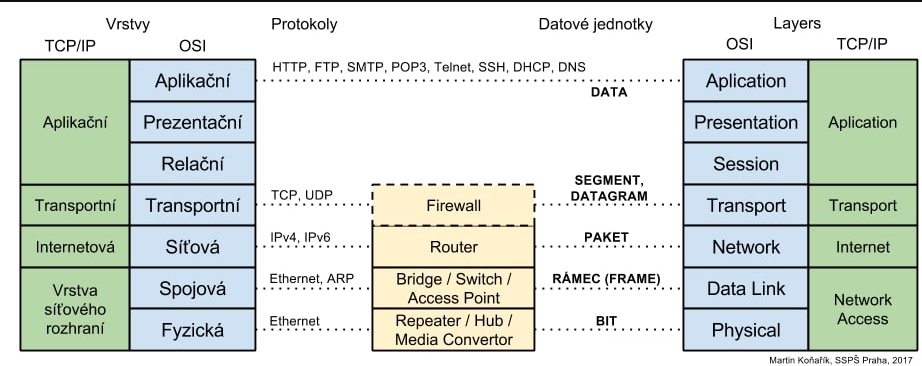
1. Přednáška – úvod



**Anycast** – Například DNS (Více lidí ví informaci a obdrží request, ale stačí pouze jedna odpověď a komunikace je úspěšná) – očekává se odpověď na rozdíl od jiných druhu komunikace

**Multicast** – Více příjemců

**Unicast** – Jeden příjemce

**Broadcast** – Celá síť (Posílá se na nejvyšší adresu sítě)

IPv4 – 32 bitů (4 byty)

IPv6 – 128 bitů

Maska vlastně definuje, kolik bitů je pevně daných. Volitelných (možných stanic) je 2 na 32 – maska = PREFIX.

* Např.: Pro 17 počítačů bude /27 = prefix (32 adresních míst)
  + Jiné vyjádření prefixu = maska v tomto případě je vlastně 27x1 bitů (Počet shodných bitů) a pak 5x0
  + 11111111 11111111 11111111 11100000
  + 255.255.255.224
  + 1. Adresa = adresa sítě
  + Poslední adresa = broadcast
  + = GW

Adresa sítě = <nejnižší možná adresa/prefix>

1. Přednáška – linková vrstva (2. vrstva)

Switche + Bridge

**Podvrstvy:**

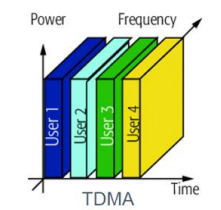
* Fyzické adresování – MAC podvrstva
  + Předávání dat fyzické vrstvě
  + Přístup k médiu, **určení VELIKOSTI & KONCE rámce**
  + Fyzická adresace (MAC Adresace – má 48 bitů), sdílený přístup k médiu se řeší pomocí multiplexu
* Zajišťuje logické řízení toku, kódování, spolehlivé doručení rámců – LLC podvrstva (logical link control)

**Kolizní doména** = množina stanic, které sdílí společné médium, kde současné vysílání 2 a více stanic vede ke kolizi

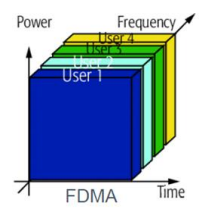
## Druhy multiplexu

(multiplex = technika umožňující současné použití stejného přenosového média více účastníky (stanicemi, počítači)

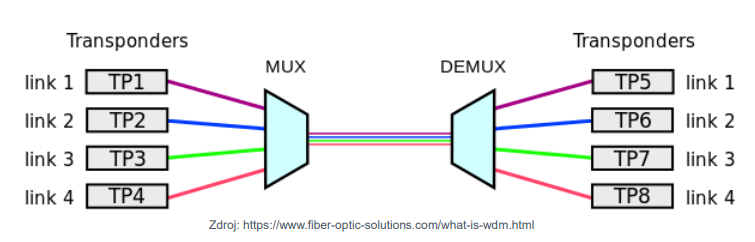
* **Časový multiplex (TDMA)**
  + Výhoda – 100% jistota, že bude konstantní odezva – že na vás bude řada, ale zase je odezva vyšší
  + Nevýhoda – Maximální počet účastníků



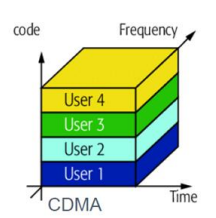
* **Frekvenční multiplex (FDMA)**
  + Výhoda – mohou všichni komunikovat najednou, nižší odezva
  + Nevýhoda – Také maximální počet účastníků



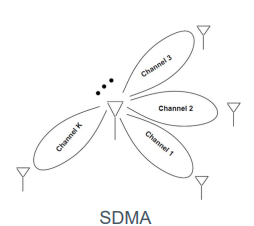
* Wavelength Division multiplex (WDM)
  + Multiplexor a demultiplexor, využívá se jedno vlákno pro více vlnových délek (pro různé barvy)



* Kódový multiplex (CDMA)
  + Médium využíváno mnoho lidmi, účastnící však zpracovávají pouze to, čemu rozumějí – svému kódu



* Prostorový multiplex (SDMA)
  + Často využívají telefoní operátoři při vysílačích na budovách
  + Prostorové oddělení směrů, kterými jednotlivé stanice vysílají



Metody s příposlechu přenosového média

Naslouchání stavu přenosového média (Carrier Sense Multiple Access), čeká se náhodnou dobu než se znovu odešlou data (předchází perioditicitám)

* CSMA/CD (Collision detection – detekuje kolizi)
  + Vysílání JAM signálu, pokud se detekuje kolize, který informuje účastníky o kolizi
  + Kolize stejně může nastat, protože čas, než se informace odvysílá (Kolizní okénko)
* CSMA/CA (Collision avoidance – dokáže se kolizi vyvarovat)
  + Hlavně u WiFi, jednodušší než CSMA/CD, účastníci se o kolizi nemusí vůbec dozvědět

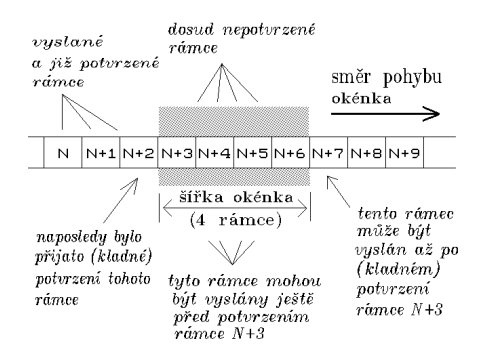
## Potvrzovací schémata

Jednotlivé potvrzování (Stop&Wait)

* Až po přijetí zprávy o přijetí se vysílá další informace

Klouzavé okénko (Sliding Window)

* Posouvá se okénko dle přijatých (1. prvek okénka je hned ten následující po posledním přijatým)



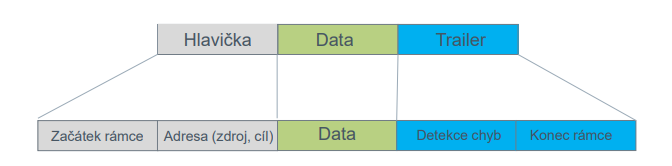
Také existují schémata, které posílají informace furt a pokud se vrátí negativní potvrzení (chybí paket) tak se ten daný paket dopošle.

## Kontrola, detekce a oprava chyb

Bezpečnostní kódy – například Hammingovy kódy (7 bitů datových, 4 redundantní) – Samoopravný ( pro 1 chybu), Detekční (pro 2 chyby)

Další způsob je také CRC – Cyclic redundancy code (kontrolní součet)

## Podoba RÁMCE



MTU = Maximální množství dat v rámci

## Switch (Přepínač)

Mnoho portů (s buffery), rozhoduje se na základě přepínacích tabulek (switching table). Přijme rámec a dle tabulky se na základě MAC adres rozhodne, kam zaslat paket.

Tabulka se postupně učí – když nezná, pošle všude, čeká na odpověď a ten port odkud přijde si napáruje s danou adresou

* **Store-and-forward metoda práce**
  + Přepínač přijme celý rámce, uloží jej a až poté odešle
* **Cut-through**
  + Přepínač načte jen cílovou adresu (typicky 6 bajtů) z hlavičky rámce.
  + Překontroluje ji a odesílá data okamžitě.
* **Fragment-Free**
  + Jako cut-through, ale čte více informací (celou 64bajtovou hlavičku), odhalí více chyb

## Bridge

Bridge v dnešní době trochu podmnožinou switchů – Propojuje 2 části sítě. Umí být SW (na úrovni OS) i HW.

Mají méně portů, propojují většinou dvě sítě a neobsahují buffery

Broadcastová doména = množina všech stanic dané sítě, kterým je doručen rámec s všesměrovou (broadcastovou) adresou.

## Ethernet

MAC Adresy na prvních 3 bitech obsahují informaci o výrobci, broadcast = FFFF FFFF

Běžná velikost přenášených dat v Ethernetu je rovna maximální (1500 bytů). Není uvedena v rámci!

Ethernet neobsahuje žádný mechanismus (např. klouzavé okénko), který by garantoval spolehlivé doručení dat)

**Autonegociace**

* Metoda, kterou si vysílač a přijímač dohodnou parametry přenosu (max. průtok, použité kódování…)
* Domlouvá se pomocí speciálních pulzů nebo specifické rámce

Při zapojení kabelů, tak se jeden pár vodičů využívá pro vysílání a druhý pro příjem.

* PC-PC -> Zapojení pomocí kříženého kabelu
* PC-Switch -> Přímý kabel

## VLAN

Cílem VLAN je bez nutnosti fyzických přepínačů oddělovat sítě. Nastaví se to softwarově přímo na přepínači, který z portů jsou jaká VLAN. Typické využití = oddělení pracovišť ve firmě, ale jednodušší správa.

Trunk port = port, který odbavuje provoz více různých VLAN

Dělení VLAN sítí může být dle: portů, adres, protokolů, tagů.

Switch může umět také tagy při odeslání ke koncové stanici odebrat (např.: když to stanice neumí) a poté zpět přidávat při odchodu trunk-portem

**Přeposílání dat mezi VLAN má na starost ROUTER – vrstva síťová.**

1. Přednáška - Síťová vrstva (3. vrstva) – IPv4

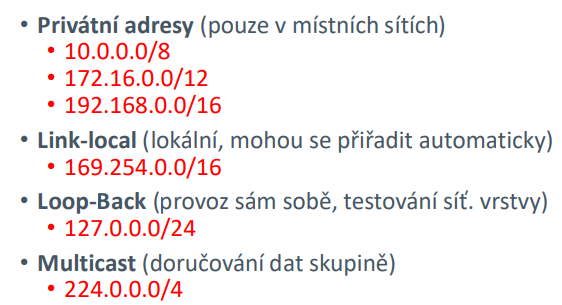
Privátní adresy mohou být přidělené v mnoha sítích stejné, jelikož se s vnějškem komunikuje přes veřejné IP

NAT = router, když nahrazuje privátní adresu z vnitřní sítě na veřejnou IP adresu pro vyslání do světa

* Na to je NATovací tabulka

## IPV4

**Linkové adresy** jsou použitelné pro doručování dat jen v lokální síti. **Síťové adresy** jsou použitelné jak v lokální síti tak i mezi různými sítěmi.



