Úvod do počítačových sítí - shrnutí základních pojmů

Karel Velička

30-12-2022

Vyučující: *RNDr. Libor Forst* a *Mgr. Klára Pešková, Ph.D.*

Obsah

1 Lekce

1.1 Komunikace - odolnost a bezpečnost

1.1.1 Okruhy X Pakety

Okruhy síť pro propojení *našeho* a *koncového* zařízení musí najít posloupnost uzlů.
Náchylné k vypadkům - odpojením uzlu zanikne spojení. Přesnos dat je rychlý ale nespolehlivý.

Packet rozdělení dat na menší blok. Každý paket si najde vlastní cestu k cílovému uzlu.
Při výpadku uzlu si paket najde alternativní cestu. Přenos je pomalý a spolehlivý.

1.1.2 DoS a DDoS

Denial of Service útok snažící se zahlcením znepřístupnit komunikaci s ostatními uživateli

Distributed Denial of Service využití cizích serverů k provádění *DoS*

1.2 Síť a její rozšiřitelnost

Animované video: YouTube.

1.2.1 LAN

LAN (Local Area Network): místní privítní síť, je rozdělena na 3 vrstvy

1. **Core:** hlavní část sítě, napřímo spojená s *IPS (Internet Service Provider)*
   (a) **Router:** uzel propojující různé sítě
   (b) **Switch:** přepínač - propojuje uzly navzájem.
2. **Distrubuční:** distrubuuje síť do všech částí budovy.
3. **Přístupová (access):** umožňuje připojení k síti koncovym zařízením.

1.2.2 WAN

WAN (Wide Area Network): globální veřejná síť *(LAN je součástí WAN)*. Rozdělena do tří vrstev.

1. **Tier 1:** klíčoví hráči internetu - společnosti s přímym přístupem. Propojují kontinenty.
2. **Tier 2:** regionální/ národní operátoři
3. **Tier 3:** ISP propojující koncové zákazníky, kteří chtějí LAN připojit k internetu

1.2.3 Přenosové parametry

Latence: zpoždění komunikace - doba od odeslání do doručení
Jitter: rozptyl zpoždění - vyjadřuje pravidelnost přijatych dat
Ztrárovost: kolik packetů nebylo doručeno
Bandwidth (Šířka pásma): *"rychlost"* - kolik dat lze přenášet

1.2.4 Kvalita služeb

QoS (Quality of Service): určuje prioritu.
Best effort: Síťovy uzel vybírá z fronty podle priorit.

1.3 Veřejné a privátní sítě

1.3.1 VPN

Virtual Private Network spojuje dvě privátní sítě přes veřejnou síť.
Dvě LAN jsou propojeny *VPN tunelem*, ktery vede přes WAN. Machanismus je v rámci obou LAN transparentní a tváří se jako jedna LAN.
Provoz z LAN_1 se zašifruje a vstoupí do *tunelu*, odešle se přes WAN, při výstupu z *tunelu* se dešifruje a nakonec je doručen do LAN_2.

1.4 Historie internetu

1.4.1 ARPA(NET)

Advanced Research Project Agency agentura specializující se na přepínání paketů. Vytvořili první *point-to-point* spojení přes telefonní linky - *ARPANET.*

1.4.2 RFC

Request for Comments prostředek standardizace intenetu

1.5 Souhrn 1

1.5.1 Jaké jsou výhody a nevýhody přepojování paketů?

Vyhodou je spolehlivost přenosu, nevýhodou je rychlost *(resp. pomalost).*

1.5.2 Jak se na síťových protokolech projevilo to, že vznik sítí iniciovala armáda z důvodů zvýšení bezpečnosti komunikace?

Bylo bráno v potaz jen fyzické bezpečnostní riziko (překopnutí kabelu), ne ochrana dat a soukromí.

1.5.3 Co je smyslem požadavku na škálovatelnost sítě?

Aby přidání nového počítače bylo snadné a nemusely se kvůli němu provádět zásadní změny.

1.5.4 Jak se liší nároky elektronické pošty a telefonování po IP síti na přenosové parametry sítě?

Pošta může mít vyšší latenci, ale požadujeme po ní nulovou ztrátovost. Při telefonování si můžeme dovolit pár dat poztrácet, protože chceme mít latenci co nejnižší.

1.5.5 Jaká je definice LAN?

Přesná definice neexistuje.
Můžeme ji definovat například jako místní privátní síť, která je napojena na IPS a která umožňuje připojení k síti koncovym uživatelům.

1.5.6 Co je podstatou VPN?

Spojuje dvě privátní sítě přes jednu síť veřejnou.

2 Lekce

2.1 Síťovy (referenční) model

Síťovy model počet vrstev, jejich struktur; rozdělení práce mezi vrstvy

2.1.1 OSI model

Open System Interconnection síťová architektura sestavená z Basic Reference Modelu a sady protokolů. Je vhodny pro dokumentaci, v reálu nepoužitelné - navrženo shora, neprakticky. Rozdělen do sedmi vrstev:

1. **vrstva - fyzická:** zajišťuje fyzicky přenos bitů mezi uzly
2. **vrstva - linková:** zajišťuje přenos dat mezi uzly, určuje zdroj a cíl přenosu, hledá a opravuje chyby
3. **vrstva - síťová:** zajišťuje přenos datovych bloků s **proměnlivou** délkou pakety mezi dvěma uzly v různych sítích *(jinak předá pižadavek linkové vrstvě).*
4. **vrstva - transportní:** zajišťuje přenos a příjem datovych bloků s **neomezenou** délkou mezi dvěma aplikacemi. Možná segmentace velkych bloků. End-to-End přenos datovych bloků.
5. **vrstva - relační:** řídí dialog mezi dvěma aplikacemi
6. **vrstva - prezentační:** datová konverze pro aplikace
7. **vrstva - aplikační:** zajišťuje interakci mezi uživatelem a protokolem, ktery implementuje komunikaci k vykonání požadavku

2.2 Síťová architektura

Síťová architektura síťový model s konkrétními technologiemi, službami a protokoly

2.2.1 TCP/IP

1. **vrstva - síťové rozhraní:** Ethernet, WiFi, ...
2. **vrstva - síťová:** odpovídá 3. vrstvě v OSI. Používán protokol IP (IPv4, IPv6) *(Internet Protocol)*
3. **vrstva - transportní:** odpovídá 4. vrstvě v OSI. Používány protokoly TCP a UDP
4. **vrstva - aplikační:** odpovádá 5,6,7 vrstvě v OSI. Je snazší definovat pravidla dialogu a vyznam dat bez mezivrstev.

2.2.2 TCP

Transaction Control Protocol spojovaná služba $\implies$ spolehlivé doručení dat, je jednodušší aplikace, nemůže řídit komunikaci, složitá implmentace.
Odešle datovy blok, rozsegmentuje data na menší bloky, odešle v jednotlivych paketech a mezitím potvrzuje úspěšné doručení *(pokud se nedoručí, odešle se znovu).*

2.2.3 UDP

User Datagram Protocol nespojovaná služnba $\implies$ nespolehlivá - není zaručeno pořadí ani doručení paketů, kontrolu provádí aplikace, může řídit komunikaci.
Odešle pakety a odpovědnost přenechá na aplikaci.

