **bezstavová:** vše ví router
 - routery periodicky informují všechny uzly o konkrétní konfiguraci - Router Advertisement (RA) – MTU, prefix, L2 adresa router rozhraní
 - pokud se připojí nový uzel požádá o konfiguraci - Router Solicitation (RS)
 - **Link-Local adresa:**
 - prefix FE80
 - kombinace informací z routeru a MAC adresy nebo identifikátoru
 - Link-Local Address Uniqueness Test – spíše při použití ID
 - **Global Address:**
 - složená z prefixu sítě poskytnutého routerem a identifikátoru vygenerovaného pro link-local addr
 - autokonfiguration flag:
 - M – použití stavové konfigurace
 - O – další info při stavové metodě

Mobility Support

- Adresy:
 - **Home Address** – a global unicast persistent address - always accessible (even though not being in its home network)
 - **Care-of Address** – (the address is based on the network where the host is currently located)
- **Home Agent (HA)** – router v domácí síti, přijme datagram a forwarduje ho skrz tunel na uzel
 - seznam všech HA na stejné lince a provádějící HA službu
- binding – asociace care-of adresy a home adresy
 - care-of adresa je odeslána HA
- možnosti komunikace:
 - obousměrný tunnelling – posílané skrz HA
 - optimalizace:
 - přímá komunikace se zařízením a korespondenčním uzlem
 - uzel registruje care-of adresu přímo u korespon. uzlu
 - Return Routability (RR) Procedure – autentizace?
 - Home Test Init: MN -> CN skrz HA
 - Care-of Test Init: MN -> CN
 - Home Test: CN -> MN skrz HA
 - Care-of Test: CN -> MN
 - MN a CN Management Key – secure binding update messages

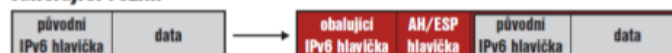
Security

- **CIA:**
 - **Confidentiality:** stored or transmitted information **cannot be read or altered** by an unauthorized party
 - encryption
 - secret key cryptography
 - public key cryptography
 - **Integrity:** any **alteration** of transmitted or stored information **can be detected**
 - secure checksum
 - hash
 - **Availability:** the information is **readily accessible** to authorized users at all times
- **AAA:**
 - **Authentication:** ensuring that an individual or group **are really the one who they say they are** (the act of clarifying a claimed identity)
 - **Authorization:** ensuring that the authenticated user or group **has the proper rights to access information** they are attempting to access
 - **Accounting:** the act of **collecting information** on resource usage (e.g., a log)
- **Nonrepudiation:**
 - means that a specific action cannot be denied by any of the parties involved
- obecné bezpečnostní mechanismy – IPSec
 - **Authentication Header – AH**
 - zdroj – vypočítá a nastaví Integrity Check Value
 - nešifruje
 - **Encapsulating Security Payload – ESP** – šifrování dat
 - ochrana dat před zkoumáním neautorizovanými stranami
 - podporuje autentikační schéma
- **Security Associations (SA):** sada bezpečnostních informací, které popisují konkrétní druh bezpečného spojení mezi jedním a druhým zařízením
 - **Security Parameter Index** – 32 bit unikátní index
 - **IP Destination Address**
 - **Security Protocol Identifier** – specifikuje zda je asociace pro AH nebo ESP
 - 3 elementy:
 - klíč
 - šifrovací nebo autentikační mechanismus
 - přídatné parametry pro algoritmy
 - Internet Key Exchange version 2 – dohodnutí klíčů
- IPSec módy:

transportní režim



tunelující režim



- Transport mode
 - message is processed by AH/ESP and the appropriate header(s) are added in front of the transport (UDP or TCP) header
- Tunnel mode
 - complete encapsulated IP datagram
 - original IP datagram is secured and encapsulated within another IP datagram

QoS

Integrated Services:

- aplikace předloží požadavky
- síť zkontroluje zda jsou požadavky dosažitelné a rozhodne zda požadavkům vyhoví
- pokud jsou požadavky nesplnitelné, je spojení odmítnuto

Differentiated Services:

- garance, že přenosová kvalita se nezhorší
- paket je označen tagem Priority class -> Traffic Class field
 - simple, good scalability, není iniciální zpoždění

Headers field:

- **Traffic Class field (PRI):**
 - 1B
 - DiffServ – 6b DiffServ CodePoints – 64 codepoints – každý specifikuje jak forwardovat pakety
- **Flow Label:**
 - 20b
 - routery zacházejí s pakety patřícími do stejného toku podobným způsobem:
 - při prvním paketu se uloží všechny informace do cache
 - informace jsou pak použity při směrování dalších paketů
 - triple is used: flow label, source addr, dest. addr

Přechod na IPv6

- problémy přechodu z IPv4 na v6:
 - zápis adresy: desítková (4) -> hex (6)
 - prostor v paměti: 32bit (4) -> 128b (problém najít vhodný dat typ)
 - problémový zápis URL
 - IPv6 má více adres pro jedno rozhraní – Link-local, global
- spolupráce IPv4 a IPv6:
 - **Dual stack** – IPv4 i IPv6 zároveň
 - možnost využít obojí, při plném přechodu na v6 stačí odebrat kód pro v4
 - velká spotřeba zdrojů, aplikace musí umět zjistit, která verze se má použít, DNS musí být schopen řešit oba, směrovací protokoly musí umět řešit oba přístupy

- **Tunneling** - IPv6 datagrams are encapsulated into IPv4 datagram's data
 - není specifický postup jak má být docíleno výsledku
 - přidání náklad, těžké řízení zapouzdřeného provozu, konce tunelu jsou body selhání
- **Translators** - a device translates IPv6 datagrams into IPv4 datagrams and vice versa (pouze v nutných případech)
 - jednoduchá komunikace IPv6 hosta s IPv4
 - nepodporuje výhody IPv6, omezení designu topologie (NAT routery musí být stejné)