Paket je základní jednotkou přenosu dat na síťové vrstvě

* **Hlavička IPv4 paketu nemá na rozdíl od hlavičky ethernetového rámce fixní velikost**
* Pakety se mapují do ethernetového rámce jako data, nutností pro dororučení je znalost cílové MAC adresy (zjišťování na základě znalosti IP adresy zajišťuje protokol ARP)
* **Hlavička obsahuje**
  + Verze protokolu, Délka paketu v bajtech, Identifikace (v případě fragmentace), Příznaky řídící fragmentaci, offset fragmentu (pořadí od začátku) TTL (Time To Live), CRC součty, Adresa odesílatele + příjemce, DATA
* TTL se snižuje pouze při průchodu Routerem.
* Doručování dat přímo mezi routery provádí linková vrstva
* Pakety se přeposílají díky routovacích tabulek, kde se v případě přímo připojených sítí posílají data přímo na koncovou stanici / jinak na další router (next hop)

Fragmentace může nastat a měnit se při průchodu přes libovolný router během směrování (záleží na tom, jak po cestě zrovna ethernetové rámce jsou velké) -> složení datagramu nastává až na cílové stanici

**Dynamická konfigurace IP adres se provádí pomocí protokolu DHCP (port 67 – server, port 68 – klient)**

## ICMP

Sloužící především k ověření funkčnosti vzdáleného zařízení. ICMP Request (code 8), ERR (11), REPLY (0)

Využíván pingem a traceroute/tracert -> funguje přes ICMP request s inkrementálně se zvětšujícím TTL

## ARP

V lokální síti probíhá doručování dat na základě linkových (MAC) adres (Ty musí stanice znát)

* ARP je důležitý, aby bylo možné doručit data mezi počítači v lokální síti jen na základě znalostí IP adres -> mapuje dle znalosti IP adresy MAC adresy

Proces dotazu

* Broadcast na úrovni ethernetového rámce (0xFFFF…).
* Žádost cílovou stanicí zachycena, vyplní do dat odpovědi svou MAC adresu -> odesílá odpověď (již unicastově, jelikož ví cílové stanice MAC)
* Odesílatel si z odpovědi přete MAC a namapuje si ji k dané IP adrese

**ARP Spoofing**

* Situace, kdy útočník odpoví dříve než očekávaný příjemce a vydává se za příjemce -> útočník získá veškerou kontrolu na daty které jsou mezi odesílatelem a příjemcem. Není tomu zamezeno na úrovni šifrování dat

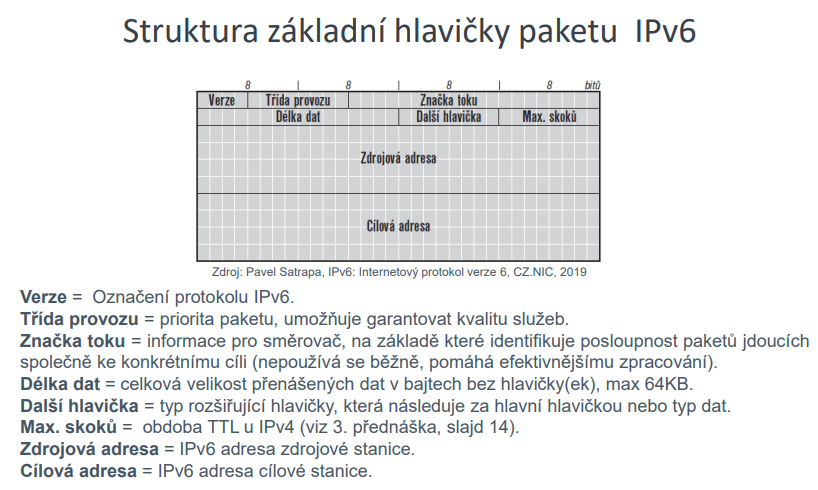
**Proxy ARP**

* Mechanismus, díky kterému se může router chovat pro zdrojovou stanici jako switch
* Dá se říct, že se router chová jako „útočník“ v ARP spoofingu, jelikož tvrdí stanici, že je daným počítačem v jiné síti (Pro odesílatele jako by to byla stejná síť)

1. Přednáška - Síťová vrstva (3. vrstva) – IPv6

## IPV6

* Protokol umí sám o sobě metodu automatické konfigurace
* IPv6 adresa obsahuje 128 bitů (8x16bitů hexadecimálně a odděleno : )
* Zkracování IP adres pomocí vynechání bezvýznamných nul
* Neexistuje broadcastová adresa, jsou nahrazeny speciálními multicasty
* NAT už není třeba -> dostatek veřejných IP adres
* Zřetězování hlaviček jako spojáků (rozšiřujících hlaviček) – na rozdíl od IPv4 může obsahovat více hlaviček
* Velikost fragmentu paketu určuje na rozdíl od IPv4 zdrojová stanice (u IPv4 router) a je konstantní pro celou dobu přenosu (žádný router ji nezmění)



* **IGA –** Ekvivalent veřejných IP adres u IPv4 (směrovatelné v internetu)
* **Link-Local Adresa-** Používá se pro adresaci konkrétního síťového rozhraní v lokální síti (fe80::/10)
* IPv6 dokáže zabezpečit svoji činnost pouze s využitím jediného protokolu (Internet Control Message Protocol for IPv6 – ICMPv6) -> zásadně jiné než normální ICMP
  + Sada funkcí pro řízení provozu, která se označuje jako objevování sousedů (Neighbor Discovery)

## Objevování sousedů – ICMPv6

Nabízí řadu funkcí:

* Automatickou konfiguraci koncových stanic (DHCP v IPv4)
* Zjišťování linkových adres v lokální síti (ARP v IPv4)
* Detekce duplicitních adres, přesměrování atd.

**Automatická konfigurace koncové stanice:**

* **Bezstavová automatická konfigurace:**
  + Založena na myšlence, že v síti existují směrovače, které mají kompletní informace týkající se provozu a dokáží je předat dále
  + Periodické / na výzvu zasílání ohlášení směrovačů (Router Advertisment), které obsahují konfiguraci sítě
  + Výzva sousedovi – dvoustupňové mapování – Jedná se o mapování známé IP adresy k MAC adrese přes multicast -> MAC adresa se odvodí z posledních 32 bitů IPv6 adresy
* **Stavová automatická konfigurace:**
  + V síti existuje specializovaný server, který zajišťuje přidělování adres, používá se protokol DHCPv6 (obdobné jako v IPv4) -> Discover atd. je také stejný
  + DHCPv6 PŘIDĚLUJE POUZE ADRESU (Zbytek zařizuje metoda ND)

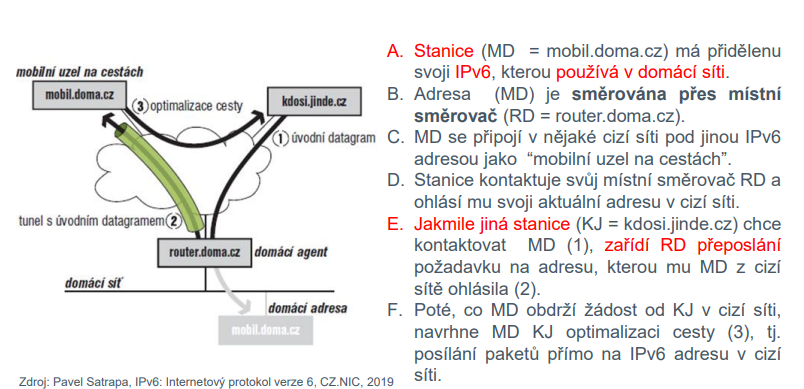
V sítích IPv4 mapování IP adres na MAC adresy zajišťuje protokol ARP, pro sítě IPv6 již ARP však použít nelze (není definován), proto zjišťování linkových adres je zahrnuto do protokolu ICMPv6

**Algoritmus zjištění linkové adresy na základě známé IPv6 adresy:**

1. Sestavení adresy vyzývané cílové stanice zdrojovou stanicí (SA = Solicitation Address). Jedná se o skupinovou adresu, odvodí se ze známé IPv6 adresy
   1. Pro zjišťování linkových adres se používá specifický skupinový prefix FF02::1:FF00:/104
2. Poslání zprávy Výzva sousedovi (NS = Neighbour Solicitation) skupinově na SA prostřednictvím linkové vrstvy
3. Přijetí zprávy NS cílovou stanicí a zaslání odpovědi prostřednictvím zprávy ohlášení souseda (NA = Neighbor Advertisement)
4. Načtením MAC adresy souseda zdrojovou stanicí a uložení této adresy do tabulky sousedů

## Vybrané technologie a principy u IPv6

* **Inverzní objevování sousedů**
  + Metoda dokáže pro známou linkovou adresu konkrétního rozhraní odvodit odpovídající adresu IPv6
  + Metoda posílá žádost prostřednictvím zprávy Inverzní výzva sousedovi skupinově na FF02:1 a přijímá odpověď unicastově prostřednictvím zprávy Inverzní ohlášení souseda
* **Směrování v sítících využívající IPv6**
  + Stanice si sami zjišťují dostupné routery automaticky pomocí ND.
  + Stanice si udržují směrovací tabulky, implicitní směrovač = default gateway
  + Z ohlášení směrovače / díky DHCPv6 se stanice dozví, který směrovač je implicitním
* **Přechodový mechanismus IPv6 pro přístup do IPv4**
  + Cílem je umožnit zařízení ze sítě s IPv6 dosáhnout zařízení v síti IPv4
  + Řešení se nazývá NAT64
  + Podmínkou je, že v síti existuje router na kterém běží současně IPv4 i IPv6 protokol
  + Pro mapování se využívá speciální prefix, ke kterému se poté přidá 32 bitů z IPv4 adresy
* **Technologie TEREDO**
  + Technologie umožňuje komunikovat stanici přes IPv6, ačkoli se tato přímo nenachází v síti podporující IPv6
  + Kontaktování TEREDO serveru -> Tvorba tunelu na TEREDO Relay (TR)-> Komunikace počítače s TR
* **Mobilita u IPv6**
  + Vydávání se za to, že vlastně komunikujeme z domácí sítě, ačkoliv jsme připojeni jinde

****

1. Přednáška - síťová vrstva (směrování v počítačových sítích)

* Přeposílat data v lokální síti je snadné -> vše zařídí přepínače, ovšem mezi sítěmi oddělenými směrovači je potřeba hledat cesta

## Druhy směrování

**ISP = Internet Service Provider** = poskytovatel připojení s přidělenými specifickými IP rozsahy

**Optimální (z hlediska nějakého kritéria = metriky)**

* Nejkratší (nejméně hopů), Nejlevnější (pokud něco je finančně zpoplatněné)

**Redundantní – Multi-path**

* Provoz může být veden různými cestami. Hodí se pro zálohování cest popř. rozdělování zátěže