2.3 Aplikační modely a adresování počítačů

2.3.1 Klient-Server X Peer-To-Peer

Klient-Server Klient musí znát adresu serveru a pokusí se ho kontaktovat, poté se zahájí přenos dat.

Peer-To-Peer Uživatel nemusí znát adresu serveru, vyměna dat je obousměrná, nejsou zde vyhraněné role $\implies$ každý je server i klient

2.3.2 Adresování počítačů

HW - Linková vrstva fyzická MAC adresa, nastavitelná *dříve dána vyrobcem*

SW - Síťová vrstva IP adresa, určuje jednoznačně počítač v síti, předělována podle topologie sítě

Lidé - aplikační vrstva doménová adresa, hierarchie je zprava doleva - poslední je **TLD = Top Level Domain** *(ostatní SLD - Second Level Domain, spravováno majitelem)*, k převodu mezi doménovymi jmeny a IP se používá DNS, k převodu mezi síťovými a MAC se používá protokol ARP

Socket jeden konec komunikačního kanálu mezi klientem a serverem - IP adresa + port *(16b int)*

NAT (Network Address Translation) LAN používá privátní adresy a ven se představuje veřejnymi adresami, video

2.4 Adresování služeb

2.4.1 URI

Uniform Resource Identifier definuje odkaz, buď umístění zdroje (URL - Locator) nebo mázev služby (URN - Name - nebylo úspěšně implementováno)
URI = *schéma*://*autorita*/*cesta*/(?*dotaz*, #*fragment*) URI = http://sunsite.mff.cuni.cz/Net/RFC#ahoj

2.5 Datovy tok v TCP/IP

Video.

2.5.1 Multiplexing a zapouzdření

Multiplexing několik komunikačních kanálů v určité vrstvě používá stejny komunikační kanál v podřízené vrstvě

Zapouzdření vrstva $n-1$ převezme řídící informace, zpracuje je a připraví PDU (Protocol Data Unit), která se skládá z těla obsahující PDU_n a záhlaví s řídícími informacemi.

3 Lekce

Autentizace proces používán uživatelem k prokázání totožnosti

Autorizace proces přiřazení serverem oprávnění nějakému ověřenému subjektu

OTP (One Time Password) umožňuje nereplikovatelnou plain-text autentikaci uživatele
používalo se *challenge-response* - server po přihlášení pošle náhodny řetezec, ktery potom uživatel zadá spolu s heslem do kalkulačky a vysledek napíše serveru jako odpověď.
Nyní se používají malá HW zařízení (tokeny), která jsou synchronizovaná se serverem a generují kód pro identifikaci - platnost jen několik sekund, na jedno použití

3.1 Kryptografické algoritmy

3.1.1 Symetrické šifrování

Pro šifrování a dešifrování se používá stejny klíč.
Vyhodou je rychlost, nevyhodou je, že obě strany si potřebují předat klíč
Například *Blowfish, AES*

3.1.2 Asymetrické šifrování

Pro šifrování a dešifrování se používá pár navzájem neofvoditelnych klíčů.
Třeba mít dva klíče - tajny a veřejny.
Vyhodou je, že obě strany nemají společny klíč, nevýhodou je rychlost + autenticita veřejného klíče (ověřuje třetí strana - *web of trust* nebo *CA - Certification Authorities*)
Například *RSA, DSA.*

3.1.3 Hashovací funkce

Vytvoření z jakéhokoli datového vstupu pevny kód krátké délky.
Malá změna vstupu zajistí naprosto odlišny vysledek. Funkce je jednocestná.

3.2 Šifrování dat

Efektivní šifrování kombinuje symetrické i asymetrické šifrování.
Vygenerujeme klíč → symetricky zašifrujeme text → asymetricky zašifrujeme klíč veřejnym klíčem příjemce → předáme příjemci → asymetricky dešifruje tajnym klíčem klíč → klíčem symetricky dešifruje text.

3.2.1 Elektronicky podpis

Kombinace asymetrického šifrování a hashovací funkce.
Odesílatel vezme text → vypočítá jeho hash → vezme svůj soukromy klič a zašifruje hash → připojeno k původnímu textu → příjemce dešifruje hash veřejnym klíčem odesílatele → pokud jsou obě hodnoty hashe stejné, je podpis pravy.
Odhalí změny v souboru a změnu odesílatele.

3.3 Algoritmy

3.3.1 Diffie-Hallmanův algoritmus

Umožňuje dvěma partnerům dohodnout se na společném tajemství pomocí komunikace přes otevřeny nezašifrovany kanál.

1. Alice vygeneruje tajné číslo a a veřejná prvočísla p, q.
2. Spočítá $A = p^a \mod q$ a pošle p, q Bobovi.
3. Bob vygeneruje tajné číslo b, spočítá $B = p^b \mod q$ a pošle B Alici.
4. Alice spočítá $s = B^a \mod q$ a Bob taktéž $s = A^b \mod q$

Kanálem se posílají pouze vysledky diskrétních logaritmů. Funkce jsou jednocestné.
$A^b = (p^a)^b = p^{ab} = p^{ba} = (p^b)^a = B^a$

3.3.2 SSL a TSL

Secure Socket Layer a Transport Layer Security Mezivrstva mezi transportní a aplikačnívrstvou umožňující autentikaci a šifrování *(například HTTPS - HTTP přes SSL)*

1. Klient pošle požadavek na SSL + parametry
2. Server pošle odpověď + parametry + certifikát serveru
3. Klient ověří server, vygeneruje šifrovací klíč, zašifruje ho veřejnym klíčem serveru a pošle mu ho
4. Server rozšifruje šifrovací klíč a vytvoří z něj oba hlavní šifrovací klíč
5. Klient i server si potvrdí komunikaci šifrovanou tímto klíčem

3.4 Aplikační vrstva TCP/IP

Určuje pravidla komunikace mezi serverem a klientem. Protokol definuje: *formát zpráv (txt, bin...), průběh dialogu (kdo inicioval spojení...), typ zpráv (požadavky a odpovědi), vyznam zpráv (musí ∃! interpretace), interakce s transportní vrstvou (UDP nebo TCP, ...)*

3.5 DNS

Domain Name System služba *(klient-server)* pro překlad doménovych jmen na IP adresy a naopak.
Běžné dotazy se vyřizují přes UDP, větší datové vyměny přes TCP *(pokud data překročí limit a zopakuje se zpráva za pomoci TC (truncated)).*
Klient: posílá dotazy na servery, které má ve své komunikaci, postupně zvyšuje timeout na příchod odpovědi, dokud ji nedostane. Pokud odpověď neobsahuje potřebné informace, obsahuje odkaz na další servery.
Binární protokol: zpráva obsahuje hlavičku, počet záznamů *(RR - Resource Record)* . Každy záznam obsahuje jméno, TTL - Time To Live*(doba platnosti v sekundách)*, typ záznamu a odpovídající data.
Například: *ns.cuni.cz 3600 in A 195.113.19.78*

3.5.1 DNS záznamy

1. **SOA** (Start Of Autority): obecné informace o doméně
2. **NS** (NameServer): jméno nameserveru domény
3. **A**: IPv4 adresa počítače
4. **AAAA**: IPv6 adresa počítače
5. **PTR**: reverzní jméno, slouží pro převod adresy na jméno. Důvodem je hierarchie jmen (zprava doleva) a adres (zleva doprava).
6. **CNAME** (Canonical NAME): záznam pro tvorbu aliasů. Na levé straně jméno aliasu, na pravé kanonické jméno PC.
7. **MX** (Mail eXchanger): definuje ktery server přijímá pro danou doménu poštu.

