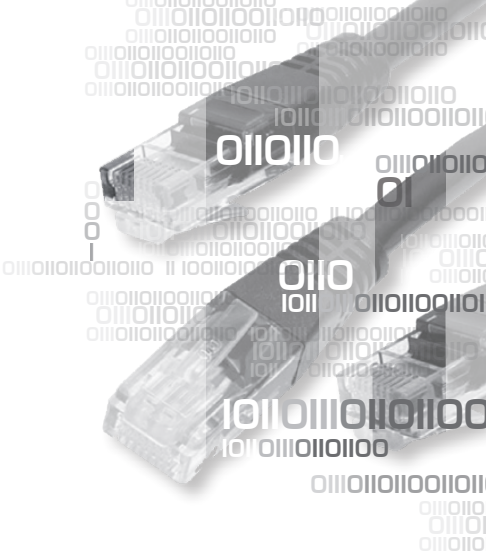
POČÍTAČOVÉ SÍTĚ

PRAKTICKÁ PŘÍRUČKA SPRÁVCE SÍTĚ

Ivona Spurná

- Princip fungování sítí
- Přenosová média
- Popis síťových zařízení a protokolů
- Užitečné návody a postupy pro správu sítí

Nakladatelství a vydavatelství
ComputerMedia
Vzdělávání, které baví
www.computermedia.cz



Počítačové sítě

Praktická příručka správce sítě

Ivona Spurná

Nakladatelství a vydavatelství
ComputerMedia®
Vzdělávání, které baví
www.computermedia.cz

Počítačové sítě
Praktická příručka správce sítě

Autor: Ivona Spurná
Návrh vnitřního layoutu: Pavel Navrátil
Zlom a sazba: Jan Paroulek
Návrh obálky: Ing. Michal Jiříček
Jazyková úprava: PhDr. Dagmar Procházková

Vydání první, 2010

ISBN: 978-80-7402-036-0

Computer Media s.r.o.
Hrubčická 495
798 12 Kralice na Hané
Česká republika

Telefon: +420 582 302 666
Fax: +420 582 302 667
E-mail: info@computermedia.cz
Web: http://www.computermedia.cz

Kompletní nabídku literatury Computer Media s.r.o. můžete získat i objednat na internetové adrese **www.computermedia.cz**.

Nakladatelství a vydavatelství
Computer Media®
www.computermedia.cz

Obsah

1. Úvod

Současná doba je typická využíváním počítačových sítí, ať už z pohledu získávání informací, publikování vlastních myšlenek, nebo zábavou, komunikací, studiem i prací přes internet, nákupem a prodejem zboží pomocí webových stránek atd…

Díky počítačovým sítím a zejména internetu se v současnosti děje mnoho věcí snáze a rychleji. Získávání a předávání informací je díky vyhledávačům a elektronické poště rovněž snadné a rychlé. A to je jen střípek z celkového množství výhod a služeb, které lze dnes pomocí počítačových sítí využívat.

V této knize se zaměříme na principy fungování počítačových sítí, rozebereme, jakým způsobem se připravují a zpracovávají data přepravovaná po sítích, vysvětlíme základní pojmy z oblasti počítačových sítí a základní principy fungování síťových zařízení. Podrobně se zaměříme na IP adresaci, nastavení a testování sítě.

Síťové prvky

Na přenosu dat spolupracují síťová zařízení, přenosová média a pravidla přenosu nazývaná protokoly.

Za **síťová zařízení** se považují například **osobní počítače, notebooky, servery, IP telefony, směrovače (router), přepínače (switch), rozbočovače (hub)**.

Přenosová média jsou **optická** (optické kabely), **metalická** (UTP, STP, **koaxiální kabel** – měděné kabely), **bezdrátová** (prostor, atmosféra).

- Optickými kabely se data vedou jako **světelný signál**, měděnými kabely jako **elektrický signál**, a v případě bezdrátového přenosu, kdy je přenosovým médiem atmosféra, resp. prostor, se data přenášejí elektromagnetickými vlnami, například **mikrovlnami**.

Pravidla pro přenos se nazývají **komunikační protokoly**. Jsou to soubory informací, které definují, jak se s přenášenými daty během přenosu sítí nakládá.

Nejpoužívanější sadou protokolů v lokálních sítích i na internetu jsou **protokoly TCP/IP** (***Transmition Control Protocol/Internet Protocol***).

- Patří sem například **protokol HTTP** (***Hypertext Transport Protocol***) potřebný pro provoz služby **WWW** (***World Wide Web***), **SMTP** (***Simple Mail Transport Protocol***) a **POP** (***Post Office Protocol***) potřebné pro provoz služby **E-mail**.

Náhled na proces odesílání dat

Co se děje se zprávou, obrázkem, videem, prostě s čímkoliv, co je potřeba odeslat po síti? Bez ohledu na to, jaká data se odesílají, musí se nejprve zkonvertovat do podoby binárního kódu, který se následně vysílá po síti. Data v podobě bitů vysílá na síť **síťová karta**.

Po síti mohou data putovat pomocí různých médií, pomocí měděných nebo optických kabelů nebo bezdrátově. Pro spojení různých sítí se používají **směrovače** – **routery**. Ty zajišťují přesměrování z jedné sítě, například naší domácí, do jiné sítě, například veřejné sítě Internet. Zajistí, že data doputují do svého cílového zařízení tou nejrychlejší a nejvhodnější cestou. Ta může být různě dlouhá, po cestě mohou data projít mnoha různými sítěmi. Data jedné komunikace mohou putovat do svého cíle různými cestami.

V závěru své cesty jsou data přijímána cílovou síťovou kartou a převáděna do své původní podoby. Data putující po síti v sobě nesou velké množství dalších přídavných informací, které například určují, kam mají být doručena a pro jakou aplikaci jsou určena.

Podrobnější popis, jak se vytváří proud bitů určených k vysílání po síti a jak se s tímto proudem bitů nakládá v cíli, najdete v následujících kapitolách.

Stavba internetu

Internet je supersíť spojující nejrůznější privátní i veřejné sítě, má hierarchickou stromovou adresní strukturu. Internet není regulován žádnou jednotlivou organizací, ale pro úspěšný chod musí jednotliví operátoři zajišťující spojení spolupracovat a dodržovat určité standardy a protokoly.

V poslední době výrazně rostou možnosti jak využívat internet nejen k základnímu spojení a komunikaci, ale také například k přenášení hlasu, videa, velkých souborů dat, a to s sebou nese potřebu vytváření odolných, spolehlivých a rozšiřitelných sítí.

Ochrana proti chybám a výpadkům

V případě, že některé síťové zařízení přestane správně pracovat, musí je nahradit jiné, data si pak najdou náhradní cestu pro putování k cíli. Na přenášených datech se přitom z pohledu uživatele nic nezmění. Zobrazí se v cílové aplikaci nezávisle na cestě, kterou musela urazit.

Tato vlastnost byla jedním z prvních požadavků na funkčnost sítě. Na počátku se jednalo o telefonní síť.

Pro uskutečnění hovoru se nejprve musely nastavit všechny telefonní přepínače mezi cílovými zařízeními. V případě výpadku některého zařízení po cestě se hovor ztratil a nebyl schopen se znovu automaticky navázat, a proto celý proces nastavení musel začít znovu od začátku.