Advanced Routing Mechanisms

Routing = the process of finding a path in the network between two communicating nodes

- Distance Vector (Bellman-Ford)
 - sousední routery si periodicky vyměňují své směrovací tabulky
 - metrika je počet hopů
 - pomalá konvergence
 - pro malé sítě
- Link State (Dijkstra's)
 - sousední routery si periodicky vyměňují informace o stavu linky
 - routery mají kompletní informace o celé topologii
 - pro velké sítě
- Path Vector (varianta DV)
 - jsou posílány celé cesty
- autonomní systémy: rozdělení internetu do autonomních systémů
 - jednodušší směrovací tabulky, méně informací pro výměnu
 - jednodušší management sítě
 - 16b identifikátor Autonomous System Number
 - **routing:**
 - interior routing – uvnitř AS (IGP): RIP OSPF IGRP
 - exterior routing – mezi AS (EGP): BGP-4

Distance Vector

- **RIP - Routing Information Protocol**
 - metrika: počet hopů
 - periodická odesílání informací: každých 30s přes UDP
 - detekce eroru 180s
 - malé sítě
 - **RIPv1** – broadcast, 16 hopů je nekonečno, nepodporuje CIDR
 - **RIPv2**
 - subnet mask -> podpora CIRD
 - multicast 224.0.0.9
 - **RIPng** – rozšíření pro podporu IPv6
- **IGRP - Interior Gateway Routing Protocol**
 - Cisco
 - max. hop count 255
 - základní verze nepodporuje CIDR
 - kompositní metrika – bere v úvahu: Delay, Bandwidth, Reliability, Load + koeficienty
 - **EIGRP** - provides loop-free routing, provides reliable delivery, **allows variable length subnet masking** – CIDR

Link State

- **OSPF - Open Shortest Path First**
 - metrika je cena (cost)
 - rozhraní má přiděleno číslo 1-65535, menší číslo je lepší,
 - defaultně $\text{cost} = 100000000/\text{bandwidth}$
 - autentizace zpráv
 - směrovací oblasti – poddoména AS
 - load-balancing
 - CIDR
 - zapouzdřeny v IP datagramech – multicast 224.0.0.6, ff02::5, ff02::6
- **IS-IS**
 - standardizované organizací ISO
 - podobnosti s OSPF:
 - hierarchie sítě do dvou úrovní oblastí
 - Hello packets
 - CIDR
 - link state databáze
 - rozdíly oproti OSPF:
 - zapouzdřeny v rámci (L2) – relativní ochrana proti spoof a útokům
 - neutrální k verzi IP
 - povoluje deklaraci přetížení
 - cena 0-63
 - lepší škálovatelnost

Path Vector

- **BGP - Border Gateway Protocol v. 4**
 - informace pro komunikaci mezi sítěmi v AS
 - 179/TCP
 - definice směrovacích pravidel – politiky
 - metrika je hop count
 - CIDR
 - zprávy:
 - OPEN
 - UPDATE
 - KEEPALIVE
 - NOTIFICATION – used to close a session or to report an error
 - ROUTE-REFRESH
- **IBGP – Internal BGP**
 - informace o sousedních AS vnitřním routerům

Router Functions

- úkoly:
 - směrování – nalezení nejlepší cesty
 - forwarding – paket je předán z jednoho rozhraní na jiné
- **Funkce:**
 - **IP Header Validation** – správná verze protokolu, délka hlavičky, checksum
 - **Packet Lifetime Control** – decrement TTL, TTL <= 0 zahozen, ICMP odesílateli
 - **Checksum Recalculation** – změna TTL -> přepočítání header checksum
 - **Route Lookup** – stanovení odchozího rozhraní podle dst addr
 - **Fragmentation** – zmenšení paketu, aby vyhovoval MTU
 - **Handling IP Options** – indikace IP voleb
 - **Packet Classification** – rozlišení src addr, dst port, src port
 - **Packet Translation** - NAT
 - **Traffic Prioritization** - QoS
 - **Routing Protocols** – implementace směrovacích protokolů
 - **System Configuration**
 - **Router Management** – monitoring, SNMP
- **Router Elements:**
 - **Network Interfaces** – porty na fyzické linky
 - **Forwarding Engines** – na které rozhraní má být paket předán
 - **Queue Manager** – buffer pro dočasné uložení při přepnutí
 - **Traffic Manager** – prioritizace a regulace odchozího provozu
 - **Backplane** – provádí konektivitu pro síťová rozhraní
 - **Route Control Processor** – implementace směrovacích protokolů
 - konfigurace
 - operace s errorry
 - odesílání ICMP zpráv
- CIDR – nejdelší prefix se vybere
- Packet Classification/Filtering
 - typ provozu
 - flexibilní účtování
 - stanovení pravidel a politik
 - akce: permit, deny, encrypt
 - algoritmy: naivní alg., dvou dimensionální řešení, rozhodovací strom

Traffic Engineering

is all about **discovering what other paths** and links are available in the network, what the **current traffic usage** is within the network, and **directing traffic to routes** other than the shortest so that **optimal use of the resources** in the network is made.