3.5.2 Servery DNS

Autoritativní nameserver primární + sekundární server; každá doména musí mít alespoň jeden

1. **primární (master):** spravuje databázu záznamů domény
2. **sekundární (slave):** stahuje a uchovává kopii dat o doméně; "záloha"
3. **caching-only:** udržuje jen (ne)vyřešené dotazy po dobu platnosti

Obnovu databáze iniciuje většinou slave v závislosti na periodě v SOA záznamu. Master pouze, když cití, že je to třeba (velká změna dat).

3.5.3 Vyřizování DNS dotazu

Video.
www.mff.cuni.cz → rekurzivně na nameserver v doméně → adresu nemá v databázi → pošle nerekurzivně na kořenovy nameserver → nemá v databázi, takže vyhodnotí TLD → server uloží informaci o TLD do cache → pošle dotaz na navrhovany server → např. ns.cesnet.cz → opakuje se, dokud nedostaneme konečnou odpověď.
(odkaz posíláme kořenovému serveru cely, protože je DNS navrženo tak, že ve jméni může byt tečka)

Dotaz záhlaví - 2B číslo dotazu (ID), příznaky (požadavek na rekurzi), QUERY (RR obsahující jméno a typ: *www.cuni.czu IN A*).

Odpověď záhlaví - ID, příznaky (autoritativnost odpovědi), QUERY, Answer (RR s odpovědí), AUTHORITY (seznam nameserverů mohoucí dát autoritativní odpověď/ informaci), ADDITIONAL (dodatečné informace)

3.5.4 Bezpečnost v DNS

Náročné se dostat ke znění dotazu $\implies$ nemůže změnit odpověď.

cache-poisioning útočník donutí klienta poslat DNS dotaz, v sekci AUTHORITY změní informaci o tom, že server pro doménu cz je také útočníkův server (uloží do cache) $\implies$ úplná kontrola nad dotazy *řešení je neukládat do cache neautoritativní odpovědi*

DNSSEC rozšíření; podepisování záznamů klíčem; extrémně komplikované

4 Lekce

4.1 FTP

Video.

File Transfer Protocol textovy protokol k umožnění vzdáleného přístupu pro přenos dat (s otevřenym heslem). Dnes anonymní přístup *(místo hesla email)* k volně šířenym datům. Klient naváže řídící spojení na server (port **21**).

4.1.1 Kódy odpovědí

- 1xx - **Předběžne kladná odpověď:** server přijal požadavek a bude ho řešit.
- 2xx - **Kladná odpověď:** server úspěšně dokončil požadavek
- 3xx - **Neúplná kladná odpověď:** server přijal požadavek, je vyžadováno klientem doplnění informací *(např. jméno a heslo)*
- 4xx - **Dočasná záporná odpověď:** operace se nezdařila, je možné požadavek opakovat. Obvykle chyba serveru *(přetížení)*
- 5xx - **Trvalá záporná odpověď:** fatální chyba, opakování požadavku nebude úspěšné

4.1.2 Aktivní/ pasivní datové spojení

FTP má dodatečny datovy kanál, pomocí nějž přenáší veškery datovy obsah (download, upload souboru...)

Aktivní datové spojení navazuje server, na portu 20 s názvem *ftp-data*

Pasivní datové spojení navazuje klient, potřebuje od serveru adresu a port *(požádá příkazem PASV)*

4.1.3 Aplikace pro FTP

webové prohlížeče *(pro download stačí URL)*, Total Commander, Command app ftp *(příkazy jako normálně v UNIXu, pro přenos: mget, mput)*

4.2 SMTP

Simple Mail Transfer Protovol textovy protokol na TCP s portem 25, odesílá zprávy a odpovědi *(jako u FTP)*

4.2.1 Příjem a odeslání pošty v SMTP

odešli dopis → server zkontroluje adresu za zavináčem → přiřadí doménu → mail se předá serveru v LAN *(mail-forwarder) = mail-submission* → každy uzel, ktery přijímá a doručuje se nazyva MTA *(Mail Transfer Agent)*, chová se jako server a server → MTA pošle dalšímu MTA, dokud nejsme na správném serveru *(případně čeká ve frontě)*

MX záznam zabránění čekání ve frontě - nastevení v DNS, jméno *mail exchangeru* pro dočasné přijímání pošty a prioritu. Potom MTA zkouší doručovat podle priorit, dokud neuspěje.
Není vhodné pro velké soubory, neustálé ukládání do front v MTA uzlech

4.2.2 MUA

Mail User Agent zajišuje pro uživatele přístup do poštovního systému

1. přímé připojení - uživatel se připojí na MTA, kde má mailbox. Aplikace má přímy přístup k mailboxu a zároveň k lokálním MTA → odesílané zptávy přímo do fronty MTA
2. protokol POP / IMAP - klient se připojí na server (ktery je na MTA). Umožňuje pouze doručovat, odesílání musí provádět mail submission přes SMTP

4.2.3 Elektronicky dopis

Záhlaví (Hlavička) určeno pro informaci pro koncového uživatele, pro práci mailovych programů

- **Date:** Datum pořízení dopisu (americky formát)
- **From:** autor dopisu (”jméno ¡email¿”)
- **Sender:** odesílatel dopisu
- **Reply-To:** adresa pro odpověď
- **To:** adresát dopisu, příjemce
- **Cc:** Carbon Copy - adresáti kopie
- **Bcc:** Blind Carbon Copy - program přidá adreáta do obálky, ale ne do textu → ostatní příjemci neví, že byl mezi adresáty
- **Messege-ID:** identifikace dopisu, vyrábí program, pro pořadí vláken
- **Subject:** předmět dopisu
- **Received:** záznam o přenosu dopisu - uvedeny všechny uzly, které dopis přeposlaly

Možnost používat nonascii znaky v rozšíření ESMTP (8b kódování přenosu) nebo MIME (definuje strukturu a vyznam těla dopisu)

4.2.4 Diakritika v poště a kódování

Problém zažívali také UNIXové sítě UUCP (UNIX-to-UNIX copy), vymysleli UUENCODE.

UUENCODE vezmou se 3B původního souboru → rozdělí se do 4 skupin po 6b → 6b kódy se převedou na 4 tisknutelné znaky pomocí pevné tabulky. Zakódování má o 33% větší velikost.
Chybělo systematické začlenění do dopisu

Multipurpose Internet Mail Extension řeší strukturu dokumentu. Obsahuje *hlavičku* a *tělo*. Hlavička definuje:

- **typ dokumentu** - *(txt, html, jpg, ...)*
- **znakovou sadu a kódování dokumentu** - *(Base64, Quoted-Printable)*
- **původní název souboru**
- **způsob zpracování** - *(zobrazení jako příloha, vložení do textu apod.)*

Multipart: tělo je strukturované, aplikace vygeneruje náhodny řetězec, ktery odděluje části dokumentu

Base64 kódování UUENCODE s novou kódovací tabulkou (i velká písmena). Vyhodou je stabilní velikost vysledného kódu 133%.