**Symetrie**

* Jestli cesta tam a zpět je nebo není stejná (symetrické/asymetrické)

**Dle způsobu hledání cesty**

* Záplavové (nejjednodušší, ale používá je řada technologií a prokolů)
  + Začíná ve zvoleném počátečním uzlu -> osloví všechny sousední uzly s požadavkem na směrování, požadavek opatřen ID (Hodí se když záplava začne například z více míst)
  + Každý uzel pak při přijmu požadavku kontroluje, zda neobdržel požadavek s tímto ID, když neobdržel, tak posílá všem sousedům kromě toho, od kterého to přišlo
  + **Je to trochu takový BFS**
* Proaktivní (Cesta je předpočítána a uložena ve směrovacích tabulkách)
  + Využívá se v sítích, co se příliš nemění – NEJBĚŽNĚJŠÍ ZPŮSOB
* Reaktivní (Cesta se vytváří až když je potřeba) -> adhoc či mobilní sítě (proměnlivá topologie)
  + Nepoužívají se směrovací tabulky
  + **2 fáze:**
    - 1) Nalezení cesty – většinou záplavovým směrováním
    - 2) Posílání dat

## BFS

* Prohledávání do šířky = po vlnách se rozšiřuje informace
* Uzly – fresh, open, closed
  + Fronta-based, to co je aktuálně ve frontě je open, nedotčené uzle jsou open a pokud už jsem přešel na sousedy těch open uzlů, tak jsou jakože closed
* Jakmile zapojím i vzdálenosti, tak podle nejnižší vzdálenosti je vkládám do fronty (lowest=first), udržuju nejkratší cesty, zbytek tras do stejných bodů zahazuju
* Ve výsledku vznikne kostra grafu = ty hrany, co zaručují nejkratší doručení

## Statické vs Dynamické směrování

V dalším výkladu pod pojmem směrování budeme chápat **pouze proaktivní směrování**

**Statické směrování**

Vlastnosti:

* Funguje okamžitě po zapnutí směrovačů
* Není schopné reagovat na změny
* Vyžaduje manuální konfiguraci, vhodné pro malé sítě

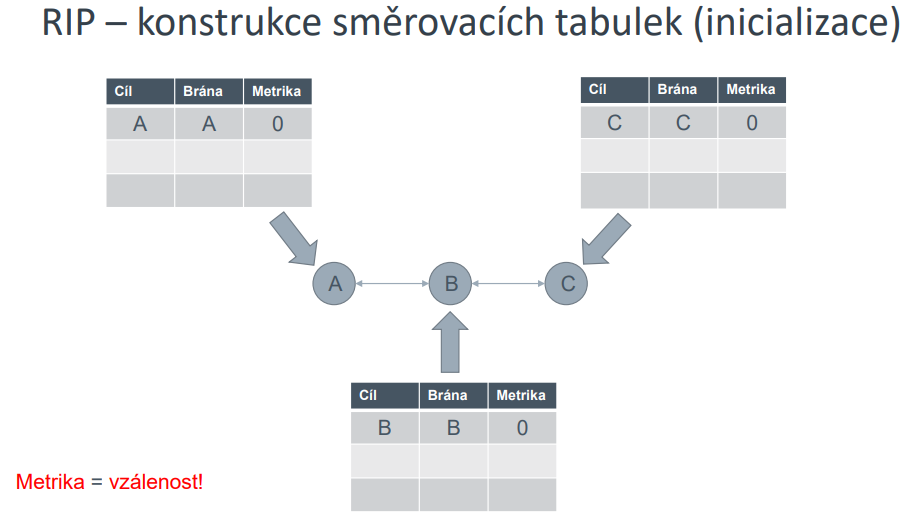
**Dynamické směrování**

Vlastnosti:

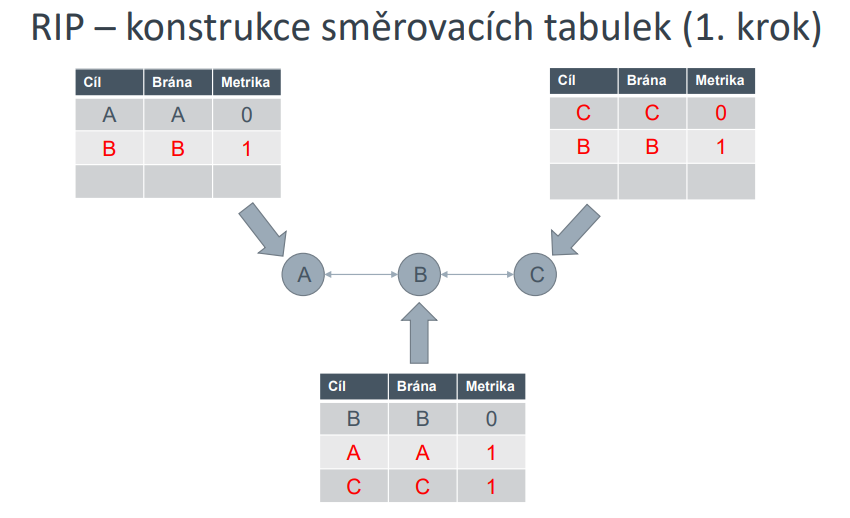
* Může trvat desítky vteřin ve větších sítích, ovšem dobře reagují na nové zařízení v síti apod.
* Rozlišujeme 3 druhy dynamických směrovacích algoritmů (A), které jsou pojmenovány dle obsahu zasílaných zpráv a sice:
  + Distance Vector Algoritmy (DVA)
  + Link State Algoritmy (LSA)
  + Path Vector Algoritmy (PVA)

## Distance vector algoritmy (DVA)

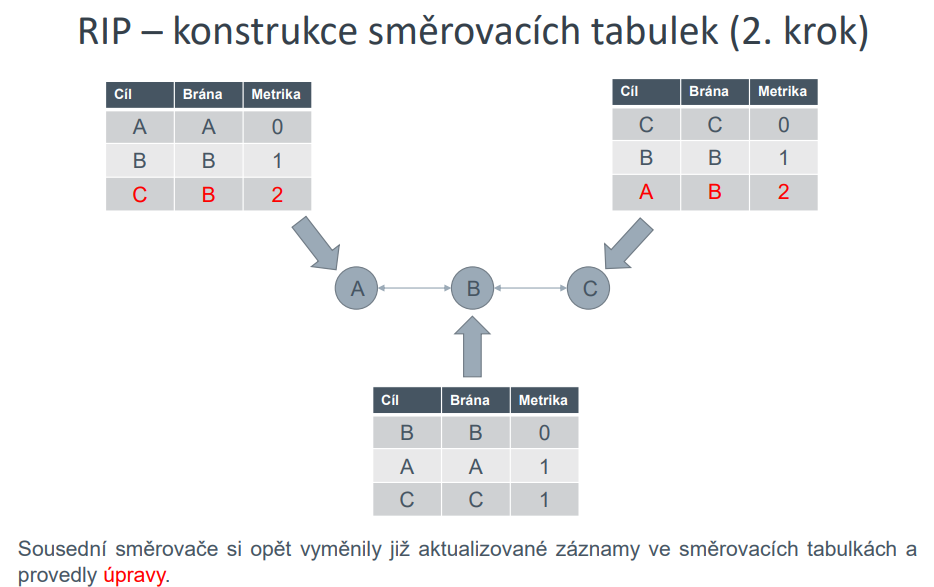
* Směrovače si vyměňují navzájem záznamy svých směrovacích tabulek (vektory hodnot vzáleností) prostřednictvím zpráv (M). Metrika je vzdálenost
* Pro optimalizaci záznamů používají distribuovanou verzi **Bellman-Fordova Algoritmu** (DBFA).
  + Funguje to tak, že se dva routery baví. Router 1 má cestu k cíli která je X daleko. Router 2 má nějakou cestu k cíli, co je Y daleko. Router 1 si spočítá cestu k Router 2 + Y a pokud je rychlejší, než jeho aktuální cesta k cíli (X), tak ji nahradí právě touto cestou přes Router 2. -> Výchozí brána pro cíl bude Router 2.
* Tento algoritmus se používá v **RIP (Routing Information Protocol)**
* Inicializace (spuštění) + 1. Iterace



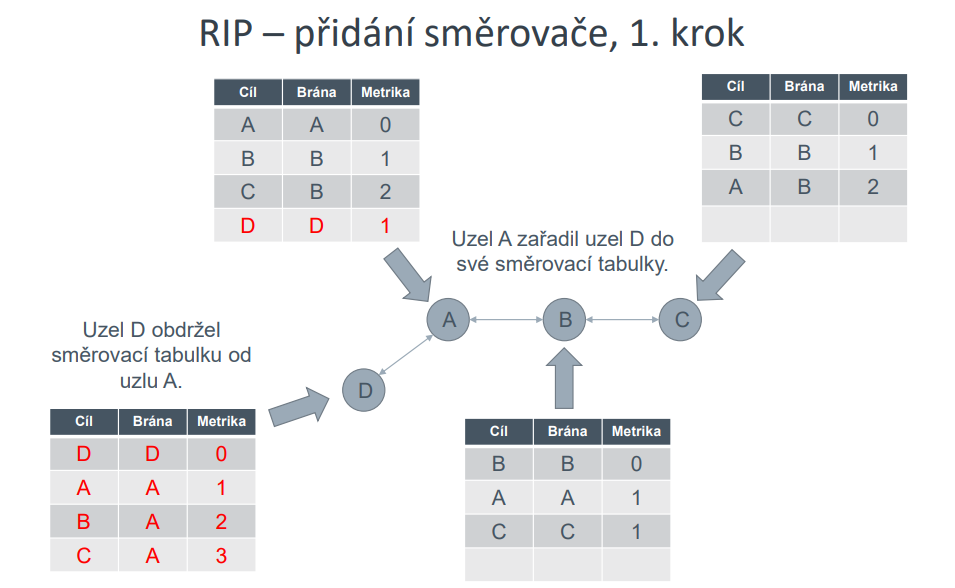
* + Řeknou sousedním routerům o tom, kde oni sami jsou. Sousedi si tedy uloží cestu k nim jako A | 1



* 2. Iterace -> Výměna informací



Při přidání nového směrovače se informace ihned získají od nejbližšího routeru. Postupně se pak také rozšíří k dalším routerům



V případě poruchy linky, tak se to postupně dozví všechny routery a nastaví cestu k danému routeru, kde chyba nastala, na nekonečno.