- dosaženo:
 - extensions to existing IGP protocols
 - traffic monitoring tools
 - traffic routing techniques
- kroky:
 - měření a sbírání informací o provozu
 - SNMP (Simple Network Management Protocol)
 - NetFlow – Cisco tool
 - sFlow, ntop

- získání topologie a konfigurace ze sítě
 - manuálně
 - automaticky - extensions to existing IGP protocols – OSPF, IS-IS
- nastavení vah linkám – jedenkrát týdně nebo denně
 - problém optimalizace – minimalizace ceny směrování

Multiprotocol Label Switching (MPLS)

- forwardování paketů na základě label
- connection-oriented – směrování po předkonfigurované cestě Label Switched Paths (LSP)
- klasifikace paketů do FEC tříd (Forward Equivalence Class)
 - typ služby, VPN, ...
 - vytvoření labelů pro všechny FEC třídy
- tabulky Label Information Base (LIB)
 - mapování {incoming interface, incom. label} to {outgoing interface, out. label}
 - **label distribution protocol** – LDP – TCP/UDP
 - **Downstream-on-demand**, ordered control approach
 - nevysílá FEC-to-label binding až do žádosti
 - **Downstream-unsolicited**, independent control approach
 - nečekají na žádost a vysílají FEC-to-label binding svým sousedům
- Síťové komponenty:
 - Edge Label-Switched Routers (Edge-LSRs) = border routers
 - Ingress-LSR
 - Egress-LSR
 - Core Label-Switched Routers (Core-LSRs) = forwarding založen na labelech
- MPLS header:
 - Label
 - Traffic Class field
 - Stack – set to one for the last entry in the label stack, and zero for all any
 - TTL
- *Generalized MPLS*:
 - *Packet-Switch Capable* – (i.e., GMPLS encompasses MPLS)
 - *Time-Division Multiplexing Capable* – for timeslot-based circuit switching
 - *Lambda-Switch Capable* – for wavelength switching at optical cross-connects
 - *Fiber-Switch Capable* - for fiber-level switching at optical cross-connects

QoS-Based Routing

- a routing mechanism under which **paths for flows are determined based** on some **knowledge of resource availability** in the network as well as the **QoS requirement** of the flows, or
- a **dynamic routing protocol** that has **expanded its path-selection criteria to include QoS** parameters such as available **bandwidth**, link and end-to-end path utilization, node resources consumption, **delay** and **latency**, and induced jitter
- metrika je vybraná z: available bandwidth, delay, jitter, ...

směrovací algoritmy:

- **Source-based routing algorithms**
 - router má globální info o síti
 - stanoví cestu a src router upozorní ostatní routery, jak traffic flow forwardovat
 - problém škálovatelnosti – není vhodné pro velké sítě

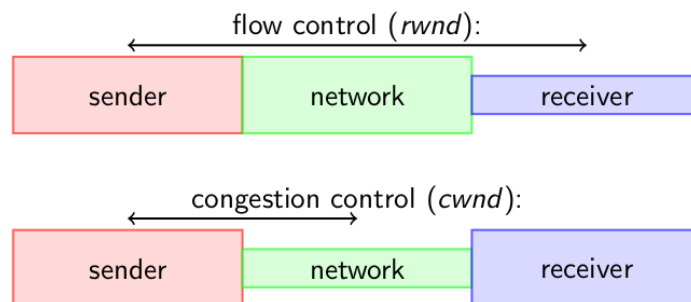
- **Hop-by-hop routing algorithms** (also called distributed routing algorithms)
 - každý router zná jen další hop k cíli
 - dopře kompatibilita s existujícími směrovacími protokoly
 - problém škálovatelnosti – není vhodné pro velké sítě, problém smyček
- **Hierarchical routing algorithms**
 - více úrovní:
 - nižší level – routery
 - vyšší level – routery seskupeny do skupin
 - směrovací informace jsou integrovány do hraničních uzlů skupin
 - dobře škálovatelný
 - Private Network-Network Interface (PNNI)
 - protokol pro ATM sítě
 - nepodporuje multicast a politiky
 - QoS routing extensions to OSPF
 - link **bandwidth** and **propagation delay** are the only metrics extension added to Link State Advertisements (LSAs)

Behind Traditional TCP

- chování protokolů:
 - aggressiveness – ability to **utilise available bandwidth**
 - responsiveness – ability to **recover from a packet loss**
 - fairness – **getting a fair portion of network throughput** when more streams/participants use the network

Tradiční TCP:

- **Flow control**
 - an explicit **feedback from receiver(s)** using **rwnd** deterministic
- **Congestion (zahlcení) control**
 - an approximate **sender's estimation (odhad) of available throughput** (using **cwnd**)



- $ownd = \min\{rwnd, cwnd\}$
- bandwidth $bw = (8 * MSS * ownd) / RTT$
- **Tahoe**
 - $cwnd = cwnd + 1$. . . per RTT without loss (above $ssthresh$)
 - $ssthresh = 0,5 cwnd$
 - $cwnd = 1$. . . per every loss
- **Reno**
 - rychlý opakovaný přenos
 - TCP příjemce odesílá ihned duplikátní ACK když dojde k out-of-order segment (segment mimo pořadí)
 - ztráta je indikována 3 duplikátními ACK
 - pokud jsou jednou přijaty, TCP prování rychle opakovaný přenos bez čekání na znovupřenosový timer
 - fast recovery – slow-start phase not used any more
 - $ssthresh = cwnd = 0,5 cwnd$
- reakce na ztrátu paketu souhrn:
 - Tahoe: the whole actual window $ownd$
 - Reno: a single segment in the Fast Retransmission mode
 - NewReno: more segments in the Fast Retransmission mode
 - Selective Acknowledgement (SACK): just the **lost packets**
- **férovost v tradičním TCP:**
 - streams with different RTT
 - streams with different MTU

- **TCP Vegas**
 - congestion control
 - když je síť zahlcená (přetížená), RTT se zvyšuje
 - RTT je monitorována během přenosu
 - pokud je detekováno zvýšení RTT, cwnd je lineárně sníženo