Quoted-Printable každy non-ASCII znak se převede na sekvenci **=HH** *(HH je hex hodnota znaku)*. Pouze non-ASCII = 300% původního kódu a pouze ASCII znaky = 100% původního kódu.

4.2.5 Bezpečnost pošty a Spam

Open-relay-server server dovolí komukoliv, aby se připojil a poslal dopis kamkoliv, riziko hromadnych mailů

ESMTP rozšíření SMTP, obsahuje šifrovací příkaz *STARTTLS*, kterym zahájí SSL/TLS spojení. Využíváno ve firmách.

Gray-listing spamovací automaty nekontrolují úspšnost doručení → nejprve mail odmítne s odpovědí 450, vyčká a poté už soubor přijme s odpovědí 250 a doručí. Adresu non-spam uživatele si uloží do seznamu a neblokuje prvotní žádost po dobu obvykle 5ti tydnů. Adresu spam uživatele naopak napíše na ”černou listinu” a blokuje zprávy dál.

Sender Policy Framework doména definuje, jaké servery používá pro odeslání pošty, od nikoho jiného poštu nepřijme. Problém s přeposlanymi emaily.

DomainKeys Identified Mail (DKIM) odesílací stroj dopis podepíše (text a hlavičky) a příjemce zkontroluje podpis.

Antispam algoritmy odhadující pravděpodobnost, že mail je spam *(atributy jako: frekvence vyskytu slov, Subject pouze velkymi písmeny)*. Časté riziko *false positive*.

5 Lekce

5.1 Poštovní protokoly

Video.

5.1.1 POP

Post Office Protocol protokol pro přístup k počtovní schránce, bezpečnostní problémy a nevýhody *(otevřené posílání hesla, nutno stahovat cely dopis ze serveru, nelze pracovat se strukturou dokmentů)*

5.1.2 IMAP

Internet Message Access Protocol nástupce POOP, možnost šifrovaného spojení, vyhledávání v dopisů, stažení jen konkrétní části, podpora více schránek, uchování stvu dopisu na serveru...
Používá se šifrování TLS na portu 993 (nebo příkaz STARTTLS).

5.2 Web

Video.

5.2.1 Gopher

První celosvětově rozšířená služba distruované databáze - fungoval jako web, ale uměl pracovat pouze s textem

5.2.2 World Wide Web

Hypertext hierarchicky text, základní myšlenka vznikla po válce, později rozšířeno o netextové informace, implementováno v CERNU roku 1989 spolu s WWW

WWW obrovská distribuovaná hypertextová databáze (distribuováno na obrovském množství serverů), základní jednotka je hypertextová stránka, kterou server posílá na žádost klienta
Přenos stránek zajišťuje protokol HTTP, zabezpečení pomocí TLS (HTTPS)

5.2.3 Metody HTTP

Idempotence opakované použití má stejny efekt

- **GET:** klient žádá o poskytnutí stránky, požadavek=nic, odpověď=dokument, nemění obsah serveru, je idempotentní
- **HEAD:** zjednodušené GET, odpověď jsou pouze hlavičky
- **POST:** klient pošle na server nějaké informace a dostane zpět obsah dokumentu, může měnit obsah na serveru, není idempotentní
- **PUT:** má za úkol přepsat obsah dokumentu na serveru poslanym dokumentem z požadavku, je idempotentní
- **DELETE:** smaže dokument na serveru, je idempotentní
- **CONNECT:** pouze ověření spojení podle požadovanych parametrů, umožňuje budování tunelu pro realizaci jiného spojení na jiném protokolu (bezpečnostní riziko)

Hlášení:

- 1xx - **Informativní dpověď:** server přijal požadavek, zpracovává se.
- 2xx - **Kladná odpověď:** server úspěšně dokončil požadavek
- 3xx - **Přesměrování:** server přijal požadavek, je vyžadován od klienta další požadavek
- 4xx - **Chyba na straně klienta** nesprávny požadavek *(špatná URL, autentikace, ...)*
- 5xx - **Chyba na straně serveru:** požadavku se nedalo vyhovět

5.2.4 HyperText Transfer Protocol v.1

V podstatě pouze textová verze. Formát: úvodní řádka *(požadavek, odpověď)*, (doplňující hlavička - kódování, jazyk, typ dokumentu), prázdná řádka, tělo dokumentu
Z hlaviček je povinná pouze hlavička **Host** = na jaky server se klient obrací.
Vlastnosti:

perzistentní spojení klient nemusí zavírat TCP spojení a navazovat nové, může používat jedno spojení pro více požadavků

Cookies serverem vygenerovaná data na základě informací od uživatele poslána klientovi při odpovědi ve formě hlaviček *Set-Cookie*. Prohlížeč si je uloží a posílá hlavičku při každém požadavku na stejny server.

5.2.5 HTTP v.2

Změna z textového protokolu na binární.
Vytvořen koncept vlastních streamů provozovanych nad jedním TCP (vzájemně se neblokují, dají se prioritizovat)
Server může poslat více dat než klient požadoval, pokud usoudí, že budou potřeba (stránka s obrázky)
Možnost komprimovat hlavičky

HyperText Markup Language (HTML) textovy obsah stránky, doplněn formátováním, struktorou atd.

5.3 Vzdáleny přístup

Video.

5.3.1 Telnet

TELecommunication NETwork protokol pro přihlašování na vzdálené stroje, *(port 23)*
K dispozici je terminál, existují čtyři zprávy: *DO ECHO, DON'T ECHO, WILL ECHO, WON'T ECHO.*
Problémy: otevřeny přenos dat, riziko zacyklení

5.3.2 SSH

Secure SHell bezpečná náhrada Telnetu, šifrovaná komunikace, server je ověřen.
Umožňuje otevírat paralelně více kanálů *(můžeme byt přihlášeni v terminálu a zároveň přenášet soubory)*
Umožňuje tunelovat kanálem jiny provoz a zpřístupnit souborovy systém SSHFS

Bezpečnost na SSH Klient ověřuje server → kontrola klíče/ certifikát Server ověřuje klienta → heslo/ veřejny klíč/ pomocí vyzev a odpovědí Je třeba dávat pozor na MITM a na internetové červy (pokud máme recipročně nastavené přihlášení)

5.4 Voice over IP

VoIP není konkrétní protokol, ale obecně technologie pro přenos hlasu po IP.
Časté problémy jsou digitalizace hlasu, dohadování vlastností zařízení, propojení s telefonní sítí
Existují tři způsoby - standard H.323, standard SIP a proprietárně (např. Skype)

5.4.1 H.323

Vytvořeno společností ITU, založeno na ASN.1 .
Řeší všechny problémy, ale je stary a neprakticky. Protokoly jsou binární (doslova, nepracuje se s bajty ale s bity) - jsou nečitelné a těžko implementovatelné.
Nahrazován protokolem SIP.