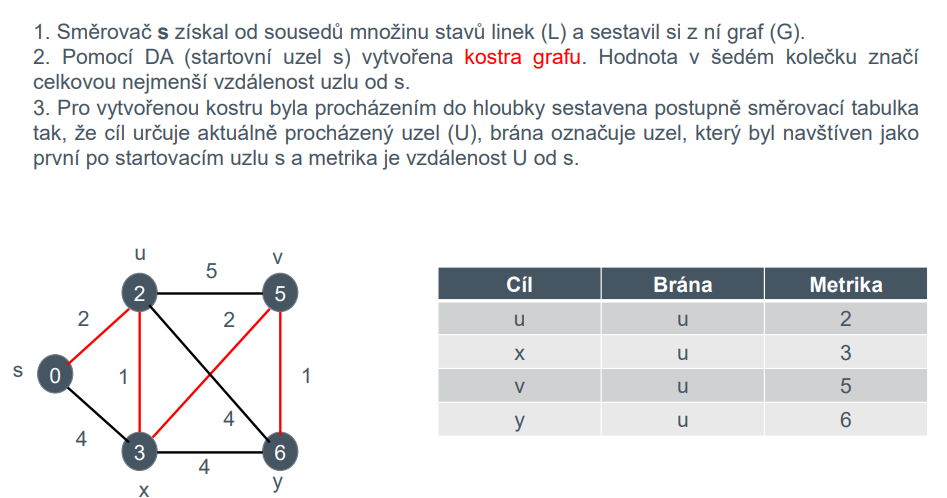
## Link-State (LS) algoritmy (A)

Méně náročné oproti DVA na zatížená linky výměnou dat. U LSA pouze krátké zprávy o stavu linek. Zase ale náročnější na CPU – změna libovolné linky vyvolá nové prohledávání.

**Specifikace:**

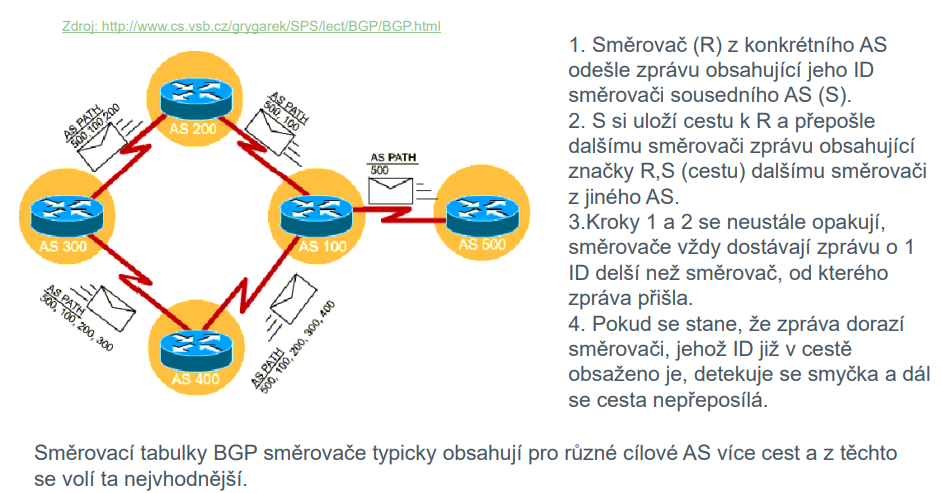
* Linka = propojení dvou sousedních směrovačů (X a Y) s určitým ohodnocením (K)
* Spojením všech linek dohromady vznikne graf sítě (G)
* Směrovače si vyměňují navzájem stavy linek a z těchto si každý směrovač vytvoří vnitřní reprezentace ohodnoceného grafu sítě (G)
* Směrovač nad G hledá kostru, která obsahuje nejkratších cesty (KNC). **Pro nalezení KNC se používá Dijkstrův algoritmus (DA).**
* Z nalezené kostry se procházením do hloubky **(DFS**) sestaví postupně směrovací tabulka.

**Implementace LSA algoritmu:**

* OSPF (Open Shortest Path First) využivá Dijsktrova algoritmu. Využíván výhradně pro vnitřní směrování.
* 

## Path Vector Algoritmy (PVA)

* Směrovače si vyměňují kompletní cesty ke konkrétním cílům.
* Implementace PVA:
  + Využíváno u BGP (Border Gateway Protocol)
  + BGP však namísto směrovače předpokládá autonomní systém (AS)
  + Cesta u BGP = Posloupnost směrovačů jednotlivých AS k cíli
  + Používá se především pro vnější směrování
* Vytváření cesty u BGP



## Vnitřní vs Vnější směrování

* **Autonomní Systém (AS)** = skupina všech IP adresních rozsahů konkrétního ISP. Každý AS má svoje ID. ID se využívají u vnějšího směrování
* **Vnitřní směrování** = Směrování uvnitř sítě patřící jednomu ISP. Využívají se vnitřní směrovací protokoly (OSPF, RIP)
* **Vnější směrování** = Směrování mezi sítěmi různých ISP. Pro toto směrování se používají vnější směrovací protokoly (např. BGP)

## Vnější a vnitřní směrování aneb jak funguje Internet

* Internet = množina vzájemně propojených autonomních systémů
* **Hraniční směrovače** používají BGP protokol pro dynamickou konfiguraci směrovacích tabulek pro adresní rozsahy náležící jiným AS
* RIP, OSPF VŽDY pouze uvnitř jednoho AS

1. Přednáška – Transportní (4.) vrstva (TCP, UDP)

## Chybové situace

Cílové porty < 1024 se nazývají jako privilegované, ostatní jsou přiděleny jádrem OS dynamicky

Pro sledování kvality přenosu dat monitorujeme:

1. Ztrátovost (packet loss) – Kolik % paketů se po cestě ztratí a musí být znovu odesláno
2. Zpoždění (delay) – Průměrná doba na doručení paketů mezi zdrojem a cílem
3. Obousměrné zpoždění (RTT) – Typicky dvojnásobek delaye
4. Nestabilita, variabilní zpoždění (jitter) – Kolísání zpoždění, pakety stejného toku přicházejí v různých časových rozestupech po sobě, což má za následek jejich zpracování po různě dlouhých částech

Může nastat také situace, kdy se ztratí potvrzení nebo přijde potvrzení opožděně (po stanoveném timeoutu). V obou případech nastává znovu zaslání packetu.

**Problém dvou / Tří armád**

* 3-Way handshake ilustrace problému. Informace o útoku, zpět odpověď, potvrzení odpovědi (aby 2 oddělené armády najednou zaútočili na cíl mezi nimi)

## Algoritmy pro doručování dat

**Stop & Wait – Spolehlivé doručení dat**

* Jednoduché, málo efektivní. Jedná se o doručování dat přesně v pořadí, jak jdou za sebou

**Stop & Go – Pro řízení toku**

* Dokáže ovlivnit množství paketů, které přichází od vysílače, nedokáže řešit problém ztráty paketů
* Vysílač posílá pakety, dokud neobdrží zprávu STOP. Poté co STOP od přijímače obdrží, přestane vysílat dokud nepřijde zpráva GO.

**Klouzavé okénko – Pro spolehlivé doručení i řízení toku**

* Složitější, efektivnější. Řízení toku probíhá úpravou velikosti okna -> optimalizuje se na základě parametrů přenosů (delay, RTT, packet loss…)
* Vysílač může poslat [velikost okénka] paketů a když přijde potvrzení na první v okénku, tak může okénko posunout.

## Protokoly transportní vrstvy

**Spolehlivé – Například TCP**

* Data jsou vždy doručena od zdroje k cíli za cenu toho, že je protokol pomalejší
* Navazuje se a drží virtuální spojení = spojově orientované
* Řídí se tok, předchází se zahlcení, složitější implementace

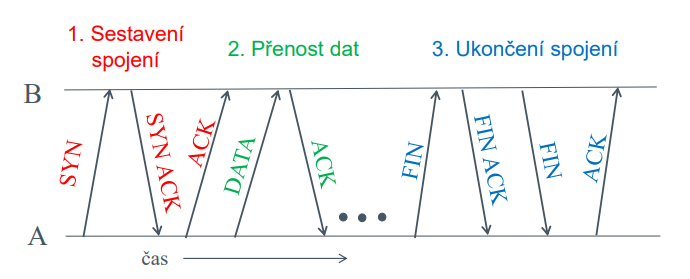
**Nespolehlivé – Například UDP**

* Data se mohou během přenosu ztratit, protokol je rychlejší, jednodušší implementace
* Paketově orientované

Transmission Control Protocol (TCP)

* TCP paket = položka data IP protokolu
* Umožňuje řízení toku (klouzavé okénko) i eliminaci zahlcení (CW = Congestion Window)
* Není šifrovaný, garantuje pořadí doručených paketů, detekuje duplicity, je duplexní
* Tok je řízen příznaky v hlavičce paketu (obsahuje sekvenční čísla, zdrojový/cílový port, kontrolní součty atd.)
  + Příznaky jsou: U = Urgent, P = Aplikační data, R = Reset, S = SYN, F = FIN

**Spojení TCP**



* 3-Way handshake je SYN - SYN ACK – ACK, pro přenos dat je pouze DATA a jejich ACK, nakonec je 4 cestné ukončení spojení
* Když je spojení udržováno, stav se nazývá **ESTABLISHED**

**Nastavení a úprava délky okna (DO)**

* Používá se mechanismus klouzavého okna, aplikuje se pouze ve směru vysílač -> přijímač
* Velikost okna říká, kolik TCP paketů se posílá najednou bez potvrzení
* Velikost DO navrhuje vždy přijímač -> v případě nastavení na 0 přestane vysílač posílat data
  + Přijímač také dynamicky mění DO, aby nedošlo k zahlcení

**Fast retransmit u TCP** = Situace, kdy vysílač si drží timeout, do kterého když nepřijde odpověď, tak danou část dat odešle znovu -> Zvýšení efektivity přenosu

**Zahlcení přijímače vs zahlcení linky**

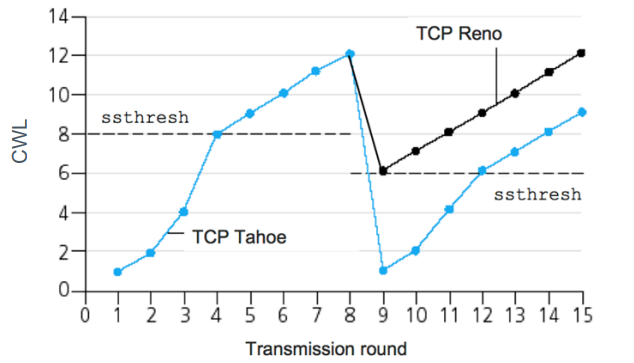
* Přijímač si nastavením velikosti okénka řídí množství odesílaných dat – STEJNÉ pro všechny varianty TCP
* Vysílač reguluje množství odesílaných dat. Použitý algoritmus se LIŠÍ dle konkrétní verze TCP protokolu

DO pro vysílání se označuje CWL (Congestion Window Length)

* CWL = Maximální počet paketů, které může vysílač odeslat najednou, aby nedošlo k přehlcení linky – reguluje vysílač
* Různé implementace nastavení CWL jsou podstatou různých verzí TCP protokolu (Reno, Tahoe, Vegas, New Reno, CUBIC…)

**Příklady nastavení CWL**

* Varianta TCP TAHOE
  + CWL na začátku = 1, při úspěšném doručení vždy zvýšení 2x. Jakmile počet odeslaných paketů dosáhne hodnoty ssthresh (na začátku 32KB), tak se CWL zvyšuje lineárně do doby, než dojde k prvnímu výpadku potvrzení
  + Poté se ssthresh sníží na polovinu a CWL jde opět na 1.
* Varianta TCP Reno
  + Funguje na stejné bázi, ale CWL nastaví na polovinu hodnoty před výpadkem a ssthresh se sníží o 25% nikoliv o 50%



User datagram protocol (UDP)

* V ideálních podmínkách lepší propustnost než TCP kvůli nižší režii. Typicky se využívá pro přenos videa (RTP protocol), kde občasná ztráta paketů tolik nevadí
* Není zabezpečnený = negarantuje doručení paketů a ani jejich pořadí

## Transportní vrstva a překlad adres (NAT)

* Díky portům je možné v případě NATu odlišit různá současné spojení jdoucí od více stanic ke stejnému cíli přes stejnou veřejnou IP adresu.
* Každé spojení je mapováno jako {Zdrojová IP, Zdrojový Port} – Zevnitř. Zvenku se přidělí stejná dvojice pro dané spojení znovu. Tyto dvě dvojice jsou spolu spjaté a při příjmu odpovědi NAT router opět přeloží vnější dvojici za vnitřní.