Takový typ sítí se nazývá **spojované sítě s přepnutými obvody**. Existence mnoha těchto sítí, které existují od svého počátečního nastavení mezi zdrojovým a cílovým zařízením bez ohledu na to, zda jimi v danou chvíli proudí data, se zdála být neefektivní a neúměrně drahá, z toho důvodu se začalo hledat jiné řešení.

Takovým řešením jsou **nespojované sítě s přepínáním paketů**.

Princip přenosu těmito sítěmi spočívá v tom, že data ze zdroje jsou rozdělena do mnoha menších dílů zvaných **pakety** a ty potom putují sítí ke svému cíli různými cestami, podle aktuální situace v síti. *Problematika paketů bude probrána podrobně později.* Nesou v sobě zakódované informace o svém zdroji a cíli. V cíli se pak jednotlivé díly dokážou poskládat do své původní podoby. Mezi zdrojem a cílem není vytvořen žádný trvalý okruh, jednotlivé pakety mohou putovat nezávisle na sobě různými cestami. Vždy, když paket doputuje na nějaký směrovač v síti, ten udělá rozhodnutí, kam paket poslat. Pokud nějaká cesta selže, najde jinou alternativní. Jestliže se nějaký paket při své cestě po síti ztratí, je vyslán znovu. Tento způsob přenosu dat má tu vlastnost, že uživatelé na síti sdílejí pro přenos dat celou síťovou infrastrukturu. V případě výpadku části sítě dokážou směrovače najít náhradní cestu, což pomáhá omezovat výpadky.

Přestože jsou tyto nespojované sítě s přepínáním paketů dnes hlavním způsobem přenosu dat přes internet, stále ještě se mohou díky svým výhodám využívat i spojované sítě. Například v případě, kdy uživatelé chtějí mít vyhrazenu určitou linku s garantovanou kvalitou přenosu. Poskytovatelé takové služby si pak mohou nechat platit za dobu, kdy je takové vyhrazené spojení uskutečňováno.

Schopnost růstu

S přibývajícím počtem uživatelů a síťových zařízení v síti souvisí požadavek, aby se síť mohla stále rozrůstat a nemělo to negativní dopad na stávající uživatele.

Struktura internetu je stromová. Na vrcholu je tzv. **páteřní síť,** k níž se připojují **regionální sítě.** Ty poskytují připojení k internetu jednotlivým poskytovatelům internetu, kteří tuto službu nabízejí dalším poskytovatelům internetu a koncovým uživatelům.

Adresace je hierarchická, takže každý **DNS** (***Domain Name System***) server ví jen část informací, které přísluší jeho umístění v síti.

Díky hierarchii se síť vyššího umístění nezatěžuje požadavky, jež může vyřídit síť nižšího umístění, která je blíže k původci požadavku.

Zajištění kvality přenosu

Některé aplikace, jako například zobrazení webových stránek, nevyžadují tak kvalitní a spojitý přístup k internetu, jinými slovy – mohou na příchozí data chvíli počkat, na rozdíl například od spojitých hlasových a video přenosů, které vyžadují pro svůj zdárný průběh stabilní přístup na síť. Některé aplikace mají přednost před jinými. S tímto hlediskem se dnes také musí vypořádat navrhovatelé sítí – musí do sítí zařadit a správně nakonfigurovat příslušná síťová zařízení, která jsou schopná požadovanou kvalitu zajistit. Taková zařízení se pak dokážou rozhodnout, který typ síťového provozu má přednost před jinými.

Protože v sítích může docházet k zahlcení, kdy určitým místem proudí více paketů, než kolik jich v danou chvíli může síťové zařízení odbavit, dochází k tomu, že se pakety řadí do fronty. Pokud doba čekání přesáhne určitou hranici, pakety jsou zahozeny. Někdy je možné například zvýšit **šířku pásma** (angl. ***bandwidth***) a tím dosáhnout lepší propustnosti, ale i toto má svá omezení.

Kvalitní přenos je možné zajistit také klasifikací priority přenosu – některým typům přenosu lze přiřadit vyšší, a jiným naopak nižší prioritu (například stahování webových stránek nebo e-mailů může mít nižší prioritu než stahování videa či zvuku nebo přenos obchodních dat, kdy je spojitý a rychlý přenos žádoucí). Pakety s vyšší prioritou jsou pak pouštěny linkou častěji než pakety s nižší prioritou.

Bezpečnost

V architektuře internetu má své místo také bezpečnost přenosu dat a zajištění soukromí. Využití nachází například v různých obchodních aplikacích, kdy je zajištění bezpečného přenosu, který nelze snadno odposlechnout nebo jednoduše rozšifrovat, prioritou.

Síťové prvky, jimiž data při své cestě internetem procházejí, musí být zabezpečené, aby se k nim nemohl fyzicky nebo i vzdáleně dostat narušitel, který by pak mohl tato data odposlechnout a případně zneužít.

Je potřeba zajistit **důvěrnost přenášených dat** tak, aby zprávu (obecně jakákoliv data) mohli přečíst pouze odesílatel a příjemce. Data mohou být šifrována a soubory zabezpečeny hesly, která nelze snadno uhodnout.

Dále je možné zajistit **integritu přenášených dat** tak, aby nebyla průchodem sítí změněna, ať už záměrně, nebo náhodně. K tomu slouží například digitální podpis nebo kontrolní součty (určitá početní operace provedená na zprávě dá určitý výsledek, který musí být stejný i po průchodu sítí, pak je pravděpodobné, že data nebyla změněna).

Aby mohla být síť využívána, musí být především **dostupná**. Určité aktivity mohou způsobit nedostupnost sítě.

Jsou to například **viry** nebo útoky typu **DoS** (angl. ***Denial of Service*** – způsob, jak zapříčinit nedostupnost sítě). Jedním z DoS útoků může být například zfalšovaný **ping** (příkaz pro ověření délky odezvy a dostupnosti sítě – *podrobně bude probrán později*), kdy je adresa zdrojového zařízení zfalšovaná, a proto nedostupná. Cílové zařízení se pak snaží na tento **ping** odpovídat, ale protože nedostává potvrzení o doručení, snaží se odpovídat stále znovu. Tím zahlcuje svou linku, čímž se síť stává nedostupnou. Tuto skutečnost lze snadno ovlivnit například tím, že zařízení bude nastaveno tak, aby na **ping** neodpovídalo. K potlačení podobných nežádoucích aktivit lze využít firewally s antivirovými systémy.

Firewall je hardwarové zařízení nebo aplikace, které chrání vnitřní síť před vnějšími útoky a nežádoucím přístupem.