Abstract Syntax Notaion 1 (ASN.1) formálně definuje datovou strukturu. Opět problém implementace - každá hodnota se zapíše pouze tolika bity, kolika je potřeba - řeší se nákupem knihoven, které vytvoří z textového zápisu ASN.1 kód, ktery zapis a čtení realizuje

5.4.2 SIP

Session Initiation Protocol textovy protokol se strukturou zpráv (podobná HTTP, informace v hlavičkách - uzly a cesty) pro navázání spojení, nahrazuje H.323
Může byt provozováno nad TCP i UDP.
Zavádí pojem **proxy** - článek komunikačního řetězu, ktery usnadňuje komunikaci přes hranice různych sítí (včetně privátních)
Řeší pouze problém vyhledání cíle, nalezení cesty a navazování spojení, zbytek řeší SDP

Session Description Protocol (SDP) txt protokol, přenášen pomocí SIP.
Po dohodnutí protisran se otevřou datové kanály a za pomoci protokolů **RTP/RTCP** se začne posílat audio

Ukázka hovoru přes SIP: volající pošle INVITE *(obashue URL a nabídku dat. kanálů jako SDP zprávu)* → příkaz dorazí na proxy → najde další uzel k volanému (nalezne volaného) → zkontroluje obsah SDP → volající odešle 100 Trying → volany vrátí 100 Trying → cílové zařízení začne vyzvánět 180 Ringing (end-to-end) → *ZVEDNUTÍ TELEFONU* → odešle 200 OK + SDP s datovymi kanály → zpráva dorazí na proxy → kontorola/ úprava SDP → volající potvrdí příjem SDP → ACK → *přenos audia v RTP/RTCP kanálech* → konec příkazem BYE → 200 OK

6 Lekce

6.1 Sdílení systému souborů

Připojení cizího disku z jiného počítače a pracovat s ním, jako by byl lokální disk.

6.1.1 NFS

Network File System otevřeny protokol jako RFC, použíá se především na UNIXu, autentizace prostředím Kerbenos
NFS pracuje nad UDP (je možné i TCP). Vnitřní struktura kopíruje OSI model - relační vrstva(Remote Procedure Call (RPC)) a prezentační vrstvu (eXchange Data Representation (XDR)).
Identifikace zdroje: *server:cesta*

6.1.2 SMB

Server Message Block původně vyvíjel IBM, nyní Microsoft, kvůli propojení UNIXu a Windows vznikla open implementace Samba. K autentizaci se používá jméno a heslo.
Identifikace zdroje: *server**cesta*

6.2 NTP

Network Time Protocol protokol pro synchronizaci času mezi uzly na síti - aby oba PC měly stejny čas; porovnávání času při řešení problémů
Existuje zdroj s absolutně přesnym časem: Stratum 0 → Stratum 1 → ... → Stratum N.
Podle Logické vzdálenosti od absolutně přesného zdroje, *tj. Stratum N ignoruje Stratum N+1.* Prevence před zacyklením.

Funkcionalita v LAN Stratum 0 → Stratum 1 → ... → ISP → několik NTP serverů s klienty
Problém se stupňujícím se zpožděním, řeší Murzellův algortimus (nalezne nejlepší průnik intervalů, ktery byl vytvořen podle časovych známek)

6.3 BOOTP a DHCP

Video.
Přiřazení IP adresy klientovi a předání základních informací o LAN

6.3.1 Bootstrap Protocol

vyvinut pro konfiguraci bezdiskovych stanic (nebylo možné uložit konfigurace, včetně IP)
BOOTP umožňoval vyslat žádost o přidělení adresy → server posoudil žádost podle MAC adresy → klient na seznamu → odeslání správné IP adresy klientovi

Limited broadcast klient pošle žádost všem uzlů v síti (ti je ignorují), protože nemá informaci o adresách, které se v síti nachází.
Nechceme, aby se žádosti šířily po internetu → směřovače nepropouštějí mimo síť → komplikace pokud má LAN složitou strukturu oddělenou routery

1. **řešení:** musíme mít BOOTP server v každé podsíti
2. **řešení:** na směrovačích spustit *BOOTP forwarding* - BOOTP dotazy přeposílá z rodné sítě určitému BOOTP serveru a jeho odpověď zase zpátky klientovi.

6.3.2 Dynamic Host Configuration Protocol

Nahradil BOOTP (zpětně komatibilní), zachoval formát zpráv, přidal dynamickou alokaci adres *(pevné přidělování ztratilo smysl)*, pronájem adres s časovym omezením *(doba pronájmu - lease-time)*, možnost práce s více servery v síti.

Průběh DHCP Klient pošle broadcast požadavek DHCPDISCOVER → DHCP servery pošlou nabídky DHCPOFFER → klient sbírá a posuzuje odpovědi (určity timeout) → vybere nejlepší odpověď → pošle DHCPREQUEST obsahující zvolenou adresu *(stále broadcastem musí dostat všechny navštívené servery, aby uvolnili adresu, kterou mu nabízely)* → server potvrdí DHCPACK → začne doba pronájmu → → v polovině pronájmu se klient ujistí DHCPREQUESTem svému serveru → pokud DHCPACK, má novou dobu pronájmu, pokud ne → v 7/8 doby pronájmu pošle broadcastem DHCPREQUEST → případně po uplynutí DHCPDISCOVER

Zjednodušeny průběh DHCP
Klient pošle DHCPDISCOVER (Broadcastem)
Server pošle DHCPOFFER
Klient pošle DHCPREQUEST *sám si vybere, který offer přijme*
Server pošle DHCPACK
Klient pošle DHCPREQUEST v 4/8 (buď DHCPACK nebo nic)
Klient pošle DHCPREQUEST v 7/8 (Broadcastem, opět DHCPACK nebo nic)
Klient pošle 8/8 DHCPDISCOVER

6.4 OSI vrstvy

6.4.1 Prezentační vrstva (OSI 6)

Snaha o vytvoření obecného mechanizmu, jak odstínit konkrétní architekturu uzlu sítě od formátu používaného nižšími vrstvami. (nezdařily pokus ASN.1)
TCP/IP přenechal starost na aplikaci, což je jednodušší.
Problémy: konce řádek *(Windows CR+LF; Appple = CR; UNIX = LF)*, pořadí bytů *big a little endian*

6.4.2 Relační vrstva (OSI 5)

Snaha o obecny model - jeden dialog může = více spojení, jedno spojení = více dialogů
TCP/IP opět přenechal starost na aplikaci.
Např. SMTP: jedno spojení, více mailů; SIP iniciuje dialog s více spojeními pro přenos audio/video dat

6.4.3 Transportní vrstva (OSI 4)

Poskytuje komunikační kanál pro přenos dat mezi aplikacemi na koncovych zařízeních *(end-to-end přenos dat).*
Zprostředkovává služby aplikačním protokolům
Umožňuje provoz více klientů a serverů na stejném uzlu.
Může zabezpečit spolehlivost přenosu dat *(reliable TCP, unreliable UDP).*
Může segmentovat a skládat data pro snazší přenos.
Může řídit tok dat - flow control, rychlost vysílání

6.4.4 Síťová vrstva (OSI 3)

Přenáší data získaná z OSI 4 od zdroje k cíli. Například IPv4, IPv6, ... Funkci vykonává za pomoci:
adresace - protokol OSI 3 definuje tvar a strukturu adres komunikujících partnerů
encapsulation (zapouzdření) - řídící data potřebná pro přenos se musí vložit do PDU
routing (směrování) - vyhledání nejvhodnější cesty k cíli přes mezilehlé sítě
forwarding (přeposílání) - předání dat ze vstupního síťového rozhraní na výstupní
decapsulation - vybalení dat a předání transportní vrstvě

6.5 Internet Protocol

Nespolehlivá služba - OSI 3 negarantuje doručení
Nespojovaná služba - datagramy se doručují nezávisly
Nezávislost na médiu - OSI 4 nemusí řešit použitou technologii