1. Přednáška – Transportní (4.) vrstva (DNS, DNSSEC …)

## DNS (Domain Name System) – Port 53 (Převážně využívá UDP)

* Slouží především k překladu IP adres na doménová jména a naopak, aby si lidé nemuseli pamatovat IP adresy. Také umí rozdělovat zátěž (Například odpovídá na jedno doménové jméno více IP adresami) nebo omezit někam přístup.
* **Jedná se o hierarchický systém**, který v původní verzi nebyl nijak zabezpečený. To dalo za vznik DNSSEC -> pomůže zabránit zfalšování odpovědi.
* OS obvykle obsahuje **resolver**, který pomáhá v komunikaci s Name Serverem (DNS serverem)
* Odpověď i dotaz mají definovaný společný formát paketu (může přenášet v jednom záznamu více dotazů i odpovědí)
* V Linuxu se seznam doménových jmen udržuje v souboru ***/etc/hosts***
  + Pokud se překlad resolveru ze souboru /etc/hosts nepovede, tak zkusí nadřazený DNS Server

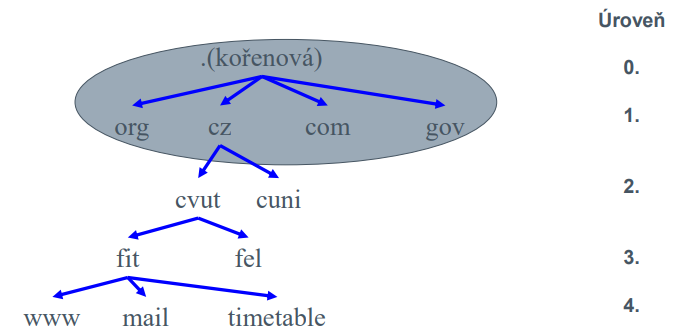
Doména = množina různých doménových záznamů sdružených pod společným doménovým jménem (např.: cz, cvut.cz…) -> Každá doména má definovaný její hlavní DNS server. Konkrétní doménový záznam se jmenuje DZZ (Doménový zdrojový záznam).

DNS provádí překlad doménových jmen:

1. **Přímo** = Doménovému jménu se přiřadí odpovídající IP (např.: [www.fit.cvut.cz](http://www.fit.cvut.cz) -> 147.32.232.212)
2. **Reverzně** = IP adrese se přiřadí odpovídající doménové jméno
   1. Formuluje se jako přímý dotaz v podobě DD.CC.BB.AA.in-addr.arpa

Kořenový DNS Server = DNS server 0. úrovně (13 skupin). Dotazy se doručují na servery pomocí anycastu (v IPv6 nativně, IPv4 pomocí BGP)

**Hierarchie**



**Top-level domains**

* Můžou být Generické (.com,.gov…), Country-Codes (.cz,.de), New Generic – libovolný řetězec (.london, .paris)
* 4. druhem je reverzní doména in-addr.arpa, která je určena pro převod IP na doménové jméno

**Pojem doména** = Množina doménových jmen pro servery, které jsou pod doménovým jménem dosažitelné

**Pojem zóna** = Konkrétní podmnožina množiny doménových jmen (Server, co se stará o překlady z dané zóny se nazývá autoritativní pro danou zónu)

**Vybrané typy DZZ**

* A – IPv4 adresa pro doménové jméno
* AAAA – IPv6 adresa pro doménové jméno
* MX – Server, který zajišťuje doručování emailů pro doménu

**Iterativní vs. Rekurzivní způsob dotazování v DNS**

* **Iterativní**
  + Ptá se po krocích, kdy celý proces má na starost nadřazený DNS server, který se postupně ptá všech instancí a vrátí pak odpověď
* **Rekurzivní**
  + Ptá se svého lokálního, který se dále ptá kořenového, ten se pak ptá serveru 1. úrovně a tak dále, dokud se nedostanou k autoratitivnímu DNS serveru a postupně se stejnou cestou vrací odpověď

**Dynamické DNS (DynDNS)**

* DNS server, který dokáže A záznamy pro doménová jména měnit dle pokynů stanic za běhu

**DNSSEC**

* Založena bezpečnost na asymetrické kryptografii -> možnost ověřovat pravost odpovědí

**Útok na DNS – Cache Poisoning**

* Situace, kdy útočník na dotaz jaká je IP adresa k doménovému jménu XXY odpoví dříve, než skutečný DNS server. Druhá (správná) odpověď je již zahozena. Obranou je DNSSEC

1. Přednáška – Bezpečnost v internetu (vrstva 3. a 4.)

**Demilitarizovaná zóna (DMZ)** = část počítačové sítě, který je zevnějšku dostupná (typicky tam bývají web servery, e-mailové servery…)

**Intranet** – Pouze vnitřní část sítě, která je dostupná jen oprávněným uživatelům (neveřejné weby jako raspberrypi.local)

## Firewall

Odděluje vnější a vnitřní provoz sítě

**Paketový**

* Pracují s informacemi SÍŤOVÉ vrstvy (3.)
* Vyhodnocují síťový provoz na základě zdrojové a cílové IP
* Deep Packet Inspection (DPI) – Hloubková analýza hlaviček paketu apod.
  + Nelze identifikovat na základě specifického formátu hlavičky každý protokol. Pokud to lze, tak na bázi takzvaných otisků.
* Například iptables -> mění, logují, zakazují, dovolují provoz
  + Nejlepší vše zakázat a specifické povolit
  + PREROUTING
    - 1) Forward – Pakety dále přeposílané
    - 2) Input – Takové, co jsou určené místním aplikacím
  + POSTROUTING
  + Tři základní tabulky pravidel
    - Filter = povolení či zakázání doručení paketu
    - Nat = překlad IP adres či portů v hlavičce paketu
    - Mangle = úpravy položek v hlavičce paketu pro další zpracování

**Stavové**

* Pracují s informacemi TRANSPORTNÍ vrstvy (4.)
* Složitější konfigurace vyžadující znalost protokolů 3. vrstvy. V pokročilých systémech jsou využívány v kombinaci s Intrusion Prevention Systems (IPS) / (IDS)
  + IDS = Systém pro detekci hrozeb, informuje administrátora, který provede opatření pro eliminaci hrozeb
  + IPS = Systém pro detekci a eliminaci hrozeb, kdy při hrozbě IPS generuje požadavky na Firewall na změnu pravidel

**Aplikační (Proxy)**

* Kontrola a přeposílání provozu mezi zdrojem a cílem výhradně na vrstvě aplikační
* Používají se často za účelem skrytí zdrojové IP adresy, jelikož proxy nastaví při odeslání jako zdroj sebe

## Útoky v počítačových sítích

**Honeypot** = počítač, který slouží jako návnada na útočníka, za účelem získání informací o něm (v demilitarizované zóně) a tváří se jako zdroj nějaké služby -> při napadení útočníkem se na základě dat o něm vytvoří pravidla do Firewallu.

1. **Skenování portů**
   * Zjišťování služeb, které běží na daném počítači (může se poté hodit k zahájení konkrétně cílenému útoku)
   * Port je otevřený, odpoví-li na útočníkovu snahu navázat TCP spojení
   * Horizontální skenování
     1. Útočník skenuje konkrétní port na počítačích s různými IP adresami
   * Vertikální skenování
     1. Útočník skenuje více portů na počítači s jednou IP adresou
   * Nejlepší obranou je firewall, IDS, IPS
2. **Prolamování hesel**
   * Útok probíhá prostřednictvím terminálových služeb (např.: SSH, TELNET)
   * Brute force attack
     1. Kombinace různých znaků systematicky zkoušející všechny možnosti
   * Dictionary attack
     1. Útočník používá seznamy nejčastěji používaných hesel
   * Nejlepší obranou je IPS, Firewall
3. **Denial of Service**
   * Realizován velkým množstvím dotazů ICMP nebo TCP-SYN, kdy je cílem danou např. webovou službu vyřadit jejím přetížením
   * DDOS
     1. Verze DIS vedená současně z různých infikovaných počítačů
   * RDOS (Reflected/Spoofed DOS)
     1. IP Adresa zdroje v hlavičce paketu je podvržena a lze těžko identifikovat útočníka
     2. Typicky útočník zašle mnoho ICMP dotazů, které se tváří, že jdou od oběti a odpovědi strojů ve stejné síti oběť zahltí odpověďmi
   * Nejlepší obranou je IPS
4. **Útoky využívající specifické bezpečnostní chyby v konkrétním systému**
   * Specifické situace, kdy je třeba poté dělat záplaty v systému, tento typ útoku se nazývá Červ (worm)
   * Např.: Buffer overfill – přepsání konkrétní části paměti, která vyvolá pak nějakou akci
5. **Odesílání spamu**
   * Nejčastěji se jedná o Phishing = útok vylákavající nějaké údaje z oběti (heslo atd.) prostřednictvím webové aplikace vypadající jako jiná, známá služba
6. **Malware**
   * Situace, kdy nějaká aplikace provádí akce bez vědomí uživatele
   * Trojský kůň = Normálně fungující aplikace, která občas spustí závadný kód
   * Ransomware = šifrování obsahu pevného disku za výkupné
   * Spyware = Snaží se uvnitř OS vyčíst údaje o uživateli
   * Adware = reklamy
   * Nejlepší obranou je Antivir a e-mailový blacklist
   * Infikované stanice využívané k DDOS
7. Přednáška – Virtuální privátní sítě (VPN)

Úkolem VPN je zpřístupnit část vnitřní sítě uživateli, který je od ní síťově / geograficky oddělen

* VPN také umí skrývat původního uživatele
* Vstupním místem VPN do místní sítě je koncentrátor
* Více druhů VPN lišící se dle toho, na jaké vrstvě pracují

**Druhy VPN**

1. Intranet – V rámci jedné organizace, např.: propojení dvou oddělení v rámci jedné firmy
2. Extranet – V rámci jedné organizace, ale prostřednictvím ISP, např.: dvě pobočky jedné firmy
3. Vzdálený přístup – Připojení zaměstnance do firmy, např.: Při homeoffice

Tunelování = posílání dat přes vytvořený tunel

* Vytvoření tunelu musí být povoleno na firewallech, mezi body
* Využívá se často, když nějaká síťová služba nemá nativní podporu šifrování. Šifrování poté zajišťuje VPN protokol -> do něho se data před vysláním **zapouzdřují**
  + Nosný protokol (NP) = Protokol nižší vrstvy, který zajišťuje doručení dat paketů VPN protokolu
  + VPN Protokol = protokol používaný k vytvoření tunelu
  + Data mohou být zapouzdřena i vícenásobně

**VPN na straně providera**

* L7 VPN = Open VPN, PPTP/SSTP
* L3 VPN = IP Over IP
* L2 VPN = VPLS

**VPN na straně zákazníka**

* IPsec GRE

**DSL (realizováno pro telefonní linky, kabelovou televizi)**

* IPSEC

**Vytáčené**

* PPP, L2TP

## PPP (Point-to-Point protokol)

Využíván na vzdálený přístup, pracuje na linkové (2.) vrstvě, podporuje kompresi i šifrování, propojuje právě dvě strany

## IPSec protokol

Využíván pro ověření identity účastníků a šifrování síťového provozu. IPv6 již od dob návrhu s ním počítá. Je nezbytný pro zabezpečení, tím, že žádné verze IP protokolů v návrhu nemají bezpečnostní mechanismy.