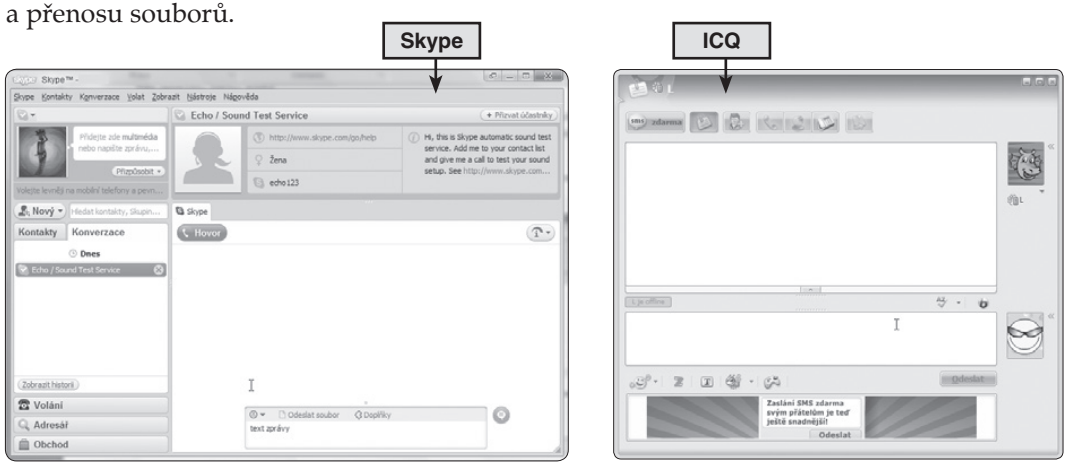
Softwarový firewall bývá často součástí směrovače, ale je nutno dát na těchto zařízeních pozor na přednastavená hesla, která bývají obecně známá. Pak by takové zařízení bylo snadno napadnutelné.

Způsoby komunikace

Pokud vám nevyhovuje komunikace s protějškem přes elektronickou poštu – e-mail, nebo potřebujete okamžitou odezvu, můžete komunikovat například prostřednictvím tzv. ***Instant Messaging*** nebo **chatu**. Tento typ komunikace obvykle nabízí také možnost přenosu souborů, což u běžného telefonního hovoru, kde mají účastníci také okamžitou odezvu, chybí.

Princip Instant Messagingu spočívá v tom, že někde existuje server, který komunikaci mezi dvěma protějšky zprostředkovává. Komunikující osoby se k němu přihlašují pomocí tzv. klienta. Příkladem Instant Messagingu může být v ČR populární **ICQ** nebo **Skype**.

Tyto služby často nabízejí kromě textových zpráv také možnost audio nebo video přenosu a přenosu souborů.



Bity a byty

Protože celou síťovou problematikou se prolínají pojmy bit a byte, vysvětlíme si je.

Bit je základní jednotkou informace. Má dva stavy, znázorňované obvykle číslicemi **1** a **0**. Čte se stejně, jako se píše – **bit**. Značí se **b**.

Z osmi bitů vzniká vyšší jednotka – **byte**, čte se bajt. Značí se **B**.

Protože se tato jednotka skládá z osmi bitů (osmi dvoustavových políček), je možné na těchto osmi bitech získat různými variacemi stavů 256 různých variací ($256 = 2^8$). Například 10101010, 11111111 nebo 01010001 atd. Dekadicky se jedná o čísla **0–255**, hexadecimálně o **00–FF**, binárně o **00000000–11111111**.

Běžně se pracuje s násobky této jednotky. Nejedná se zde o násobek tisíce, ale čísla 1 024.

Kilobyte – kB: **1 kB = 1 024 B**

Megabyte – MB: **1 MB = 1 024 kB**

Gigabyte – GB: **1 GB = 1 024 MB** atd.

Tímto způsobem se běžně označují násobky základní jednotky, ačkoliv to není úplně v souladu s normami. Podle norem by násobky **kilo**, **mega**, **giga** atd. měly být vždy tisícinásobky menší jednotky. Tyto 1024násobky jsou podle normy nazvány **kibi**, **mebi**, **gibi**, **tebi**.

Správně by tedy mělo platit, že **1 KiB** (kibibyte) **= 1 024 B**, **1 MiB** (mebibyte) **= 1 024 KiB**, **1 GiB** (gibibyte) **= 1 024 MiB** atd.

Obvykle se ale používají označení **kB** (kilobyte), **MB** (megabyte), **GB** (gigabyte) pro 1024násobky menší jednotky, i když to není v souladu s normami.

Přenosová rychlost

K vyjádření, jakou rychlostí se mohou data na přenosové médium vysílat, slouží jednotka „bity za sekundu“ (angl. ***bits per second***). Zkratka této jednotky je **bps** (***bits per second***) nebo **b/s** (bit za sekundu). Běžně se používají násobky této jednotky – **Mbps**, **kbps**, **Gbps**.

Můžete se setkat také s jednotkami **KiB/s** (kibibyty za sekundu), **MiB/s** (mebibyty za sekundu) atd.

2. Přenosová média

Kroucená dvojlinka

Kroucená dvojlinka (angl. ***Twisted Pair***) je druh kabelu, který se zpočátku používal v telekomunikacích, ale postupem času se prosadil i v přenosu dat po lokální síti.

Tento typ kabelu sestává z párů vodičů, kdy každý vodič je obalen plastovým obalem a každý pár vodičů je stočen dohromady. Nakonec jsou dohromady stočené i všechny páry. Tímto zkroucením dostává kabel lepší přenosové vlastnosti, je to i určitá ochrana proti rušení přenosu a vytváření přeslechů na kabelu.

Dva základní typy kroucené dvojlinky jsou **stíněná kroucená dvojlinka** a **nestíněná kroucená dvojlinka**.

Stíněná kroucená dvojlinka – STP

STP – angl. ***Shielded Twisted Pair***

Je to typ kabelu, který se skládá ze čtyř párů stočených vodičů, kde každý pár je obalen kovovou fólií a pak jsou ještě všechny páry dohromady obaleny rovněž kovovou fólií.

Díky tomuto stínění je omezeno jak vyzařování elektromagnetického záření ven, tak rušení elektromagnetickým polem z vnějšího prostoru.

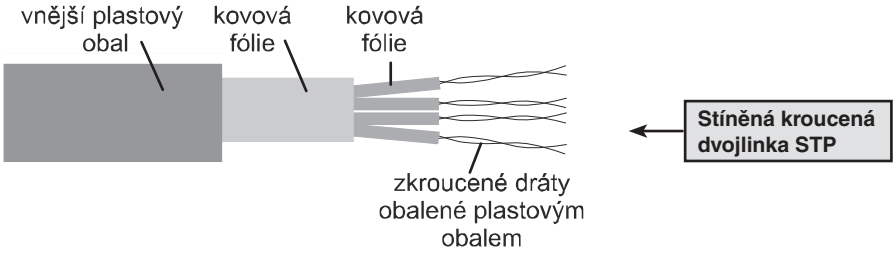
Prodejci tento kabel někdy označují jako **FTP** (***Full-shielded Twisted Pair***), což může terminologicky mást, neboť jiný kabel s částečnou metalickou ochranou zvaný **ScTP** (***Screened Twisted Pair***) se také občas označuje **FTP** (***Foil Twisted Pair***).

Ukončení kabelu **STP** je náročnější než u nestíněné kroucené dvojlinky – kovové stínění musí být v koncovce správně uzemněno, jinak by kabel se špatně vyrobenou nebo chybně uzemněnou koncovkou fungoval naopak jako anténa a byl by velmi náchylný k rušení.

Tímto kabelem je možné vést datový přenos o rychlosti 10 až 100 Mbps.

Délka kabelu je maximálně 100 metrů.

Ve srovnání s nestíněnou kroucenou dvojlinkou je dražší.



Stíněná kroucená dvojlinka STP

Částečně stíněná kroucená dvojlinka – ScTP

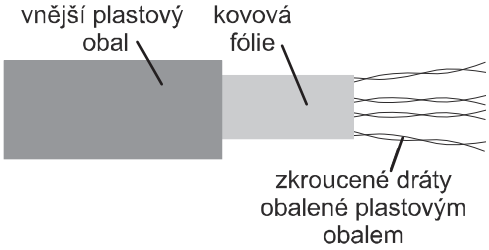
ScTP – angl. ***Screened Twisted Pair***

Na rozdíl od předchozího typu kabelu **STP** má **ScTP** stínění jen vnější kovovou fólii obalující všechny čtyři zkroucené páry.