Přidělení IP rozhoduje správa sítě - privátní (tvoří sama) nebo veřejné uzly (tvoří ISP → jeden z 5 regionálních registrátorů (RIPE NCC, APNIC, ARIN, LACNIC, AFRINIC) → IANA)

6.5.1 IPv4

Hlavička ze : Verze (polovina bajtu), Délka hlavičky (druhá polovina bajtu), QoS, Fragmentace, TTL, číslo protokolu, kontrolní součet hlavičky, IP odesílatele, IP příjemce

Speciální IPv4 adresy

- **0.0.0.0/8** - potřebujeme komunikovat, ale neznáme svoji adresu
- **127.0.0.1/8** - adresa lokálního počítače, loopback adresa = umožňuje vytvoření smyčky
- **xxx.xxx.0.0** - adresa sítě (PC část pouze nuly)
- **xxx.xxx.1.1** - broadcast (PC část pouze jedničky), chceme-li oslovit všechny PC v síti

- **255.255.255.255** - omezeny broadcast, oslovuje všechny v síti; nesmí opustit síť
- **privátní adresy** - *10.0.0.0/8* (1 třída A); *172.16-31.0.0/16* (16 tříd B), *192.168.*.0/24* (256 tříd A)

6.5.2 Subnetting

Video.
Umožňuje rozčlenit síť na menší celky, redukuje počet použitelnych adres (ne samé nuly/ jedničky $\implies$ máme jen 70% adres)

Síťová maska číslo obsahující jedničky na místech, kde je adresa - pro ověření stačí bitové AND

Variable Length Subnet Mask (VLSM) pokud se v síti používají různé masky

Supernetting posunutí hranice opačnym směrem

6.5.3 IPv6

Má oproti IPv4 128b (16B), hex zápis: *fec0::1:800:5a12:3456/64* (možné zkrátit sekvenci nul ::)
Unicastové adresy - základní typ adres → *loopback, link-local, unique-local (obdoba privátních z IPv4)*
Multicastové adresy - nahrazení broadcastovych adres
Anycastové adresy - unicastová adresa, přidělená více uzlům; distribuuje servery po světě

6.6 Přenos dat v TCP/IP

6.6.1 UDP

V UDP hlavičce: informace o multiplexingu (zdrojovy, cílovy port, délka a kontrolní součet)

6.6.2 TCP

Hlavička (max 60B) - zdrojovy + cílovy port, sequence number, acknowledge number, posun (vúči počátku streamu), flags, okno, kontrolní součet, urgent pointer, ostatní

TCP okno navrhne se velikost okna → odesílatel posílá data → dosáhne velikosti okna → pokud přijde ACK, posune offset o velikost poslanych dat → pokud nepřijde, přeruší spojení a znovu pošle poslední blok dat

Three-way handshake tři pakety s prázdnou datovou částí nesoucí v hlavičce informaci o náhodném sekvenčním čísle

1. **paket:** (klient) poslání příznaku *SYN (synch. paket)*, *Seq. num* číslo c a *Ack. num* s hodnotou 0
2. **paket:** (server) potvrzení příznakem *SYN + ACK*, *Seq. num* číslo s a *Ack. num* s hodnotou $c+1$
3. **paket:** (klient) poslání *ACK*, *Seq. num* číslo $c+1$ a *Ack. num* s hodnotou $s+1$

Ukončení spojení

1. **paket:** (klient) poslání příznaku *FIN*, *Seq. num* číslo x a *Ack. num* s hodnotou y
2. **paket:** (server) potvrzení příznakem *ACK*, *Seq. num* číslo y a *Ack. num* s hodnotou $x+1$

TCP příznaky

- **SYN:** k synchronizaci čísel segmentů (inicializace *Sequence number*)
- **ACK:** potvrzení doručení až po *Acknowledge number*
- **PSH:** informuje příjemce o obdržení kompletního bloku dat *(push)*
- **FIN:** odesílatel zavírá svou stranu spojení
- **RST:** odesílatel odmítá přijmout spojení *(reset)*
- **URG:** paket obsahuje urgentní data, adresa v *Urgent pounteru*

7 Lekce

7.1 Směrování

Video.
Měla by umět každá stanice v TCP/IP.

7.1.1 Směrovací tabulka

"cedule na křižovatce", cíl obsahuje adresu sítě a její rozsah zadaný síťovou maskou
Např: chceme z 1.1.1.1 na 3.0.0.0/8 $\implies$ 1.1.1.1 $\rightarrow$ Gateway 1.0.0.1 $\rightarrow$ 2.0.0.3 $\rightarrow$ 3.0.0.0/8

- **Adresa sítě** - destination
- **maska (rozsah)** této sítě - netmask
- **Gateway**:
  - *next-hop router*, předá se paket, pokud vede do cizí sítě
  - adresa některého vlastního síťového rozhraní, pokud se jedná o záznam přímý (vede do přímo připojené sítě).

Typy záznamů:

- **Direct** - přímo připojená síť - gateway
- **Indirect, default** - směřuje na ISP

Vznik záznamů:

- **Implicitní** - automaticky po pořízení adresy
- **Explicitní** - ručně zadán
- **Dynamický** - v průběhu práce od parametrů v síti

7.1.2 Směrovací algoritmus

V tabulce se najdou všechny záznamy shodující se s cílem paketu $\rightarrow$ pokud neexistuje, nelze doručit (pokud neobsahuje Defaultní záznam) $\rightarrow$ z nalezených se vybere ten s nejširší maskou $\rightarrow$ pokud odkazuje záznam na můj PC, vrátí na vstup (loopback) $\rightarrow$ záznam přímý (v mojí síti), pak se odešle příjemci $\rightarrow$ záznam nepřímý, pošle se směrovači (next-hop router)

7.2 ICMP

Internet Control Message Protocol mezi OSI 3 a OSI 4; přeposílá řídící informace ve formě datagramů, např Echo/ Echo Reply, Destination Unreachable *(router nemá jak doručit paket, zahazuje ho)*, Time Exceeded, Source Quench *(snížení rychlosti toku datagramů)*, Redirect, Parameter Problem

7.2.1 Ping

Základní prostředek pro diagnostiku sítě, testuje dostupnost uzlu.
Periodické odesílání **ICMP Echo**, pokud zpráva dorazí, pošle **ICMP Echo Reply** a vypíše dobu přenosu.
Na ICMP odpovídá síť sama. Není zaručena dostupnost sítě, pouze její vrstvy.