Vylepšení tradičního TCP

- **Multi-stream TCP**
 - vylepšuje chování TCP při izolované ztrátě paketů
 - komplikovanější než tradiční TCP
- **Web100**
 - software, který implementuje nástroje do Linuxového TCP/IP stacku
 - kernel patch + knihovny a nástroje
 - umožňuje:
 - monitoring
 - tuning
 - kooperace s HW
 - podporuje auto-tuning

Konzervativní rozšíření do TCP

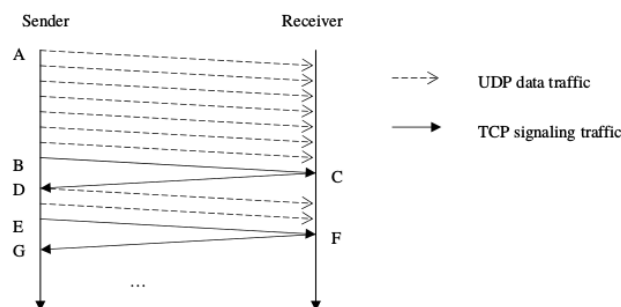
- **GridDT**
 - modifikací pro určitý účel
 - rychlejší slowstart
 - AIMD modifikace pro kontrolu zahlcení:
 - $cwnd = cwnd + a$. . . per RTT without packet loss
 - $cwnd = b \cdot cwnd$. . . per packet loss
- **Scalable TCP** - Tom Kelly
 - kontrola zahlcení již není AIMD -> MIMD (Multiplicative Increase Multiplicative Decrease):
 - $cwnd = cwnd + 0,01 \cdot cwnd$. . . per RTT without packet loss
 - $cwnd = cwnd + 0,01$. . . per ACK
 - $cwnd = 0,875 \cdot cwnd$. . . per packet loss
 - při malé velikosti okna nebo při vyšších ztrátách se mění na AIMD mode
- **High-Speed TCP** - Sally Floyd
 - kontrola zahlcení AIMD/MIMD
 - $cwnd = cwnd + a(cwnd)$. . . per RTT without loss
 - $cwnd = cwnd + a(cwnd) / cwnd$. . . per ACK
 - $cwnd = b(cwnd) \cdot cwnd$. . . per packet loss
 - $a(cwnd)$ a $b(cwnd)$ - MIMD – funkce s konstantami zjištěnými při měření
- **H-TCP**
 - AIMD
 - $cwnd = cwnd + 2(1-\beta) \cdot a(\Delta) / cwnd$. . . per ACK
 - Δ - čas uplynulý od posledního zahlcení
 - $cwnd = b(B) \cdot cwnd$. . . per loss
- **BIC-TCP**
 - defaultní algoritmus v Linuxových jádrech
 - používá binary-search algoritmus
 - packet-loss, additive increase, binary search, maximum probing
- **CUBIC-TCP** – nová verze BIC-TCP, používá kubickou funkci

TCP Extensions with IP Support

- **Quickstart**
 - interakce s IP protokolem
 - 4 byte volba v IP hlavičce, QS TTL and Initial Rate
 - nastaví QS TTL a požadovaný Initial Rate
- **E-TCP**
 - Early Congestion Notification (ECN) – komponenta Advanced Queue Management
 - ECN flag nastavený routery, když se blíží zahlcení linky/bufferu/fronty
 - reakce TCP na ECN stejně jako na packet loss
 - nutná konfigurace routerů
- **FAST**
 - Fast AQM Scalable
 - pokud je několik paketů ve frontě (detekováno pomocí RTT monitoringu), je odesílací rate snížena
 - rozdíl oproti TCP Vegas:
 - Vegas dělá fixní velikost úpravy rate (tempo, míra), nezávisle na vzdálenosti nynějšího a cílového rate
 - FAST dělá velké kroky pokud je daleko od rovnováhy a menší když je blízko
 - FAST může rozšířit použití ECN značení a nahradit delay fronty a packet loss jako měření zahlcení

Jiné přístupy než TCP

- **tsunami**
 - TCP pro řízení kanálu
 - UDP pro data
 - MIMD kontrola zahlcení
 - vysoce konfigurovatelné
- **Reliable Blast UDP – RBUDP**
 - TCP pro řízení kanálu, UDP pro data
 - pro přenosy z disku na disk, resp. přenosy kde kompletní přenášená data mohou být uloženy v paměti odesílatele



- eXplicit Control Protocol – XPC
 - uses a feedback from routers per paket
- SCTP - multistream, zprávy jako UDP (spolehlivé), congestion control jako TCP
- DCCP - zprávy jako UDP (nespolehlivé), congestion control jako TCP
- STP – bez sofistikované congestion control
- Reliable UDP - zajišťuje spolehlivé a uspořádané doručení
- XTP

Peer-to-peer

Peer-to-peer (P2P) systems are **distributed systems consisting of interconnected nodes** able to **self-organize into network** topologies with the **purpose of sharing resources** such as content, CPU cycles, storage and bandwidth, capable of adapting to failures and accommodating transient populations of nodes while maintaining acceptable connectivity and performance, **without requiring the intermediation or support of a global centralized server or authority**.

- **P2P Properties**

- **Symmetric role** – každý uzel je client i server
- **Scalability** – mohou se škálovat až do tisíců uzlů
- **Heterogeneity** - heterogeneous in terms of the hardware capacity of the nodes
- **Distributed control (Decentralization)** - ideally, no centralized structures should exist in P2P systems
- **Dynamism** - the topology of P2P systems **may change very fast** due to joining of new nodes or leaving existing ones
- **Resource sharing** - each peer contributes system resources (computing power, data, bandwidth, presence, etc).