Propojuje vždy právě dvě strany a šifrují se vždy pouze DATA.

Jedná se o posloupnost tří důležitých fází:

1. Internet Key Exchange (IKE) = protokol, který dohaduje šifrovací klíče využíváne kroky 2) a 3)
2. Authentication Header (AH) = protokol, který slouží pro zajištění vzájemné autentizace
3. Encapsulating Security Payload (ESP) – Zajišťuje symetrické šifrování přenášených dat

Dva módy:

1. Transportní mód
   1. Původní IP hlavička je zachována a šifruje se pouze datová část paketu, nižší nároky na kapacitu přenosové linky
2. Tunelovací mód
   1. Původní paket je zabalen a chráněn v nově vytvořeném IP paketu. Obsahuje IP adresy switchů, tudíž skrývá IP adresy zdroje a cíle. Je náročnější na kapacitu přenosové linky

## IP over IP

Propojení sítí, které jsou schované za NATem. Pracuje na síťové (3.) vrstvě. Je jednoduchý a nenáročný. NENÍ ŠIFROVANÝ.

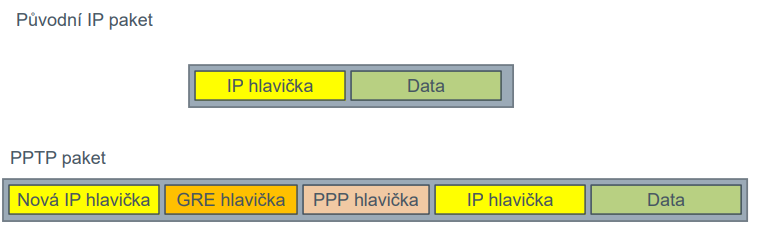
Původní pakety se zabalí do nových paketů, kde jsou obsaženy v položce „DATA“.

## Generic Routing Encapsulation (GRE)

Vyvinut CISCO, pracuje na transportní (4.) vrstvě. GRE zapouzdření je využíváno dalšími protokoly (PPTP). Hlavička neobsahuje porty a tudíž je blbě průchozí pro NAT.

## Point To Point Tunneling Protokol (PPTP)

Vyvinut Microsoftem. Využívá pro svou činnost protokoly GRE a PPP. Postaven na modelu klient-server, spojení je obousměrné.



## Layer 2 Tunneling protokol (L2TP)

Vyvinut členy PPTP fóra, Ciscem a IETF organizace.

Cílem je tunelovat rámce na linkové vrstvě. L2TP je NEŠIFROVANÝ. Lze ale použít IPsec zapouzdření.

## Secure Socket Tunneling Protokol (SSTP)

Vyvinut Microsoftem. Pracuje na aplikační (7.) vrstvě. Je přenášen v protokolu HTTPS a nahrazuje postupně PPTP. Pro zapouzdření dat také používá PPP. Tváří se jako http provoz a tudíž je dobře propustný pro Firewall.

**Fáze SSTP spojení:**

1. Sestavení TCP spojení
2. Spuštění šifrování prostřednictvím SSL
3. Přenos přes HTTPS

## Open VPN

Protokol pracuje na aplikační vrstvě (7.). VPN je v OS implementována prostřednictvím virtuálního síťového rozhraní.

Pakety OpenVPN jsou přenášeny jako data protokolu UDP nebo TCP. OpenVPN může fungovat jak pro 2 strany (point to point), tak více stran (point to multipoint). Funguje na mnoha OS.

# Anonymita v Internetu

Prohlížeč uživatele zanechává za sebou Otisk.

Otisk prohlížeče (OP) = Množina specifických atributů zasílaných v rámci http spojení -> využívá se pro cca identifikaci uživatele (lepší cílení reklam atd.).

## TOR a DarkNet

* TOR = Síť, která si klade za úkol znemožnit sledování uživatelů. Spoléhá na náhodné směrování, které je vždy jiné. Toto směrování se provádí přes TOR nody, které se vždy vyberou jiné pro cestu tam a pro cestu zpět. Tor poskytuje službu Onion service = anonymní obdoba DNS.
* Pro zprostředkování služby se kontaktuje centrální databáze s veřejným klíčem a vstupními TOR uzly. Při přístupu na danou službu, tak se využije takzvaného Rendezvous Pointu (místo setkání). Pak se vybere náhodný vstupní bod, zakóduje se zpráva (obsahující místo setkání, cookie speciální pro danou relaci) veřejným klíčem dané služby z databáze. Po obdržení druhá strana udělá podobný proces a komunikují přes Rendezvous point.

1. Přednáška – Aplikační vrstva (7.)

Úkolem aplikační vrstvy je interakce uživatele a vzdálených síťových služeb.

Typické služby, které jsou aplikačními protokoly zprostředkovávány jsou:

1. Vzdálená správa (SSH, Telnet, VNC) [Putty, Teamviewer]
2. Přenos souborů (FTP, SFTP, SCP) [Basic Linux commandy]
3. Elektronická pošta (SMTP, POP3, IMAP) [Outlook, Thunderbird, IceWarp]
4. Webové stránky (HTTP, HTTPS) [Mozilla, Opera, Chrome]
5. Zasílání zpráv
6. Peer-to-peer sharing (BitTorrent) [mTorrent]
7. Časová synchronizace (NTP) [ntpd command]
8. VoIP (SIP, RTP) [Skype]

## Vzdálená správa – protokoly SSH (Port 22) a TELNET (Port 23)

* Terminál slouží jako emulátor obrazovky a klávesnice, zprávy se přenáší v položce DATA TCP paketu.
* Protokoly jsou založeny na modelu klient-server
* Tyto služby jsou často zneužívané k nabourání do systému (např.: Dictionary attack, brute force)
  + Jako obrana proti tomuto slouží utilita fail2ban -> po X neúspěšných přihlášeních zablokuje přístup přes SSH dané IP adrese
* SSH je šifrované, lepší vs. TELNET je nešifrovaný, starší

### RDP (Remote Desktop Protocol) (Port 3389)

* Emuluje grafický přístup k počítači, především Windows, stále ve vývoji

## Přenos souborů – FTP (Port 20, 21)

* Přenáší se jako data v paketu protokolu TCP. Vytváří se 2 různá spojení zvlášť pro data a zvlášť pro příkazy (liší se v cílových portech)
* FTP není šifrované -> Šifrovaná verze je SFTP
* Obsahuje mnoho příkazů pro práci s filesystemem (načtení adresářů, editace souborů atd.)
* Vhodný pro přenos souborů
* **Aktivní FTP mód**
  + Klient -> Příkaz AUTH, otevření portu 21 (příkazová komunikace), následuje USER, PASS
  + Server -> Přijímá USER a PASS a ověří.
    - V případě úspěchu **KLIENT** navrhne port pro DATOVOU komunikaci.
    - Po přenosu dat končí spojení se serverem příkazem QUIT
* **Pasivní FTP mód**
  + Stejný proces jako výše, pouze si uživatel řekne, že chce PASV a poté port na DATA navrhuje **SERVER**
  + Důvod pasivního módu je ten, že pokud je klient za NATem, tak není zaručeno, že server může realizovat otevření spojení směrem ke klientovi

## Elektronická pošta – SMTP (Port 25), POP3 (Port 110), IMAP (Port 143)

* **Mail Transfer Agent (MTA)** = Program běžící na poštovním serveru, který odesílá či přijímá elektronickou poštu (např.: MS Exchange)
* **MTA Relay (forward)** = Něco jako MTA (takové předsazené MTA) vystavené internetu [demilitarizovaná zóna], slouží jako ochrana MTA před útoky
* **User Agent (UA)** = Program na uživatelské straně, který načítá poštu z mailboxu na poštovním serveru
  + **SMTP -**>Určený pro přeposílání e-mailů mezi poštovními servery, popřípadě e-maily mezi UA a MTA
  + **POP3, IMAP** -> Určené pro čtení pošty prostřednictvím UA a MTA
* **Doručování e-mailů mezi poštovními servery**
  1. MTA obdrží od UA prostřednictvím SMTP informace o emailu (zdroj, cíl)
     + Obsahuje také doménové jméno, ke kterému se přes DNS zjistí MX záznam
  2. MTA doručí prostřednictvím protokolu email poštovnímu serveru, který funguje jako MTA Relay na straně příjemce (Pomocí SMTP)
     + Z MTA Relay se e-mail uloží na MTA
  3. Pomocí POP3/IMAP a UA si může uživatel načíst mailbox
* **Srovnání POP3 a IMAP**
  1. IMAP
     + Projekce mailboxu -> Pouze se přistupuje k e-mailům na poštovním serveru, nic není lokálně uložené, tudíž změna na jedné stanici vyvolá změnu na všech
  2. POP3
     + E-maily se kopírují na každou uživatelskou stanici a neovlivnují přímo mailbox uvnitř Poštovního serveru. Změna lokální kopie neovlivní kopie na ostatních stanicích
* **SPAM**
  + Nevyžádaná pošta, jejíž součástí může být i například Malware
  + Filtrování spamu
    - Pomocí anti-spam filtrů, nastaveních na straně serveru apod.
    - Blacklisty apod. na straně klienta