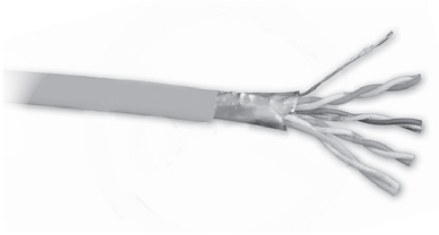
Díky úspornějšímu stínění má menší průměr a nižší hmotnost.

Tímto kabelem je možné vést datový přenos o rychlosti 10 až 100 Mbps.

Délka kabelu je maximálně 100 metrů.



Částečně stíněná kroucená dvojlinka ScTP

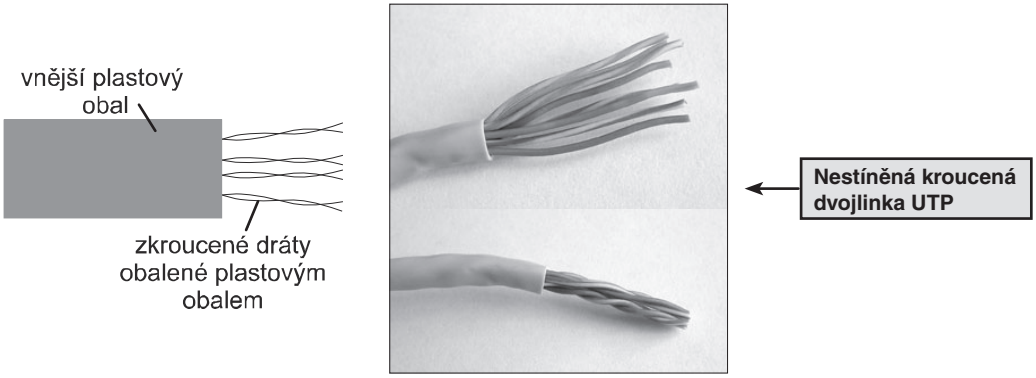


Částečně stíněná kroucená dvojlinka ScTP

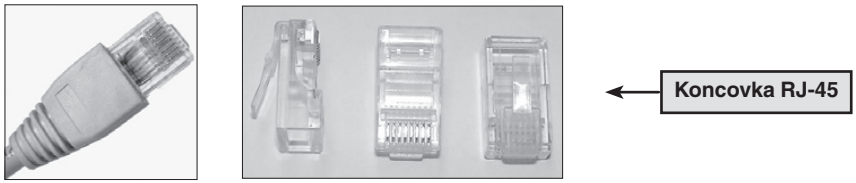
Nestíněná kroucená dvojlinka – UTP

UTP – angl. ***Unshielded Twisted Pair***

UTP je nestíněný kabel sestávající ze čtyř párů vodičů. Každý vodič je obalen plastovým obalem, každý pár je stočen dohromady a okolo všech je plastový obal.



Na konec **UTP** kabelu se instaluje koncovka **RJ-45** (**RJ** – ***registered jack***).



Tento kabel nepotřebuje uzemnění, neboť neobsahuje kovové stínění jako **STP** a **ScTP**. Má menší průměr, je lehčí, levnější a nainstalovat konektor je snadné. *Instalace koncovky bude ukázána později.*

Nevýhodou je větší náchylnost k rušení a větší vyzařování do okolí.

Tímto kabelem je možné vést datový přenos o rychlosti 10 Mb/s až 1 Gb/s, u kabelu kategorie 3–6. Délka kabelu je maximálně 100 metrů. Je to běžně využívaný typ kabelu v lokálních sítích.

Kategorie	Využití
1	Telefonní rozvody, nikoliv přenos dat
2	Do rychlosti 4 Mb/s bylo možno přenášet i data
3	Přenos dat do rychlosti 10 Mb/s na lokálních sítích s technologií **Ethernet**
4	Přenos dat do rychlosti 16 Mb/s na sítích využívajících technologii **TokenRing**
5	Přenos dat do rychlosti 100 Mb/s v sítích využívajících technologii **Ethernet**

5e	Přenos dat do rychlosti 1 Gb/s v sítích využívajících technologii **Ethernet**, využívá se všech 8 vodičů v kabelu
6	Přenos dat do rychlosti 1 Gb/s v sítích využívajících technologii **Ethernet**, využívá se všech 8 vodičů v kabelu

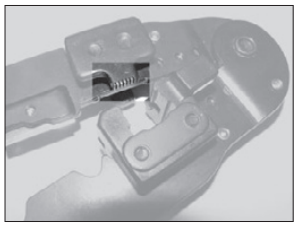
S rostoucí kategorií roste i kvalita kabelu.

Pokud budete chtít navrhovat síť pro určitou přenosovou rychlost, je potřeba mít na zřeteli kvalitu kabelu. Také je vhodné při instalaci nové kabeláže myslet na budoucí vývoj a volit raději kvalitnější typ kabelu.

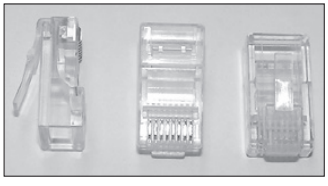
Instalace koncovky RJ-45

Pomůcky pro výrobu

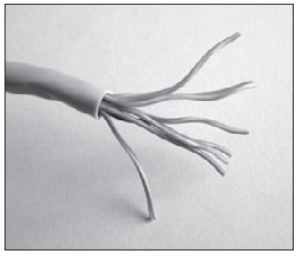
Krimpovací kleště



Detail kleští

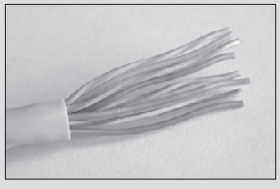


Koncovka RJ-45

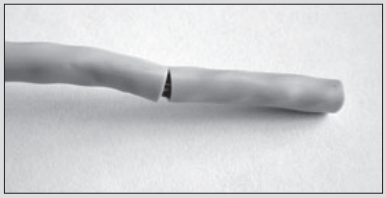


UTP kabel

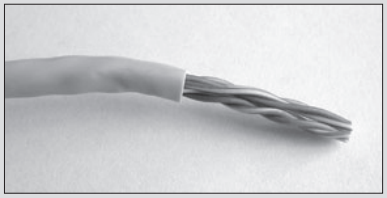
Při instalaci koncovky máte možnost výběru ze dvou základních **typů A** a **B**.

Koncovka typu A – pořadí vodičů	**Koncovka typu B – pořadí vodičů**
Bílo-zelená, zelená, bílo-oranžová, modrá, bílo-modrá, oranžová, bílo-hnědá, hnědá 	Bílo-oranžová, oranžová, bílo-zelená, modrá, bílo-modrá, zelená, bílo-hnědá, hnědá 

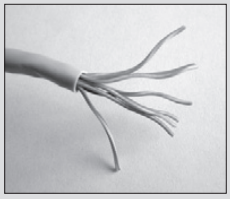
Postup instalace koncovky



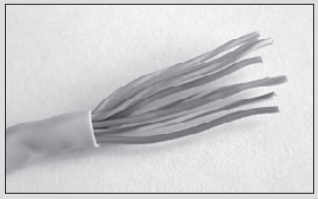
Opatrně nařízněte vnější plastový obal UTP, aby nedošlo k prořízntutí plastových obalů jednotlivých vodičů.