7.2.2 TTL (IP)

Ochrana před zacyklením v ICMP **Time Exceeded**. Udává počet hopů, **NE ČAS**, které smí paket ještě přeskočit, dosáhne-li 0, nastane ICMP Time Exceeded

7.2.3 Diagnostika směrování

Výpis routovací tabulky `netstat -r` nebo `route`.
Kontrolu cesty zajišťují `traceroute` nebo `tracert` - odhalí strukturu cesty, dokáže nalézt konkrétní chybný směrovač

7.2.4 Statické řízení směrovacích tabulek

Nejjednodušší řízení tabulky, vhodné pro jednodušší sítě. Počítač má někde uložené informace pro všechny záznmy, které po bootu postupně přidává.
Klady: stabilita, dostupnost
Zápory: nepružné při změnách v síti, problém se subnettingem, složité zálohování

Redirekce umožňuje pokrytí složitější sítě staticky, za pomoci **ICMP Redirect**.
Router zkusí poslat paket do sítě bez záznamu, redirekce přidá síť do routovací tabulky, nyní možné paket poslat. *(Není bezpečné, někdo může šířit vadné ICMP Redirect pakety a způsobit nefunkčnost sítě.)*

7.2.5 Dynamické řízení směrovacích tabulek

Založen na informacích, které si vzájemně vyměňují sousední routery a na základě nichž si upravují svoje tabulky.
Klady: jednoduchá změna konfigurace, síť se umí sama "opravit", tabulky se udržují automaticky
Zápory: citlivé na problémy a útoky
Je zapotřebí pro komunikaci v protokolu speciální software - BIRD, routed, ...
Pro lokální sítě jsou používány protokoly *distance-vector* a *link-state*.

7.2.6 Distance Vector protokoly

Uzel má u záznamů v tabulce i "vzdálenosti". Tabulku periodicky posílá sousedům, kteří ji upraví a posílají dál.
Klady: jednoduchost, snadná implementace
Zápory: pomalá reakce na chyby, nezohledňuje vlastnosti linek (rychlost, spolehlivost, cenu...), omezený rozsah sítě, chyba ve výpočtu jednoho routeru ovlivňuje celou síť (zacyklení)

Routing Information Protocol (RIP) zástupce distance-vector protokolu, velmi rozšířený.

- **Vzdálenost** počet routerů v cestě (hop count).
  Není vhodné, protože nepreferuje nejlepší, ale nejkratší cestu (80km po vedlejší **VS** 100 km po dálnici).
- **Counting to Infinity** je omezen na 15 hopů (7×30s) - po odpojení sítě, stále se tváří jako dostupná (i když není), prevence před zacyklením
- **Algoritmus** pro výpočet nejkratší cesty se využívá Nellman-Fordův algoritmus (ne rychlejší Dijkstrův → nemusí se opakovat celý výpočet)

7.2.7 Link State protokoly

Každý router zná mapu celé sítě, posílají se informace o stavech linek, přepočítává si router sám optimální cesty.
Klady: pružná reakce na změny/ výpadky; každý router si počítá sám *(chyby neovlivní ostatní)*,
síť je možné rozdělit na menší podsítě (zrychlení); výměna dat probíhá pouze při změnách (ne v intervalech)
Zápory: výpočet mapy je náročnější na výkon CPU i na paměť, zátěž sítě

Open Shortest Path First (OSPF) nejznámější představitel lin-state

- **Vzdálenost** tzv. cena cesty *(path cost)*, určuje se pomocí složitého vzorce.
  Umí zahrnout šířku pásma, latenci, propustnost, skutečnou „cenu“ provozu lince.
- **Algoritmus** pro výpočet nejkratší cesty se využívá Dijkstrův algoritmus
- **Hierarchické rozdělení** - výpočet probíhá na menší množině uzlů
  - **Oblast 0** - páteř (backbone)
  - **Ostatní oblasti** - připojují se na páteř
  - každý router zná mapu své oblasti a cestu k páteři

7.3 Autonomní systém

AS blok sítí se společnou routovací politikou. Určen pro směrování ve vzdálených sítích.

Externí Routovací Protokoly (EGP) pro řízení routování mezi AS, snazší routování na globální úrovni

Gateway Protocol (BGP) nejpoužívanější zástupce EGP *(path-vector = posloupnost čísel, přes než vede cesta, zabraňuje zacyklení)*

7.4 IP filtrování

Filtrování na transporní vrstvě, stanovuje pravidla pro typ provozu a za jakých podmínek.

Přísná konfigurace : ven vybrané (konkrétní porty), dovnitř nic

- vhodné pro protokoly s jedním kanálem (HTTP, SMTP)
- problém u protokolů s více kanály (FTP, SIP)

Obvyklá konfigurace : ven cokoliv, dovnitř nic

- naráží např. u FTP s aktivním přenosem
- nepoužitelné u protokolů s mnoha kanály (SIP)

Velký problém při poskytování služby z vlastní sítě *(např. www server, pošta)*

- musí se ve filtru otevřít permanentní díra povolující provozu z vnější sítě přístup na konkrétní server a port
- řeší se odděleným segmentem DMZ *(demilitarizovaná zóna)* - benevolentnější filtr

7.5 Proxy server

Video.
Software, který kontroluje provoz určitého protokolu *(obvykle na rozhraní lokální sítě a internetu)*.

Transparentní provoz

- Router zachytí klientův požadavek, uloží ho, připojí se na server, zkontroluje (bezpečnost) a odešle ho
- Odpověď přijde zpět na router, ten ji uloží (pro další klienty) a zároveň odešle původnímu žadateli.
- Není třeba měnit na klientovi konfiguraci

Netransparentní provoz

- Proxy server nemusí být nutně router, ale vhodnější zařízení.
- Nutná podpora v protokolu.
- Klienty je třeba nakonfigurovat, aby se požadavky neposílaly přímo, ale proxy-serveru v lokální síti.

Bezpečnost a výkonnost

- Umožňuje efektivně kontrolovat činnost klientů *(může filtrovat operace na úrovni aplikačního protokolu)*
- Umožňuje omezit provoz na připojné lince *(nemusí opakovat stejný požadavek, odpověď může do cache a žádajícím poslat sám)*

8 Lekce

8.1 ARP

Video.

Address Resolution Protocol spojovací článek mezi síťovou a linkovou vrstvou. Umožňuje uzlům v síti zjišťovat linkové (MAC) adresy odpovídající konkrétním síťovým adresám.
Neznámé adresy se zjišťují broadcastovou výzvou, krom držitele výzvu všichni ignorují. Hledaný uzel *(ARP server)* na dotaz zareaguje unicastovou ARP odpovědí s požadovanou MAC adresou.
Výsledky se vkládají do *ARP cache* (stejně tak server).

Bezpečnostní rizika: odpovědět na broadcast může kdokoli; *nevyžádané (gratuitous) ARP* je odpověď bez dotazu - 2 stroje se stejnou IP, jeden informuje ostatní o svojí MAC.

8.1.1 Proxy ARP

Ve složitých LAN, kde správné směrování zajišují směrovače.
Služba, která zachytí ARP dotaz, a protože pozná, že by se tazatel nikdy nedočkal odpovědi (linková síť), pošle mu svou odpověď místo stroje B a jako hledanou adresu uvede svoji MAC adresu
Princip

1. host A posílá broadcastem `ARP request` s IP adresou B *(A pracuje na linkové síti, kde není B)*
2. router pozná, že dotaz **nebude zodpovězen**, sám posílá `ARP reply` s MAC adresou routeru
3. přiřadí se **MAC routeru** k **IP adrese B** a uloží do ARP cache na A
4. host A nyní posílá data pro B s **MAC adresou routeru**

Možné pozorovat při náhledu do ARP cache - více IP adres se stejnou MAC adresou *(pokud nemáme v síti ARP proxy, jedná se o napadení)*

8.2 Linková vrstva (OSI 2)

Přesun od SW k HW, přesun mimo TCP/IP; dělí se na dvě podvrstvy:

- **Horní - Logical Link Control (LLC)** umožňuje různým protokolům síťové vrstvy přístup ke stejnému médiu *(multiplexing)*
- **Spodní - Media Access Control (MAC)** řídí *adresaci uzlů* a *přístup k médiu* - kdo, kdy a jak může data odesílat a jak je přijímat

Frame Check Sequence (FCS) hodnota sloužící ke kontrole správnosti doručeni.