## Webové stránky – HTTP (Port 80) / HTTPS (Port 443)

* Zprávy http protokolu se přenáší jako DATA protokolu transportní (4.) vrstvy
* Existují 3 verze:
  + **HTTP v1** = Textový protokol využívající TCP. Pro každou entitu (např.: obrázek) využívá samostatné TCP spojení
  + **HTTP v2** = Binární protokol, který dokáže vícero spojení přenášet přes jedno TCP spojení
  + **HTTP v3** = Binární protokol, pro přenos se využívá nepotvrzovaný protokol (QUIC). Stále ve vývoji -> cíl je min. latence a max. propustnost
* **Operace HTTP:** 
  + **GET – Načtení stránky ze serveru**
  + **HEAD – Načtení jen hlavičky ze serveru**
  + **POST, PUT – Posílání dat (heslo, jméno) na server**
  + **DELETE – Smazání informací ze serveru**
* **Odpovědi HTTP**
  + 200 – OK
  + 301 – Permanentně přesunuto
  + 404 – Chyba ve zprávě (Client-side chyba)
  + 500+ - Server side chyby
* Funguje na principu žádost -> odpověď
* Je bezestavový (nic si nepamatuje). Aby se do relací přenesly předchozí informace existují Cookies.
* Není šifrovaný -> šifrovaná verze je HTTPS (HTTP Secured)
* **HTTPS – Princip zabezpečení**
  + Server zasílá klientovi při požadavku svůj certifikát. Klient ho ověří a vygeneruje náhodný šifrovací klíč -> odešle serveru (zašifrovaný veřejným klíčem serveru)
    - Server pak s tímto Náhodným šifrovacím klíčem zašifruje obsah web. Stránky a odešle klientovi, ten už si to dešifruje

## 6. Peer-To-Peer – BitTorrent protokol

* Realizace sdílení dat mezi uživateli napřímo bez centrálního úložiště
* Funguje na bázi toho, že si odněkud uživatel stáhne .torrent soubor, který obsahuje informace o nějakých částech souboru, které si pak stahuje od různých uživatelů, kteří už ho mají lokálně stažený (každý kousek od někoho jiného)
  + -> V případě nelegálních kopií programů, filmů je toto jediná nelegální cesta. Stahování nelegální není, ovšem SDÍLENÍ již nelegální je. V situaci torrentu můžete být v určité situaci také serverem – zdrojem.

## 7. Časová synchronizace – NTP (Port 123)

* NTP protokol slouží k synchronizaci času, nastavuje systémové hodiny
* Zprávy NTP protokol přenáší jako DATA protokolu UDP
* Funguje na bázi **Christianova algoritmu**
  + Klient pošle požadavek, pamatuje si čas odeslání, vrátí se mu odpověď, vypočítá si čas, který si má nastavit
* **Hierarchie serverů**
  + Nejvýše – nejpřesnější -> Atomové hodiny
  + S nimi komunikují Stratum 1 servery
  + Max 3 stratum úrovně -> s každou se snižuje přesnost (v řádech mikrosekund) – kvůli zpoždění na linkách

## 8. Voice protokoly – VOIP

* VOIP prostřednictvím SIP protokolu – jeho úkolem je navázání spojení mezi stanicemi za účelem realizace hlasového hovoru
  + SIP – Obsahuje pouze signalizační příkazy
* Pro přenos hlasu se využívá RTP protokol -> Nesmí být mezi stranami NAT
* Není šifrováno

1. Přednáška – Základy bezdrátových sítí (1. a 2. vrstva)

**Elektromagnetická vlna** – šíření elektromagnetického pole prostorem v čase

* Šíří se ve dvou navzájem kolmých rovinách. Jedná se o elektrickou a magnetickou složku elmag. vlny
* Vlnová délka = vzdálenost 2 bodů, které kmitají ve fázi (Představit si to lze jako periodu sinusovky

## Antény

Anténa je prvek v bezdrátových sítích, který přijímá/vysílá elmag. vlny

Antény umožňují určit směr a sílu elmag. vln

**Parametry antén**

* Zisk antény = Úroveň zesílení intenzity elmag. vln po průchodu anténou
* Pracovní pásmo = Frekvenční rozsah, pro který byla anténa navržena
* Směrovost antény = Vyjadřuje se vyzařovacím diagramem

! Neexistuje universální anténa pro všechny frekvenční rozsahy !

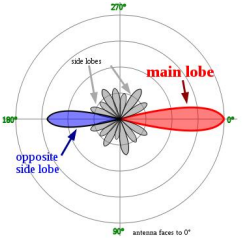
**Přenosový kanál** = pracovní pásmo antény, na kterém se přenášejí data (frekvence a šířka)

**Propustnost přenosového kanálu** = počet bitů za vteřinu, které je možné přenášet kanálem současně

**Polarizace antény**

* Rovina, ve které se šíří elektrická složka elmag. vlny
* Souvisí s polohou antény, může být vertikální i horizontální (pro obě roviny je zvlášť vyzařovací diagram) -> jejich složením vzniká kruhová polarizace (ta je více odolná vůči rušení).
* Anténa má svůj hlavní směr vyzařování, na opačnou anténa svým vysíláním ruší. Zároveň každá anténa je tak trochu všesměrová (side lobes)

**Vyzařovací diagram**



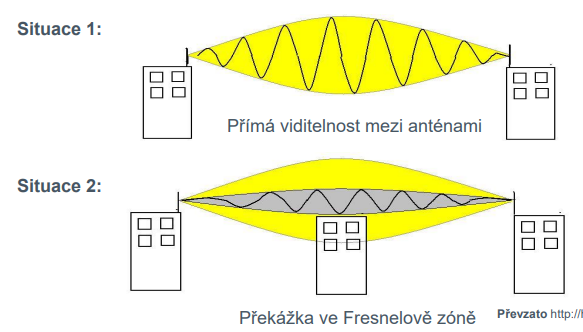
* Druhy antén dle konstrukce – Zig-Zag, Yagi, Paraboly, Spirálové
* Anténa má svůj hlavní směr vyzařování, na opačnou anténa svým vysíláním ruší. Zároveň každá anténa je tak trochu všesměrová (side lobes)

**Druhy antén dle tvaru vyzařovacího diagramu**

1. Všesměrové (360°)
   * 10ky, 100ky metrů (pokryjí celý okruh, připojení klientských stanic)
2. Sektorové (30-180°)
   * 1ky kilometrů (Klientské stanice na místech s vyšším rušením)
3. Směrové (10-30°)
   * 1ky kilometrů, Používají se na propojování páteřních oblastí.
4. Úzcesměrové (< 5°)
   * 10ky kilometrů, Používají se k propojování hlavních částí na dlouhé vzdálenosti

**Anténní pole** = Soustava antén fungující jako jedna anténa. Celkový vyzařovací diagram je součet vyzař. Diagramů konkrétních antén.

**Fresnelova zóna** = Signál se nešíří pouze po přímce, ale ve tvaru viz. Níže. Tudíž pouze přímá viditelnost antén nestačí a je nutné počítat s touto zónou, aby nedocházelo k rušení.



## Multiplex v bezdrátových sítích

Situace, kdy více stanic využívá stejné přenosové médium.

**Nejvíce používané multiplexy**

1. Frekvenční multiplex (Přednáška 2)
2. Časový multiplex (Přednáška 2)
3. Prostorový multiplex (Přednáška 2)
4. **Ortogonálně frekvenčně dělený multiplex** 
   * Více kanálů na jedné lince, dělených dle frekvenčních délek. Více kanálů = více rušení = větší propustnost
5. **Skákavé frekvenční spektrum** 
   * Velké množství podkanálů, mezi kterými se stále přelaďuje
   * Vysoká odolnost proti rušení, nízká propustnost

## Modulace signálu

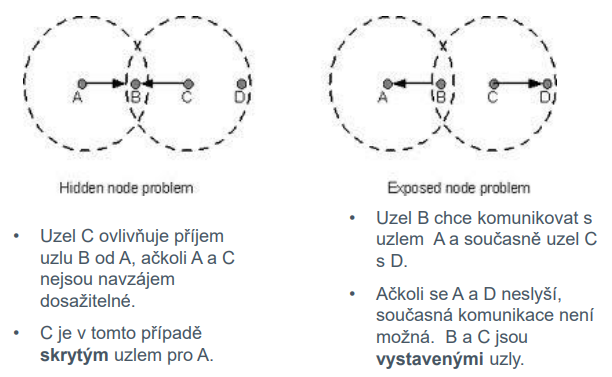
Modulace = přizpůsobení signálu, který nese datovou informaci do frekvenčního pásma antén, pro které jsou konstruovány

Druhy modulací, které dělá modulátor při odeslání:

1. **Jednoduché**
   1. Amplitudová, frekvenční, fázová
2. **Složené**
   1. Kombinují více jednoduchých modulací do sebe a umožňují přenášet více bitů v rámci jedné elmag. vlny najednou (Quadratic Amplitude Modulation)

Demodulátor pak informace „dešifruje“ při příjmu.

**Problém skrytého/vystaveného uzlu (Hidden/Exposed node problem)**



## IEEE 802.11b/g/n/ac/ax (Wi-Fi)

Jedná se o takový bezdrátový Ethernet.

**Topologie se dělí na:**

1. Infrastrukturní – centrální jednotka (AP) řídí komunikaci mezi stanicemi.
2. Adhoc – Stanice spolu komunikují napřímo bez AP

Rámců v Wi-Fi existuje více druhů (na rozdíl od Ethernetu)

1. **Kontrolní**
   * Slouží pro přihlášení stanice do sítě (přihlášení k AP)
   * Zasílají parametry, které stanice musí podporovat pro připojení k síti (Beacon)
2. **Řídící**
   * Řídí přístup stanic k médiu – určují, která stanice může vysílat (RTS/CTS, ACK)
3. **Datové**
   * Přenášejí data mezi AP a stanicí, popř. mezi stanicemi

**RTS/CTS mechanismus**

Řeší problém skrytého uzlu, který CSMA/CA řešit neumí. Občas se s CSMA/CA může kombinovat.