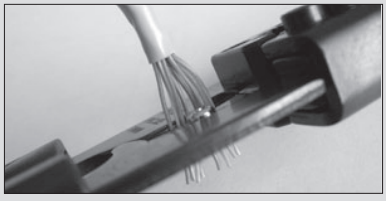
Odstraňte obal UTP na konci kabelu.



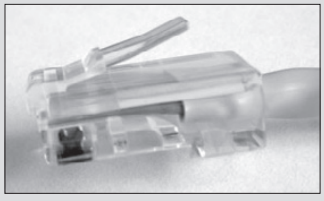
Jednotlivé vodiče rozmotejte.



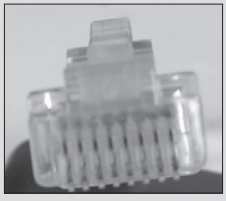
Srovnejte vodiče tak, aby odpovídaly zvolenému typu koncovky – zde je zvolen typ **B**.



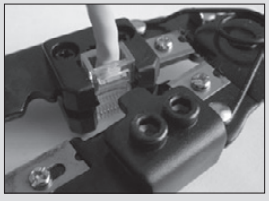
Pomocí krimpovacích kleští odstříhněte konce vodičů, srovnejte je do roviny a ponechte cca 1,5 cm pro zastrčení do koncovky.



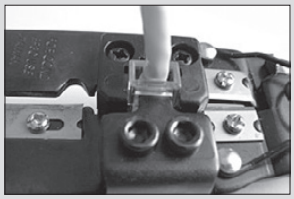
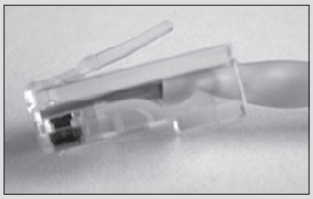
Srovnané a zastřižené vodiče zasuňte do koncovky až na konec.



Na konci koncovky musí být vidět, že jsou vodiče zastrčeny až do konce, jinak by byl problém s přenosem signálu.



Koncovku vsuňte do otvoru v krimpovacích kleštích.

	
Kleště pevně sevřete, čímž dojde k zaříznutí kovových nožíků koncovky do vodičů a plastového zobáčku koncovky do obalu kabelu.	Při pohledu z boku je vidět, že plastový zobáček koncovky svírá obal kabelu a kovové nožíky jsou zaříznuté do vodičů.

Funkčnost kabelu můžete ověřit kabelovým testerem.

Kabelový tester

Typy UTP kabelů

Pokud má kabel obě koncovky stejné (například obě **typu A**), jedná se o kabel **přímý**. Používá se ke spojení počítače a přepínače, počítače a rozbočovače, směrovače a přepínače, směrovače a rozbočovače, směrovače a mostu (***bridge***), počítače a mostu (***bridge***).

Pokud je jedna koncovka **typu A** a druhá **typu B**, jedná se o kabel **křížený**. Používá se ke spojení zařízení stejného typu – počítače a počítače, přepínače a přepínače, ethernetového portu směrovače a počítače, rozbočovače a rozbočovače, směrovače a směrovače (přes ethernetové zásuvky).

Některá zařízení mají schopnost autodetekce připojeného kabelu a dokážou vnitřně provést přepnutí na potřebný typ.

Prodloužení kabelu UTP

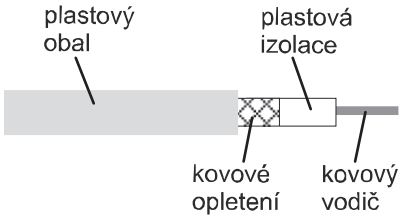
Kabel lze prodloužit pomocí pasivního zařízení – **spojky**. Tato spojka signál nijak neregeneruje, naopak zde může docházet k dalšímu útlumu, proto je potřeba její použití zvážit. Kabel UTP má maximální doporučený dosah 100 metrů (bez spojky nebo s ní).

Spojka

Koaxiální kabel

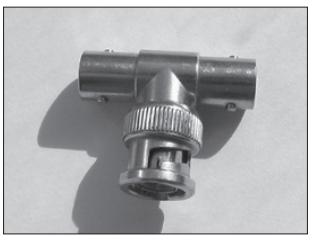
Koaxiální kabel se skládá ze dvou vodičů oddělených plastovou izolací. Jeden vodič tvoří jádro a je ve formě drátu, druhý vodič oddělený od jádra plastovou izolací je kovový obal. Vše je chráněno vnějším plastovým obalem.

Ukončení kabelu se děje pomocí **BNC konektoru** (název **BNC** vznikl podle bajonetového principu konektoru a jeho vynálezců Neilla a Concelmana) nebo ukončovacího článku (**terminátor**) o stejné impedanci jako kabel – 50 Ω. Pro vytvoření odbočky z koaxiálního kabelu k počítači se používá **T-konektor**.

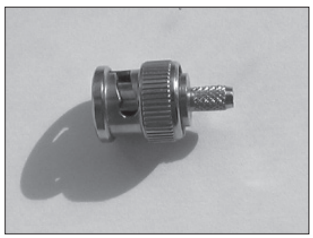


Koaxiální kabel – schéma

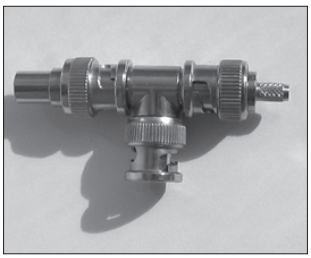
Ukončení – terminátor



T – konektor



BNC – konektor



Kovový obal funguje jako druhý vodič a navíc také jako stínění chránící vnitřní vodič proti vnějšímu rušení.

V lokálních sítích se koaxiální kabel používal hlavně dříve. Počítače se ke společnému koaxiálnímu kabelu připojovaly pomocí přípojek do tzv. **sběrnicové topologie**.

Protože se jedná o sdílené přenosové médium, musely se počítače vypořádat s častými kolizemi na síti, které dnes prakticky odbourává použití přepínačů *(přepínače jsou popsány v následující kapitole).*

Typy koaxiálních kabelů

- **Silný** – průměr je cca 1 cm

  Bez regenerace signálu je schopen přenášet data až na vzdálenost 500 metrů, pak musí být připojen opakovač (repeater), který signál zregeneruje. Používá se pro rychlost přenosu 10 Mbps.

- **Tenký** – průměr je cca 0,35 cm

  Bez regenerace signálu je schopen přenášet data až na vzdálenost 185 metrů, pak musí být připojen opakovač (repeater), který signál zregeneruje. Používá se pro rychlost přenosu 10 Mbps.

Optické kabely

Optické kabely nabízejí schopnost přenášet velká množství dat za jednotku času, proto jsou voleny pro velmi zatěžované linky v sítích **LAN** (lokální síť) nebo **WAN** (rozlehlá síť).

Mimoto signál vedený optickými kabely není ovlivňován elektromagnetickým rušením z vnějšího prostoru a také žádné takové rušení negeneruje.

Nedochází zde k tak vysokému útlumu, proto je možné přenášet data na podstatně větší vzdálenosti než u metalických rozvodů. Vzdálenost, kterou data urazí bez nutnosti regenerace, se počítá v řádech kilometrů (cca 2–10 km).