- **Client-server vs. P2P:**

- cs: jedna centralizovaná instance pasivně naslouchající
- p2p: sestává se z mnoha identických softwarových modulů, které běží na různých počítačích
 - peery komunikují přímo mezi sebou
- porovnání:
 - Ease of development
 - C-S is more established and familiar than P2P
 - C-S exhibits simple interaction patterns for clients and server, while
 - P2P involves more complex interaction patterns between peers
 - Manageability
 - it is easier to maintain a centralized server
 - Scalability
 - C-S scalability is limited by fixed server hardware
 - P2P is scalable by nature
 - Security
 - responsibility for the C-S security lies within the server
 - responsibility for P2P security is distributed across peers
 - Reliability
 - C-S multiple redundant servers
 - P2P multiple peers are usually able to provide the same service

P2P architecture

- P2P je typicky virtuální síť na existující síti
- Application layer
- Middleware layer
 - snadnější vývoj aplikací díky skrytí overlay
- Base overlay layer
- *Physical network*

- Peer Discovery
 - **Static** – každý peer má předkonfigurované adresy ostatních peerů
 - **Centralized directory** – každý peer je předkonfigurován se sítí centralizovaného serveru
 - peer kontaktuje server a získá aktualizovaný seznam peerů
 - peer se indikuje serveru jako aktivní
 - **Member Propagation Techniques with Initial Member Discovery**
 - po najití jednoho existujícího peeru, získá informace o zbylé síti od něj
- Overlay Network Topology
 - Random Mesh
 - Tiered
 - vrstevníci jsou řazeny do tříd stromu v závislosti na jejich inzerovaných zdrojích a konektivitě
 - Ordered Lattice
 - každý uzel je přímo připojen ke 4 sousedním peerům
- Service Discovery
 - centralized – servis je tážán na umístění služby
 - pure P2P – požadavek je flooded nebo hashován skrz peery

Taxonomy of P2P Systems

- **centralized** – one or more central servers (SETI@home, Napster, Jabber, Net-Z)
- **decentralized** – no central servers (Gnutella, Crescendo, PAST, FreeNet, Canon)
 - **structure**
 - **flat** – funkcionality a zatížení je rovnoměrně rozložena
 - **hierarchical** – několik vrstev směrovacích struktur
 - **topology**
 - **unstructured** - každý peer je odpovědný za svá data a udržuje sady sousedů, kterým může předávat dotazy
 - **static configuration**
 - **re-configurable**
 - **structured** - umístování dat je pod kontrolou určitých předem definovaných strategií (distributed hash table – DHT)
 - **precise**
 - **probabilistic**
- **hybrid** – výhody obou: rychlé a spolehlivé lokalizování zdrojů + škálovatelnost
 - místo serverů jsou vybrány silné peery, které se jako server chovají (super peer)
- **Centralized** – server help peers to locate their desired resources or act as task scheduler
 - server jako jediný může selhat a tím selže celá síť
 - bottleneck pro hodně peerů
 - špatná škálovatelnost
- **Decentralized** – peery mají rovnocenná práva a povinnosti
 - imunitní proti jednomu bodu selhání

Routing in P2P networks

- hlavní rozdíl ve variantách je množství informací (metadat), která se udržují v každém peeru
- metriky:
 - Storage – ukládací prostor pro metadata
 - Efficiency – dokáže lokalizovat rychle zdroje
 - Usability – jednoduchost použití a typ dotazů
 - Coverage – obsahuje prohledávaný prostor odpověď na dotazy
 - Scalability – počet zpráv, které je nutné směřovat než je lokalizována informace
- Routing in **Unstructured** P2P networks:
 - každý peer obvykle ukládá svá vlastní datové objekty a udržuje sadu odkazů na sousední uzly
 - žádný peer nemá globální znalosti o umístění dat
 - flooding techniky
 - TTL
 - **strategie:**
 - BFS
 - DFS
 - **Heuristic-Based Routing Strategies:**
 - **Iterative Deepening**
 - dotac iniciuje sekvenci multiple BFS hledání zvětšující poloměr vyhledávání postupně
 - končí když uspokojí uživateli požadavky nebo dosáhne maximální hloubky
 - **Directed BFS and Intelligent Search**
 - každý uzel se ptá pouze omezené sady sousedů místo všech sous.
 - je nutné vybrat inteligentně ty "dobré sousedy", kteří můžou potenciálně přispět více relevantními výsledky pro dotaz
 - vybrání dobrých sousedů:
 - vrátili nejvíce výsledků v minulosti
 - nejmenší hop-count zprávy v minulosti
 - forwardovali nejvíce zpráv v minulosti
 - nejkratší fronta zpráv
 - **Local Indices Search**
 - každý uzel udržuje rejstříky pro lokální data a data jeho sousedů, kteří jsou v rádiu k hopů
 - dotazy jsou zpracovány na základě globální politiky, která specifikuje seznam hloubek ve vyhledávacím stromu
 - vysoká cena uložení, vysoká cena aktualizace, nekonzistence
 - **Random Walk**
 - peers náhodně vybírají sousedy pro přijaté dotazy
 - to trvá tak dlouho dokud nenajde výsledek nebo neexpiruje TTL
 - k-walker Random Walk Alg.
 - odeslání k dotazů na náhodně vybrané uzly (místo pouze na jeden)
 - počet zpráv lineárně roste