RTS = Request to send, CTS = Clear to Send

* (CTS se pak odesílá všem stanicím, nejen té, co posílala RTS, ovšem na těch se zahodí)

**802.11b verze – POUZE 2,4 GHz**

-> Překrývají se kanály (V ČR použitelné jen 3 kombinace z celk. počtu 13 kanálů)

**802.11n verze – 2,4 GHz nebo 5.1-5.9 GHz**

-> Je podporováno více antén (MIMO systémy)

* Multiple Input Multiple Output. Pro příjem a vysílání se používá více antén současně. Podporuje současně využití až 4 antén
* **Využívá se pro:**
  + **Prostorový multiplex** (Data se rozdělí před vysíláním a jsou odesílána po částech různými anténami -> vyšší propustnost přenosového kanálu)
  + **Diverzita antén** (Situace, kdy MIMO není klientem připojujícím se k AP podporováno, a tak se více antén využije k tomu, aby se našla ta s nejlepším signálem)

-> Je možné využívat beamforcing

* Vysílání elmag. vln v úzkém směru, který lze dynamicky měnit (Díky tomu dochází k menšímu rušení ostatních stanic kvůli využití užšího vyzařovacího směru)
* Využívá se pro to *úzce-směrové nastavitelné antény* nebo *anténního pole*

-> Přenosové kanály lze spojovat

* Více přenosových kanálů se využije současně k vytvoření jediného širokého přenosového kanálu

**802.11ac/ax verze – Současné využití 2,4 a 5,1 GHz pásem**

* Pro MIMO lze využít až 8 antén, nové zabezpečení (WPA3), má dokonce MIMO-Multi User, což je situace, kdy pro každého uživatele je vyhrazena část z celkového počtu antén.
* Přenáší se ještě více bitů najednou díky lepším modulacím

1. Přednáška – Internet poslední míle a optické sítě (1. vrstva)

**Poslední míle** = Oblast mezi koncovým uživatelem a poskytovatelem internetu

**Vid (mode)** = cesta, kterou se šíří paprsek v optickém vlákně

**Disperze** = při přenosu dat v optických vláknech vznikají různé poruchy

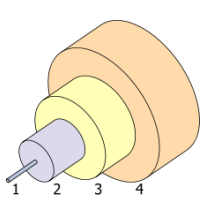
## Metalika vs Optika

Metalika = koaxiální a elektrické kabely

* Umožňují přenos dat a napájení zařízení současně, většinou měďěné
* Jsou levnější, umožňují jednodušší práci s nimi
* Dle ochrany vodičů můžou být nestíněné (Dostačující na domácí použití) a stíněné kabely
* **Asymetrické – Koaxiální kabely**
  + Především pro satelitní a televizní rozvody, obsahují vnitřní (datový) vodič, dielektrikum a vnější vodič + plášť (Na metaliku dobré odrušení)
  + Signál postupně slábne = útlum (10 - 100kymetrů)
  + **PLC (Power Line Communication)** 
    - Signál je šířen stejnými vodiči jako je vedeno elektrické napětí
    - Zneužívá již existující el. Síť (např.: v domě)
    - Velké rušení (v el. Síti), používá se TDMA (omezený počet účastníků), lze využít maximálně na domácí použití
* **Symetrické – Twisted-pair**
  + Kroucení = ochrana proti rušení, může mít v sobě 3,4,22,24,26 párů vodičů
  + Unshielded twisted pair (páry vodičů jsou jen ve vnějším plášti)
  + Shielded twisted pair (páry vodičů jsou ještě každá zvlášť obalen v metalické fólii)
  + Konektory:
    - RJ11 – Určený pro 3 páry vodičů, telefony, DSL (max 100MB/s)
    - RJ45 – Určený pro 4 páry vodičů, běžné počítač. Sítě (max 10Gb/s)
    - ARJ45 – Určený pro 6 párů vodičů, kompatibilní s RJ45 (max 4Gb/s)
  + Twisted-pairs se dělí ještě na různé **(CAT) kategorie**. Ty dle svých specifikací říkají, zda-li je kabel stíněný, jakou má maximální propustnost, šířku pásma apod.
  + **PoE (Power over Ethernet)**
    - Napájení zařízení přes ethernetový kabel, využívá se na místech, kde je problém s externím napájením (IP kamery)
    - Mohou (aktivní) upravovat přenášené napětí, ale nemusí (pasivní)
    - 4 vodiče se využijí pro data, 4 pro napájení

Optika = Optická / skleněná vlákna

* Nejrychlejší přenosové médium (až 100ky Gb/s), ale neumožňuje napájení, mají nižší útlum
* Fyzikální principy: Index lomu, Snellův zákon (specifikuje, jak se chová světlo při přechodu mezi hustšími/řidšími prostředními)
* **Struktura optického vlákna**
  + 1) Jádro (core)
  + 2) Plášť (cladding)
  + 3) Ochranná vrstva (isolation)
  + 4) Obal (jacket)



1. **Jednovidové optické vlákno**

* Poměr jádra a pláště je 8/125
* Do jádra vstupuje od diody jediný paprsek, který se přímočaře šíří prostředím
* Jsou dražší, vydrží větší vzdálenosti

1. **Vícevidová vlákna se skokovým** **indexem lomu**

* Poměr jádra a pláště bývá 50/125
* Paprsky mohou vstupovat z diody do vlákna pod více úhly. Pomocí **totální reflexe** dochází k vytváření cesty paprsku
* Pouze krátké vzdálenosti (větší útlum), jsou levnější
* **Vlnová disperze**
  + Situace, kdy ačkoliv všechny paprsky vstupují na stejném místě, tak na výstupu se neprotnou v jednom bodě. Výstupní puls je oproti vstupnímu širší

1. **Vícevidová vlákna s gradientním indexem lomu**

* Jádro je tvořeno z tisíců různých vrstev, které postupně mění index lomu, aby se signál šířil ve formě sinusovky.
* Důvodem vývoje byla eliminace disperze
* **Vlnová disperze**
  + Naprosto minimální díky pozvolným změnám indexu lomu

**Chromatická disperze**

Důvodem této disperze je, že různé barvy světla se šíří různou rychlostí – na vstupu jsou vyslány současně, na výstupu se objeví postupně.

Pro její eliminaci existují kompenzátory chromatické disperze.

**Vlnový multiplex (WDM)**

Wavelength Divison Multiplex (speciální druh frekvenčního multiplexu)

Dokáže v jednom vlákně dle barev signálů (vlnových délek) odesílat více paprsků. Funguje pro příjem i vysílání na jednom vlákně.

Má dvě varianty (Coarse, Dense – lepší, dražší)

**Optické trasy pro internet poslední míle (FTTX)**

Pro zakončení trasy se používají SPF konektory. Jejich druh závisí na tom, zda-li optické připojení vede až do domu zákazníka, nějaké budovy nebo jen rozvaděči.

Druhy:

1. **Aktivní optická síť (AON = Active Optical Network)**
   * Používají se pouze aktivní zařízení. Výstavba je dražší.
   * Aktivní optická zařízení (přepínače – switche) vybírají pro další šíření dat jeden konkrétní výstupní port dle cílové MAC
2. **Pasivní optická síť (PON = Passive Optical Network)**
   * Používají se levnější pasivní prvky (huby) – odesílají data na všechny výstupní porty. Stanice si tedy musí sama data filtrovat.
   * Logicky jsou tedy méně propustné kvůli duplicitě dat

# Random proseminář poznámky below

1. Proseminář

**Segment lokální** **sítě** = část sítě, která je ohraničena routery. Komunikaci v segmentu sítě zajišťují přepínače

* Router je vlastně z půlky součástí segmentu -> má 2 síťová rozhraní (jedno je právě součástí segmentu sítě)
* Broadcast se vysílá právě v segmentu sítě
* Zjednodušuje routování napříč sítěmi -> stačí znát všechny výstupy z naší sítě (default GW)

LAN = Local area network – obsahuje jednu nebo více segmentů sítě obsahující koncová zařízení. Typicky:

* LAN = Naše síť doma
* WAN = Poskytovatel internetu (Jako na routeru porty)

Adresní prostor = list identifikátorů, které luze přidělit v počítačové síti

* Požadavky na jeho dělení:
  + Každý síťový segment má unikátní identifikátor
  + Každá síťový segment má unikátní identifikátor
  + Z identifikátoru síťového rozhraní je jednoznačné, do kterého síťového segmentu patří

Switch nemá adresu (Pokud není L3 switch)

IP ADRESA and Maska = nejnižší adresa sítě (and = bitová operace)

Nezapomínat na to, že i router potřebuje IP Adresu – takže počítat ho jako „koncový zařízení“ při počítání počtu adres.

Při hledání adres sítě, tak můžeme používat pouze násobky počtů koncových stanic (masky) -> viz fotografie

* VLAN (Virtual Local Area Network) – Jsou určené k tomu, aby uměly oddělovat toky v sítích na 2. vrstvě
  + Lze dělit dle portů, adres, protokolů nebo připojení značky/tagu (dnes nejčastější)
* Trunk port = port, který odbavuje provoz více různých VLAN

1. Proseminář

* Switch – zařizuje komunikaci jen v lokálním segmentu
  + Jakmile přijde paket, na který v tabulce není záznam na přeposlání, tak se to pošle na všechny výstupní porty a očekává, že pouze pro koho to bylo tak to přijme a odpoví a zbytek zařízení rámec zahodí (síťové karty zahazují)
  + Updatuje svou CAM tabulku dle příchozích paketů -> source addresses
  + Pokud mám dvě stejné MAC adresy v síti, tak se vždy přepisuje tabulka a furt se přepíná asociace mezi porty – probíhají pak ztráty v komunikaci, když se třeba v mezičase příjmu odpovědi od někoho změní nastavení
  + Jeden port vyskytující se u více MAC adres je normální, protože třeba tam může být další switch
  + Pokud by byla tabulka plná, tak se prostě přemaže (To se ale spíš nestane)
  + IPV4 – ARP – PC se musí zprávami zabývat X IPV6, řeší se rovnou na úrovni síťovek
  + ARP Protokol – Protokol síťové vrstvy, ale nepoužívá IP hlavičku, takže nelze s ním komunikovat mimo naší síť
    - Arpping – pingování na MACovkách -> arp dotazy a odpovědi na ně, výhodou je, že na to musí každé zařízení odpovídat
    - Směrovače oddělují broadcastové domény
* DHCP – stavová konfigurace
  + Obvykle obsluhuje router nebo je nějaký separátní server, vytváří si tabulku stejně jako když někdo nastavuje adresy manuálně
  + Discover – Chci adresu, kde je DHCP server? – broadcastem
    - Obsahuje i IP hlavičku, ale client má adresu 0.0.0.0 (logicky, protože ještě nemá žádnou přidělenou) a server 255.255.255.255
  + Offer – SERVER
    - Vezme adresu z poolu a ověří po síti, že tam nikdo s takovou adresou není
    - Pošle clientovi všechno jako IP, GW atd.
  + Request
    - Zopakuje informace, které dostal v nabídce
  + ACK – SERVER
    - Potvrzení, že přidělil klientovi IP Adresu
  + Lease time = za jak dlouhou dobu, se má klient zeptat znovu na svou IP Adresu
* Bezestavová konfigurace – link local
  + Je to takový „backup“, když se pokazí nějaké spojení.
  + Každý client si vygeneruje náhodnou adresu s prefixem 169.254.x.x/16 a odešle arpem všem, že má tuhle a pokud nikdo neodpoví, tak si ji přidělí
    - V případě kolize vygeneruje znovu
  + Zpřístupní alespoň lokální služby a protokoly, když už ne internet

1. Proseminář

* Algoritmus směrovací tabulky:
  + Destination IP addr je přijata na routeru
  + IP and GENMASK (Genmask od těch, co mají nejdelší prefix)
    - V případě matche se stále bere v ohledu Metrika
  + Match = situace, kdy po tom andu je výsledkem destination IP
* Pokud destination = 0.0.0.0 a genmask 0.0.0.0, tak to je default gateway -> interface, kam pošle router cokoliv, na co nešlo aplikovat žádný záznam
* Při konfiguraci rozhraní se vždy rovnou přidá do směrovací tabulky ty přímo připojené sítě (např dostanu/nastavím 192.168.0.1 na eth0, tak v tabulce je automaticky 192.168.0.0 na eth0 – Brána 0.0.0.0 a maska viz. síť)
* Koncové stanice mají ve směrovací tabulce pouze
  + 1) Default GW
  + 2) Přímo připojenou síť (ta v které existuje)
* DNS adresa napřímo nastavena v OS