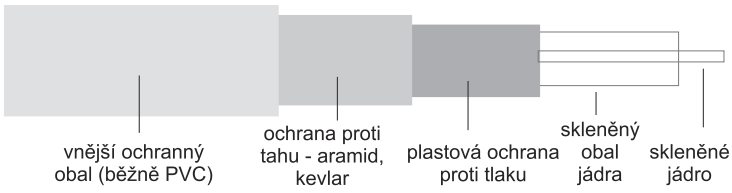
Pro narušitele není snadné přenášený signál zachytit a odposlechnout, na rozdíl od bezdrátových sítí.

Data v optických kabelech jsou přenášena pomocí světelných impulzů v infračerveném spektru. Jedničky a nuly vysílaného signálu jsou reprezentovány světelným impulzem a pauzou. Toto světlo není pro lidské oko viditelné, ale pokud by došlo ke kontaktu s očima, způsobilo by vážné poškození zraku, proto **je nebezpečné dívat se přímo do ukončení aktivního optického vlákna.**

Vlnové délky světla v optických kabelech jsou 850 nm, 1 310 nm, 1 550 nm. Tyto vlnové délky mají lepší přenosové schopnosti než jiné vlnové délky.

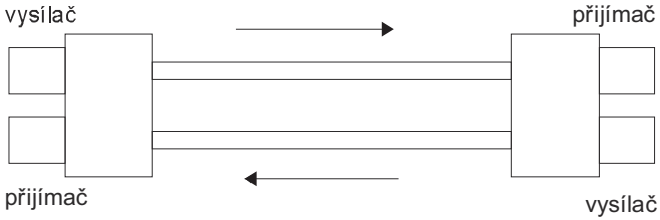
Optické vlákno se skládá z několika vrstev. Nejdůležitější je jádro, které je skleněné nebo plastové, kolem něj je opět skleněná nebo plastová vrstva s nižší optickou hustotou, než je optická hustota jádra. Když světelný paprsek cestující jádrem narazí na rozhraní těchto dvou skleněných vrstev o různé optické hustotě, dojde k absolutnímu odrazu zpět do vlákna. Tato vlastnost je způsobena tím, že světlo je posláno do vlákna pod určitým úhlem, který nesmí být příliš velký (vzhledem k podélné ose vlákna), protože jinak by docházelo nejen k odrazu do vlákna, ale i k lomu paprsku ven z vlákna, a tím k zeslabení signálu.

Okolo těchto dvou vnitřních vrstev je několik obalů, které je mají chránit před nárazy, zlomením, tahem, odřením, rozpouštědly, ohněm nebo znečištěním.



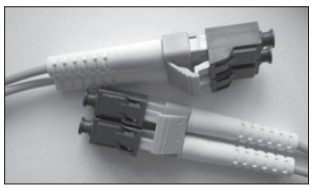
Optické vlákno – schéma

Kabel se skládá ze dvou vláken, jedním se data vysílají a druhým přijímají. Tato dvě vlákna jsou uvnitř jednoho obalu.

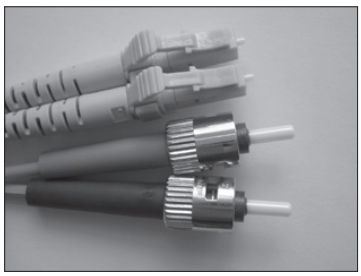


Vláknо pro příjem a vysílání

Často se v jednom obalu vyskytuje mnoho párů optických vláken, protože nenastává problém se vzájemným rušením. Nedochází k žádnému úniku signálu z vlákna, který by mohl narušovat signál v ostatních vláknech.



Kabel z optických vláken s konektory



Do optického kabelu se světelné impulzy posílají laserem nebo infračervenou LED diodou kolmo nebo pod určitým úhlem, na jehož velikosti závisí, zda se paprsek celý odrazí zpět do optického vlákna ve chvíli, kdy narazí na rozhraní dvou skleněných vrstev, nebo zda dojde k odrazu jen částečnému a část paprsku unikne pod úhlem lomu ven z vlákna. Samozřejmě se volí takové úhly svícení do optického vlákna, které zajistí absolutní odraz zpět do vlákna. Jinak by docházelo k degradaci světelného signálu a ztrátě síly.

Ve skleněných (křemíkových) vláknech je útlum signálu menší než v plastových vláknech.

K **útlumu** dochází vlivem přeměny světla na teplo při kontaktu s materiálem vlákna, rozptylem světelných paprsků v materiálu vlákna, ohybem, kdy dochází k lomu paprsku ven z vlákna, a v neposlední řadě vlivem špatného napojení vlákna v místech spojení a na konektorech.

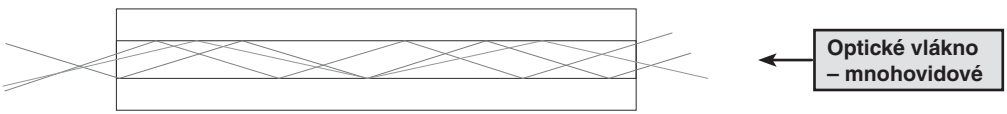
Typy optických vláken

- **Jednovidové vlákno** (angl. ***single mode fiber***) – do vlákna se data posílají paprskem vysílaným souběžně s podélnou osou vlákna. V případě ohybu vlákna dochází k odrazu paprsku zpět do jádra. Cesta paprsku je minimální, dochází k minimálnímu útlumu signálu, a proto je možné signál šířit na velkou vzdálenost – ve srovnání s mnohovidovými vlákny, u kterých je signál posílán do vlákna více paprsky různými směry. Průměr jádra je malý, do 10 mikrometrů.

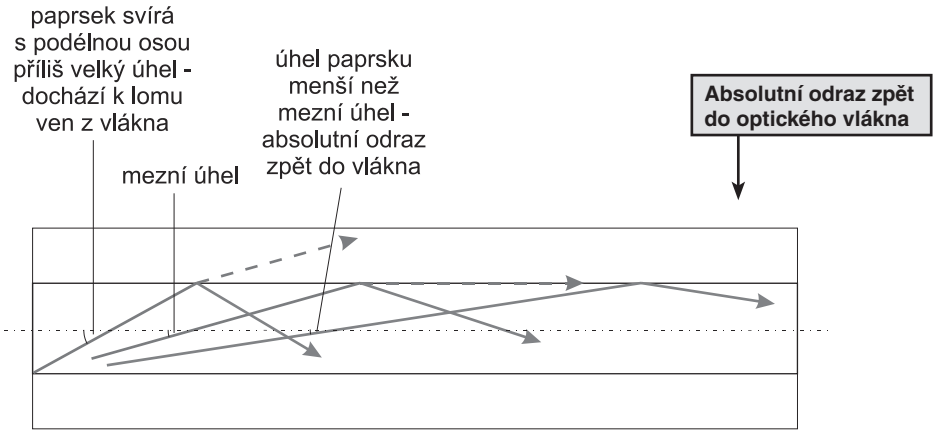
  Označení vlákna **9/125** znamená, že jádro má v průměru 9 mikrometrů a jeho skleněný obal má průměr 125 mikrometrů.

- **Mnohovidové vlákno** (angl. ***multi mode fiber***) – do vlákna se data posílají paprsky vysílanými v určitém omezujícím kuželu, který zajistí absolutní odraz zpět do vlákna při dopadu na rozhraní vrstev jádra a obalu. Průměr jádra je větší než 10 mikrometrů.