- Random BFS
 - iniciátor vybere náhodně podmnožinu jeho sousedů
 - počet zpráv roste exponenciálně
 - **Adaptive Probabilistic Search**
 - kombinace techniky k-walker a pravděpodobnostního vyhledávání
 - odeslání dotazů uzlům, u nichž je určitá pravděpodobnost na možné odpovědi
 - podle minulých výsledků se uzlům nastavují pravděpodobnosti
 - když je pravděpodobnost obsažení výsledků vysoká jsou upřednostňovány
 - **Interest-Based Shortcuts**
 - každý uzel přidá link na vrchol vyhledávací sítě pro zlepšení výkonu vyhledávání
 - každý uzel má pouze omezené množství zkratk
- Routing in **Structured** P2P networks:
 - organizace do nějaké fixní topologie
 - ring, multidimensional grid, mesh, multiple list
 - drahá údržba – změny ve směrovacích tabulkách
 - **kategorie:**
 - **Distributed Hash Table**
 - každý peer řídí část globální hashovací tabulky
 - **Chord**
 - prostor je kruh o číslech $0 - 2^m - 1$
 - systém přiřadí klíč K prvnímu uzlu n, jehož identifikátor je roven nebo následuje identifikátor K v kruhu prostoru (hodinově)
 - Vyhledávací algoritmy:
 - simple lookup alg:
 - každý uzel potřebuje znát svého bezprostředního následovníka
 - scalable lookup alg:
 - každý uzel má tabulku M následovníků místo pouze jednoho
 - pokud je nový uzel přidán, musí si najít svou pozici v kruhu a inicializovat tabulku následovníků
 - **Content Address links and neighbor linksable Network**
 - postaveno na d-dimensionálním Cartezském koordinačním prostoru
 - oddíly systému jsou uloženy na rozdílných zónách, které jsou přiřazeny uzlům
 - nový uzel: najde peer, identifikuje zónu, která se má rozdělit, zkonstruuje novou směrovací tabulku a aktualizuje tabulky sousedů
 - **Pastry**
 - systém založen na PRR trees
 - každý uzel má směrovací tabulku
 - data jsou uložena v uzlu, jehož identifikátor sdílí nejdelší prefix s identifikátorem dat
 - kromě směrovací tabulky má každý uzel i množinu listů
 - nový uzel: připojí se k nejbližšímu uzlu ve stromu a obdrží od něj množinu listů

- **Tapestry**
 - podobná Pastry
 - každý směrovací hop rozšiřuje shodný suffix
- **Skip List based**
 - **Skip List structure**
 - struktura pro ukládání seřazených seznamů itemů používající hierarchii lineárního seznamu
 - postaven ve vrstvách
 - vyhledání: začne v head elementu a pokračuje nahoru dokud není roven (nalezen) nebo větší než (opakování procedury) cíl
 - **Skip Graph**
 - postaven na Skip Listu
 - narozdíl od Skip Listu má více než jeden seznam v každé vrstvě
 - The Membership vector – definuje, které udává konkrétní elementy patřící do něj (seznamy jsou seřazeny podle klíče dat)
 - nová uzel: je přidán do seznamu jehož membership vector sdílí stejný prefix
 - **Skip Net**
 - podobný Skip Graph
 - místo do Skip Listu organizuje uzly do kruhu
- **Tree based**
 - **P-Grid**
 - postaven na struktuře virtuálního binárního stromu
 - každý peer je zodpovědný za data, jejichž prefix je stejná jako identifikátor peeru
 - **P-Tree**
 - postaven na virtualním balancovaném B^+ stromu na vrcholu Chord kruhu
 - **BATON**
 - data jsou uložena v listech i vnitřních uzlech
 - mají následující odkazy a sousední odkazy
- Routing in **Hybrid** P2P Networks:
 - silné peery – superpeers a common peers - client peery
 - každý common peer patří nějakému superpeeru
 - client odešle dotaz superpeeru -> superpeer vyhledá, který client nebo superpeer má odpověď -> pošle dotaz superpeeru klienta, který má odpověď, IP adresa peeru s odpovědí je odeslána iniciátorovi
- **Edutella**
 - dotaz směrován superpeeru v HyperCuP network, kde je možné použít suffix směrování
- **Ultrapeers**
 - modifikovaná Gnutella síť s ultrapeery
- **Structured Superpeers**
 - kontrolují určitý rozsah klíčů
 - dotaz s určitým klíčem je pak přeposlán superpeeru, který má na starost rozsah, do kterého klíč patří

	structured P2P	unstructured P2P
routing	based on a routing table	flooding, random walk, ...
lookup possibilities	based on keys only	possibility to ask more complex queries
existing item is always found	yes	cannot be guaranteed
critical part	node join/disconnect	lookup/routing

System	Overlay network	Routing table	Routing method
Gnutella	Unstructured, Random topology	Random neighbors	Breadth First Search with Time-to-Live
FreeNet	Unstructured, Random topology	Random neighbors	Depth First Search with Time-to-Live
Chord	Structured, Ring topology	Neighbors at distances 2^i in the ring	Repeatedly jump to the farthest node in the routing table whose id is still less than the search key
CAN	Structured, Mesh topology	Neighbors at adjacent positions in the mesh	Repeatedly travel through the neighbor that is closer to the destination
Pastry & Tapestry	Structured, PRR tree topology	Neighbors sharing common prefix identifier at different levels	Repeatedly forward the message to the neighbor having the longest matching prefix identifier
Viceroy	Structured, butterfly topology	Five neighbors: one at the upper level, two at the lower level, and two at the same level	Three steps: going up, going down, and vicinity search
Crescendo	Structured, hierarchical ring topology	Chord-like neighbors at different ring levels	A combination of Chord-like routing and the routing between rings at different levels