Rámec (frame) datová jednotka linkového protokolu, formát obsahuje:

- **Synchronizační pole** - sekvence bitů, probouzí cílovou stanici a odlišuje data od šumu.
- **Hlavička** - obsahuje MAC adresy příjemce, odesilatele a řídící informace LLC.
- **Data (payload)** - nadřazeného protokolu.
- **Patička** - obsahuje hodnotu sloužící ke kontrole správnosti doručení - *Frame Check Sequence (FCS)*

8.3 Síťové topologie

8.3.1 Multipoint

Umí propojit více uzlů. Technologie:

- **Sběrncie** - všechny uzly sériově na stejném médiu. Přerušení kabelu, rozpad celé sítě. Všechny uzly se pokouší poslat zprávu tzv. Kolize.
- **Hvězda** - nejobvyklejší topologie; obsahuje centrální prvek, uzly jsou vázány na něj. Sběrnice v centrálním prvku, kabeláž pouze mezi koncovými stanicemi.
- **Kruh** - jednotlivé uzly jsou propojeny do kruhu

Řízení přístupu uzlů k médiu:

- **Deterministicky** - někdo/něco deterministicky určuje, kdo právě smí vysílat
- **Nedeterministicky** - nikdo neomezuje uzly ve vysílání, řešení kolize

8.3.2 Point-to-point

Propojení pouze dvou uzlů (např. RS-232).
Dosah Point-to-point je možné prodloužit:

- **Kabel** - stále se používá stejný protokol jako v multipoint zapojení.
- **Modem** - linky nahradí modem, který moduluje datový provoz pro přenášení pomocí telefonního spojení.
- **Bezdrát** - lasery

Zzel dokáže nebo nedokáže současně přijímat i vysílat:

- **Half duplex** - nedokáže → kolize pri zapojení Ethernetu
- **Full duplex** - dokáže → zbavení se kolizí

8.3.3 Řešení kolize

Nosná = přenosové médium

Carrier Sense with Multiple Access (CSMA) "kontrola nosné", uzel čeká, dokud není nosná volná

Collision Detection (CSMA/CD) např. Ethernet - je schopen detekovat kolizi, kontroluje během vysílání nosnou;
při kolizi přestane vysílat, upozorní ostatní, náhodnou dobu počká a pak opakuje pokus

Collision Avoidance (CSMA/CA) např. WiFI - hvězdicová topologie, centrální prvek je Access point (AP), stanice připojeny k AP, všechny je v podstatě point-to-point; pokud nedorazí ACK, zahájí se exponenciální čekání

8.3.4 Ethernet

Video.
Vznikl ve firmě Xerox (kopírky), převzala IEEE (standardizace IEEE 802.3 - normy proprietární) Vůdčí technologie pro lokální sítě - pružně reaguje na vývoj HW.
Kolize se řeí za metodou CSMA/CD. Exponenciální končí po 16 pokusech chybou.
Adresa - 3B prefix (multicast,...), 3B adresa

8.3.5 VLAN

Video.

Virtální sítě LAN po jedné fyzické síti lze provozovat více nezávislých LAN
Ethernetový rámec je prodloužen o 32 bitový tag (tagovat může switch, pro koncovou stanici transparentně)
VLANID: 12B identifikátor označující sítě
Vsune se 4B úsek do rámce za MAC → změní se rámec na VLAN → vložený úsek nese info o VLANID → vše se odehrává transparentně → stanice se připojí do portu switche (součást sítě) → tváří se jako obyčejná síť
Trunk: uzly, které potřebují mít přístup k rámcům ze všech virtuálních sítí

8.3.6 CRC

Cyklický kontrolní součet hashovací funkce pro kontrolu dat *(např. v hlavičce IP, ...)*
Funkce založena na dělení polynomů - polynom se zapíše binárně a vydělí se charakteristickým polynomem
Zbytek po dělení je považován za výsledek, převeden na bity a použit znovu jako hash.

8.3.7 WiFI

Video. Video - AP

Wirelss LAN skupinu protokolů IEEE 802.11 ve frekvenčních pásmech 2,4 a 5 GHZ a kanálech (jen 13)
Řeší kolize za pomoci CSMA/CA a má hvězdicovou topologii - ve středu AP *(obvykle, může být i p2p)*, kte kterému se připojují koncová zařízení.

8.4 Fyzická vrstva (OSI 1)

Přenáší data úp fyzickém médiu, převádí digitální informace na analogovou a zpět

Používaná média metalická *(eltekrické pulzy)*; optická *(světelné pulzy)*; bezdrátová *(modulace vln)*

8.4.1 Druhy přenosu dat

Analogový, digitální vše analogové; digitální rozhoduje, zda hodnota signálu spadá do nějakého intervalu; D→A modem, A→D codec

Baseband, broadband basebend přenáší signál a kóduje ho (Ethernet - Manchester); broadband přenáší signál v širokém pásmu a moduluje ho (AM, FM)

8.4.2 UTP

Video.

Unshielded Twisted Pair připojuje stanice v LAN, v kabelu je 8 vodičů
Dva vodiče navzájem zakroucené vytváří při průchodu proudu ochranné (proti rušení) elektromagnetické pole (Alternativou je kabel s kovovým stíněním - STP)
Dvěmi páry protéká 100Mbps, takže je možné rozdělit na další dva páry.

8.4.3 Optická vlákna

Signál se šíří jako světelné pulzy křemíkovým vláknem
Výhody: nemá problém s rušením; velmi nízký útlum; velká šířka přenosového pásma *("rychlost")*.
Nevýhody: cena; náročná manipulace *(velký minimální poloměr ohybu)*.
Druhy vláken:

- **Jednovidá (singlemode)** - svítí se laserem → jeden paprsek, větší dosah (omezený lom) a šířka pásma
- **Mnohovidá (multimode)** - svítí se i LED → paprsky hodně se lámou

8.4.4 Segmentace sítě

Video.

Repeater (opakovač) spojuje od sebe vzdálené stanice, distribuuje signál na fyzické vrstvě; v kabeláži *hub*; řeší dosah signálu; zhoršuje kolizi

Bridge (most) řeší propustnost, distrib signál na linkové vrstvě; v kabeláži *switch*; snižuje kolizi

8.4.5 Learning bridge

Video.
Přepínače si udržují samy připojené MAC adresy na konkrétním portu - udržují si pro každý port tabulku MAC adres
Switch posílá všechny rámce do správných portů, *(krom broadcastů, neznámých unicastů (BUS) a multicastů (BUM)*.

8.4.6 Spanning Tree Algoritmus

Pokud pracují oba switche, síť se zaplaví přeposíláním rámců a learning bridge selže.
Algoritmus hledá acyklickou podmnožinu, kostru (spanning tree) v cyklickém grafu.
Switche se musejí dohodnout, který z nich bude mít potlačeno forwardování a bude pouze monitorovat provoz

Spanning Tree Protocol (STP) protokol pro hledání koster grafu sítě; je pomalý cyklech