  Označení vlákna **62,5/125** nebo **50/125** znamená, že jádro má v průměru 62,5 nebo 50 mikrometrů a jeho skleněný obal má průměr 125 mikrometrů.



- **Gradientní vlákno** – je to varianta mnohovidového vlákna. Optická hustota jádra se směrem ke skleněnému obalu postupně snižuje, což má za následek rychlejší pohyb světelného paprsku směrem dál od středu a pomalejší průchod, pokud paprsek cestuje přímo středem, a také jeho postupné ohýbání zpět do jádra. Paprsek putuje po křivce podobné sinusoidě. Výhodou je, že paprsky jdoucí vláknem pod různými úhly dorazí na konec vlákna přibližně ve stejnou chvíli, bez ohledu na délku trasy, kterou musely urazit.



Optická vlákna jsou schopna přenášet data rychlostí více než 1 Gb/s na vzdálenosti až 10 km v případě jednovidového a 2 km v případě mnohovidového vlákna (v závislosti na síle vysílaného signálu).

Světelné zdroje – vysílač

Vysílač přijme data v elektronické formě a přetransformuje je do podoby světelných signálů, které následně v podobě světelných impulzů vyzáří.

Infračervená LED dioda používá infračervené světlo na vlnové délce 850 a 1 310 nanometrů. Využití nachází v mnohovidových vláknech **LAN** sítí.

Laser vyzařuje infračervené světlo většinou na vlnové délce 1 310 nebo 1 550 nanometrů. Vysílá tenký paprsek vysoké intenzity do optického vlákna. Používá se pro jednovidové vlákno. Využití nachází ve **WAN** sítích nebo v hlavních rozvodech sítě.

Přijímač světelného signálu

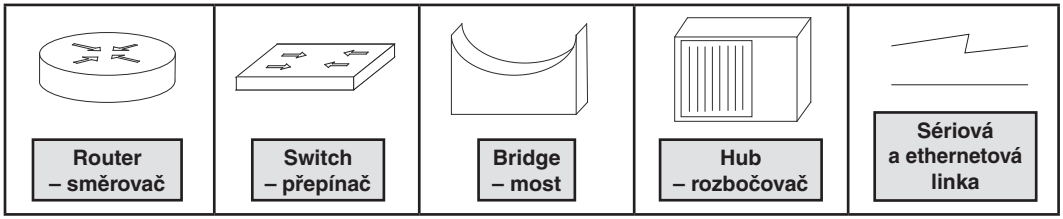
Na konci optického kabelu naráží světelné impulzy do fotoelektrické diody, která následně produkuje elektrický signál. Ten je možné již vést metalickými rozvody do zařízení, jako jsou počítače, směrovače, přepínače.

3. Síťová zařízení

Symboly používané pro síťové prvky

Nejsou standardizované, podobají se prvkům, které znázorňují.

V následující tabulce jsou uvedeny síťové prvky použité v knize.



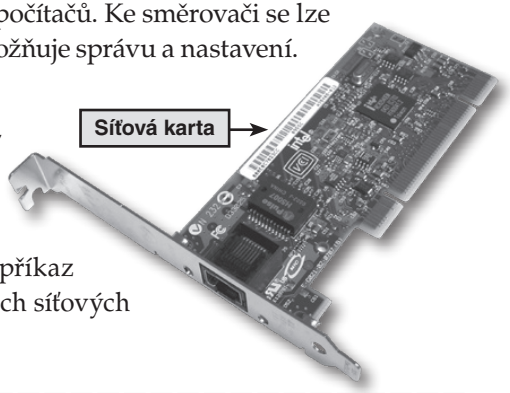
Síťová karta

Koncová zařízení jsou připojena do sítě pomocí **síťové karty** (**NIC** – ***Network Interface Card***). Té je již od výrobce přiřazena unikátní fyzická adresa, tzv. **MAC adresa**. Někdy je možné ji softwarově změnit, nicméně v jedné lokální síti musí mít každé koncové zařízení nastaveno jinou MAC adresu, jinak by v síti docházelo k problémům s adresací a přenosem.

MAC adresa (zkratka angl. ***Media Access Control***) je 48bitová adresa zapisovaná většinou jako šest hexadecimálních dvouciferných čísel oddělených pomlčkami nebo dvojtečkami – například **00-A2-B1-52-63-4E**.

Síťová karta je zařízení pracující na druhé vrstvě OSI modelu (*bude probrán detailně v dalších kapitolách*), pomocí MAC adresy dokáže komunikovat s ostatními počítači v lokální síti.

Běžně lze MAC adresu měnit na směrovači, kdy rozhraní směřujícímu do vnější sítě je přiřazena MAC adresa některého počítače umístěného v lokální síti. Pro venek se pak směrovač jeví z hlediska MAC adresy jako jeden z počítačů. Ke směrovači se lze často připojit pomocí webového rozhraní, které umožňuje správu a nastavení. *Více o směrovači později.*



Zjistit MAC adresu v počítači můžete několika způsoby, jeden je poměrně univerzální. V operačním systému Windows spusťte příkazový řádek (například v nabídce **Start** zvolte položku **Spustit** a zapište příkaz **cmd**). V příkazovém řádku zadejte příkaz **ipconfig /all**, následně se zobrazí nastavení a stav všech síťových připojení.

Na obrázku má počítač fyzickou MAC adresu 00-21-85-E2-71-4B.

Repeater – opakovač

Toto zařízení pracuje na první vrstvě **OSI modelu** (*OSI model bude probrán detailně v dalších kapitolách*). Upravuje elektrický nebo optický signál, který jím prochází, opravuje časování, sílu a kvalitu a v této upravené a zesílené podobě jej vysílá dále.

Obvykle obsahuje dva porty, jedním signál přijímá, druhým v upravené podobě vysílá.

Pomocí opakovače můžete prodloužit délku kabelu. Je to aktivní síťový prvek, který regeneruje signál na úrovni bitů. Regenerace signálu je po určité délce kabelu nutná, neboť dochází k útlumu, šumu a ztrátám. Přílišné prodloužení kabelu na lokálních sítích však není žádoucí, neboť by se tím příliš zvětšovala kolizní doména, a počítače v této kolizní doméně by nebyly schopny včas rozpoznat nastalou kolizi signálů kvůli velkému zpoždění na kabelu. Kolize signálu jsou v lokálních sítích běžné a jejich rozpoznání je nutností.

Kolize a kolizní doména budou probrány později.

Hub – rozbočovač

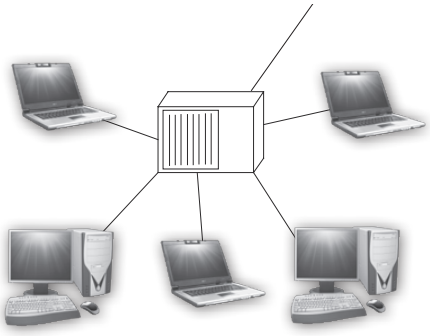
Rozbočovač je ve své podstatě multiportní opakovač, má také za úkol regenerovat a opravovat příchozí signál. Rozbočovač přijme signál jedním portem a zesílený a opravený jej pošle všemi ostatními porty. Pracuje stejně jako opakovač na první vrstvě OSI modelu, na úrovni bitů.