System	Overlay network	Routing table	Routing method
Skip Graph	Structured, multiple linked lists topology	Neighbors sharing common prefix membership vector at different lengths	Travel from the highest to the lowest level of the list. At each level, jump to the neighbor closer to the destination if such a neighbor exists
SkipNet	Structured, hierarchical ring topology	Neighbors are predecessors and successors at different ring levels	Skip Graph-like routing, traveling from the highest to the lowest level of the ring.
P-Grid	Structured, binary tree topology	A neighbor at the other side of the tree rooted at each internal node from the root to the leaf	Travel from the root to the leaf. At each level, jump to the neighbor closer to the destination
P-Tree	Structured, a combination of a B^+ -Tree and a Chord ring topology	Neighbors are nodes in the left-most root-to-leaf path of the B^+ -Tree	Travel from the root to the leaf. At each level, jump to the neighbor closer to the destination
BATON	Structured, balanced tree topology	Neighbors are parent, children and Chord-like neighbors at the same level	If not having full routing tables, go to parent. Otherwise, go to the neighbor or the child closer to the destination
Edutella & Ultrapeers	Hybrid, a combination of structured and unstructured topology	Neighbors exist only at superpeer level. At client side, each client peer connects to a superpeer	A client peer always routes its requests to its superpeer while routing at supper peer level depends on the topology employed at that level

Wireless Ad-hoc networks

- síť bez infrastruktury
- síť pro speciální účel
- kolekce autonomních uzlů, které spolu komunikují pomocí multihop radio sítě
- zachování konektivity decentralizovaným způsobem
- výhody:
 - rychlá konstrukce
 - odolnost – nemá jeden bod selhání
 - účinnější než celulárních sítě
- problémy:
 - nedostatek centrální entity pro organizaci sítě
 - limitovaný dosah bezdrátové komunikace
 - pohyblivost účastníků
 - **MANET** - mobile
- možné aplikace:
 - nalezení prázdného parkovacího místa
 - pomoc při pohromách
 - osobní síťování (brýle, hodinky, pda)
 - military networking
- **VANET – vehicular**
 - technologie pro pohybující se automobily
 - pohyb aut má určitý trend (směr)

	Infrastructure-based network	Ad hoc network
Prerequisites	Pre-deployed infrastructure, e.g. routers, switches, base stations, servers	None
Node properties	End system only	Duality of end system and network functions
Connections	Wired or wireless	Usually wireless
Topology	Outlined by the pre-deployed infrastructure	Self-organized topology maintained by the nodes
Network functions	Provided by the infrastructure	Distributed to all participating nodes

- **Wireless Sensor Networks**
 - interakce s prostředím
 - aplikace:
 - ambulantní operace - shazení senzoru do ohně a vytvoření teplotní mapy
 - monitorování zvěře
 - agro – hnojení pouze když je potřeba
 - inteligentní budovy – snížení plýtvání ener., monitorování po zemětřesení
 - HW:
 - procesor
 - radio vysílač/přijímač
 - baterie
 - senzor – světelný, teplotní, pohybový

- SW – OS řízený událostmi
- důležitý aspekt je **úspora energie**

Resources and properties	MANET	WSN
Available energy	High	Low
Processing power	High	Low
Memory and storage	High	Low
Density and scale	Low	High
Mobility	High	Limited*
Heterogeneity	Medium*	Low*
Varying user demands	High	Low

Difference	P2P Network	MANET
Motivation for creating the network	Logical infra structure to provide a service	A physical infra-structure to provide connectivity
Connection Between two nodes	Fixed medium and direct	Wireless and indirect
Connection confidence	High (physical connections)	Low (wireless connections)
Peer location	Any Internet point	Restricted area
Structure	Physical apart from logical structure	Physical structure corresponds to logical structure
Routing	reactive	Proactive, reactive
Peer behavior	Fixed	Mobile
Broadcast	Virtual, multiple unicasts	Physical, to all nodes in transmission range area

Medium Access Control

- minimalizace kolizí
- není možné vysílat a přijímat zároveň
- protokoly:
 - **Contention-based protocols**
 - no priori resource reservation
 - možnosti jak umlčet odesílatele:
 - odesílatel informuje potenciální zdroje rušení **zatímco** probíhá příjem
 - Busy Tone Multiple Access protocol (
 - odesílatel informuje potenciální zdroje rušení **před** příjmem
 - Multiple Access Collision Avoidance (MACA protocol)
 - **Contention-based protocols with reservation mechanisms**
 - use mechanisms for reserving bandwidth a priori
 - synchronous (require time synchronization among all nodes in the network) and asynchronous (usually rely on relative time information)
 - e.g., MACA/PR (MACA with Piggy-Backed Reservation)
 - **Contention-based protocols with scheduling mechanisms**
 - focus on packet scheduling at nodes and also scheduling nodes for access to the channel
 - priorities
 - LEACH (Low-Energy Adaptive Clustering Hierarchy), SMACS, TRAMA
 - Power-Control MAC
 - řízení přenosu změnou energie
 - handshake je odeslán s maximálním možným výkonem
 - vypočítá se žádoucí výkon jako $P_{max} / P_r * R_x * c$, kde P_r je úroveň energie přijímače
 - Sensor-MAC
 - vylepšení energetické účinnosti
 - periodická změna mezi nasloucháním a uspaním

Ad-hoc Routing

- Address-based routing
- Data-centric forwarding
 - směrování podle sémantiky přenášených dat

	Address-based routing	Data-centric forwarding
Routing approach	Identification of a path according to the destination address of the data message	Determination of the destination of a data message according to the content of the packet
Prerequisites	Network-wide unique addresses	Pre-defined message types and semantics
Routing techniques	Proactive routing (continuous state maintenance) or reactive routing (on-demand path finding)	(probabilistic) flooding schemes or interest-based reverse routing
Advantages	Usually low delays in connection setup and data dissemination	No address information required and simplified self-management and redundancy
Disadvantages	Network-wide unique address identifiers required	Increased overhead for single transmissions

Klasifikace směrovacích protokolů:

- **Proactive**
 - najdou routery předtím než jsou potřeba
 - periodická výměna informací (link-state, table-driven (distance vector))
 - **Destination Sequence Distance Vector**
 - periodická výměna celých směrovacích tabulek
 - **Optimized Link State Routing**
 - links' state informace jsou floodovány celou sítí
- **Reactive**
 - objevují routery až v případě potřeby
 - **Dynamic Source Routing**
 - používá Route Request a Route Reply
 - flooding přes celou síť
 - **Ad Hoc on Demand Distance Vector**
 - všechny uzly si pamatují odkud paket přišel a upravují podle toho své směrovací tabulky
 - pokud nějaký mezilehlý uzel zná cestu k cíli, odpoví odesílateli přímo on
 - **Dynamic MANET On Demand**
 - následovník AODV
 - snižuje zatížení v nastavení cesty a údržbu cest
- **Table-driven**
 - každý uzel zná pouze next hop cíl
- **Source-routing**
 - uzel zná kompletní cestu k cíli
- **Flat**
 - všechny uzly používají stejný algoritmus
- **Hierarchical**
 - některé uzly mají přidanou odpovědnost
- **Location-based**
 - používá uzly podle fyzické lokace – nutnost GPS
- **Non-location-based**
 - nehledí na fyzickou lokaci

- Speciální směrovací protokoly:
 - Geographic Routing – podle fyzické polohy uzlů
 - Energy-Aware Routing Protocols – bere v úvahu energetickou kapacitu uzlů

Závěr

- Ad-hoc se hodí pro dočasné i fixní sítě, mají dobrých vlastností oproti infrastrukturovaným sítím

Computer Networks & Multimedia

- **multimedia applications**
 - powerful technologies that can enable remote sharing of resources or interactive work collaborations
 - ex.:
 - video conference, multimedia electronic mail, web TV, radio/TV broadcast
 - kompresní algoritmy
 - ztrátová
 - bezztrátová
- media characteristics
 - **text** – FTP, HTTP, SMTP
 - některé vyžadují TCP jiné UDP
 - kódování – Shannon-Fano (variable length code), Huffman coding (podobný Shannon-Fano), LZW (nahradí řetězce jedním kódem, Unix Compress (používá LZW s rostoucím slovníkem)
 - **audio** = zvuk/řeč converovaná do digitálního signálu
 - sampling – redukce analogového signálu do diskrétního
 - skenování vzorků
 - quantization techniques – proces aproximace vzorků diskrétními symboly
 - požadavek na šířku pásma
 - tolerance ztráty 1-2% paketů
 - aplikace mohou mít real-time požadavky 100-200 ms
 - **graphics and animation**
 - aplikace nemají real-time požadavky
 - **video** - sekvence obrázků
 - obvykle vysoké požadavky na šířku pásma
 - error requirements and real-time characteristics similar to the audio ones⁷

Klasifikace médií

- **real-time**
 - **discrete media**
 - *text chat, instant messaging*
 - **continuous media**
 - **delay tolerant**
 - *streaming audio/video*
 - **delay intolerant**
 - *interaktivní audio/video*
 - *ovládání vzdálené plochy*
- **non real-time**
 - *text, data*
 - *obrázky*

Multimedia Requirements on the Communication Network

- **traffic requirements** – limity na real-time parametry (delay, chvění, bandwidth a spolehlivost)
- **functional requirements** – podpora multimediální služby (multicasting, bezpečnost, mobilita a řízení sezení)

- **Traffic Requirements:**

- Delay

- Packet Processing Delay - a constant amount of delay faced at both the source and the destination
 - Packet Transmission Delay - time taken by the physical layer at the source to transmit the packets over the link
 - Propagation Delay - the flight time of packets over the transmission link
 - Routing and Queuing delay - the best-effort Internet treats every packet equally)

- Bandwidth

- multimedia traffic streams have usually high bandwidth demands

- Error Requirements

- different media types = different error requirements
 - error correction:
 - **Sender-based Repair**
 - **active** repair - Automatic Retransmission Request
 - **passive** repair
 - **Forward Error Correction** – přidání informace
 - Media-independent FEC - uses block or algebraic codes to produce additional packets which help in loss recovery (parity coding)
 - Media-dependent FEC - each data unit is sent in multiple packets
 - **Interleaving** – pokud je velikost jednotky media menší než paket a pokud nezáleží na zpoždění
 - based on resequencing media units before transmission
 - **Receiver-based Repair**
 - **Error Concealment (zatažení)** - predicting/compensating the lost information from correctly received packets

- **Functional Requirements**

- Multicast Support

- delivery of a message/information to a group of destination computers simultaneously in a single transmission from the source, creating copies automatically in inner network elements (e.g., routers) only when the topology of the network requires it
 - protokoly: IGMP, DVMRP

- Bezpečnost a mobilita

- Integrita
 - Autenticita
 - Šifrování
 - zajištěné IPv6 – mobilita i bezpečnost

- Řízení sezení (Session Management)

- **Media Description** - umožňuje distribuované multimediální aplikace pro distribuci relačních informací
 - Session Description Protocol - describing streaming media parameters
 - **Session Announcement**
 - Session Announcement Protocol - used for advertising multicast conferences and other multicast sessions

- **Session Identification** - a multimedia session often consists of multiple media streams
 - Real-Time Transport Protocol (běží na vrcholu UDP)
 - provádí:
 - Sequencing
 - Payload identification
 - Frame indication
 - Source identification
- **Session Control**
 - Real-Time Control Protocol - provides out-of-band statistics and control information for an RTP flow
 - Real-Time Streaming Protocol - an out-of-band control protocol that allows the media player application to control the transmission of the media stream