Rozbočovače jsou v dnešní době nahrazovány přepínači, a to z toho důvodu, že rozbočovač posílá přijatý signál všemi ostatními porty, na rozdíl od přepínače, který si po určité době vybuduje znalost o připojených síťových zařízeních a přijatý signál posílá jen cílovému zařízení příslušným portem. Proto zbytečně nezatěžuje síť.

Tím, že rozbočovač posílá přijatý signál všemi ostatními porty, zabrání ve vysílání všem ostatním připojeným počítačům, které musí v danou chvíli počkat. Jinak by došlo ke kolizi signálů. Kolize je na lokálních sítích běžným jevem. Síťová zařízení ji umějí detekovat a následně čekají, než se znovu pokusí vysílat. Všechny počítače připojené na hub vytvářejí tzv. **kolizní doménu**.

Rozbočovače se využívaly spíše dříve na lokálních sítích s rychlostí přenosu 10 nebo 100 Mb/s.

Hub – rozbočovač



Hub – schéma zapojení do topologie typu *hvězda*

Bridge – most

Bridge je zařízení pracující na druhé vrstvě OSI modelu. Je v síti umístěn tak, aby odděloval dvě nebo více skupin počítačů. Na začátku se chová jako opakovač nebo rozbočovač, neboť informace o umístění počítačů se teprve učí z provozu na síti. Během času se naučí, na kterém portu má připojeny jaké počítače. Ke svým portům si přiřadí MAC adresy počítačů v daném segmentu. Vytváří a udržuje tabulku MAC adres.

Jakmile tyto informace získá, dokáže na základě MAC adresy rozhodnout, kam signál pouštět, a tím provoz filtruje. Rozhodování provádí softwarově, na rozdíl od přepínače, který provádí přepínání hardwarově.

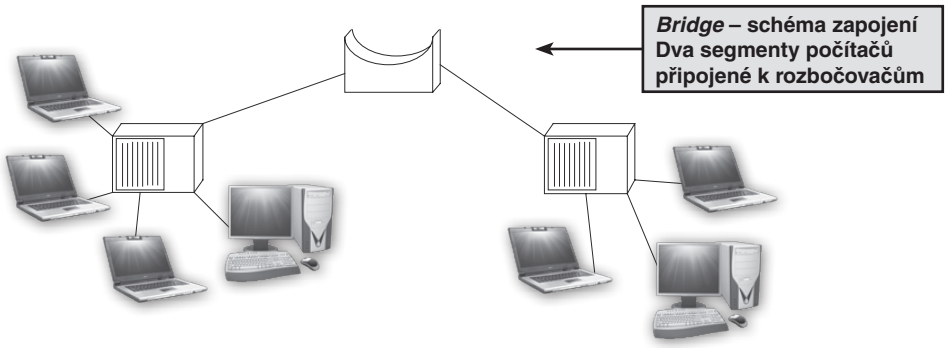
Pokud má příchozí signál cíl ve stejném segmentu, ve kterém leží zdrojový počítač, neposílá tento signál nikam dále. Je totiž zřejmé, že cílový počítač již musel vysílaný signál přijmout, neboť je ve stejném segmentu jako zdrojový počítač. Oba jsou součástí jedné kolizní domény.

Pokud je cílový počítač na jiném segmentu než zdrojový počítač, bridge pošle data do tohoto druhého segmentu.

Pokud cílový počítač obsahuje neznámou MAC adresu, pak bridge pošle data všemi porty s výjimkou příchozího portu.

Takto zachází s daty, která jsou určena pro jednoho konkrétního příjemce (vysílání typu **unicast**). Pokud se jedná o data, která jsou určena více počítačům (vysílání typu **broadcast** nebo **multicast**), chová se jako rozbočovač a filtrování neprovádí.

Broadcast je typ vysílání určený všem počítačům v síti. Proto jej bridge posílá všem počítačům v síti. Bridge je tedy zařízení, které dokáže oddělovat kolizní domény pomocí filtrování provozu, ale všechny připojené segmenty jsou součástí jedné broadcast domény.



***Bridge* – schéma zapojení**
Dva segmenty počítačů připojené k rozbočovačům

Switch – přepínač

Přepínač je zařízení pracující na druhé vrstvě OSI modelu. Dokáže dělat rozhodnutí na základě MAC adresy.

Je to zařízení obsahující mnoho portů. Funguje podobně jako bridge, jen přepínání se děje obvykle hardwarovou cestou, výkonněji a rychleji. Nahrazuje dřívější využití rozbočovačů, dělí síť na jednotlivé kolizní domény. Ke každému portu je připojena jedna kolizní doména. Kolize na jednom segmentu neomezují provoz na ostatních segmentech.

K přepínači jsou připojeny buď přímo koncové počítače, nebo celé segmenty počítačů připojených k jinému přepínači nebo rozbočovači.

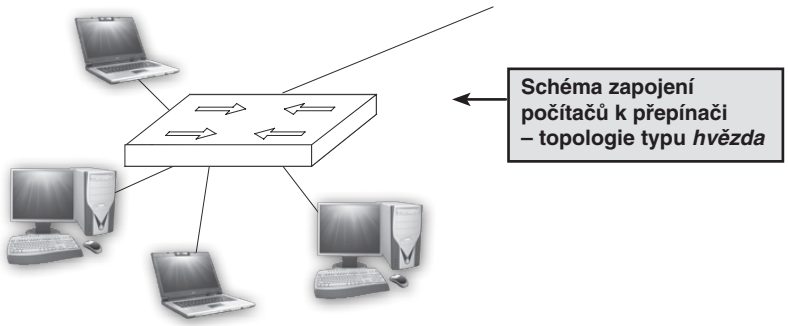


Schéma zapojení počítačů k přepínači – topologie typu *hvězda*

Na začátku se přepínač chová jako rozbočovač. Během provozu na síti se naučí, ke kterému portu jsou připojeny jaké počítače, a vede si tabulku jejich MAC adres.

Pokud dostane data směrovaná k počítači, jehož MAC adresu zatím nemá ve své tabulce MAC adres, pošle tato data všemi ostatními porty jako rozbočovač. Je pravděpodobné, že cílový počítač takto zaslaná data přijme a přijetí potvrdí, a z této odpovědi se přepínač naučí, na kterém portu cílový počítač leží. V dalším vysílání již umí počítač identifikovat a poslat data jen příslušným portem.

Přepínač filtruje provoz na základě cílové MAC adresy v případě vysílání typu unicast určeného jen jednomu cílovému zařízení. V případě vysílání typu multicast nebo broadcast – vysílání pro více cílových zařízení – se chová jako rozbočovač, vysílá data všemi ostatními porty s výjimkou příchozího portu.

Je tedy zřejmé, že nefiltruje vysílání typu broadcast. Všechny počítače a zařízení připojená na přepínač jsou součástí jedné broadcast domény.

Některé přepínače mají schopnost pracovat i na třetí nebo dokonce čtvrté vrstvě OSI modelu, dokážou rozhodovat na základě nejen MAC adresy, ale i IP adresy – pak fungují jako směrovač, nebo také umí zpracovávat data i na základě čísel portů, které určují výslednou aplikaci, do níž data směřují. Nicméně základní funkcí přepínače je pracovat v lokální síti a zpracovávat provoz na základě MAC adres.

Je několik režimů, ve kterých může přepínač data zpracovávat. Jedním z nich je způsob zvaný **store and forward**, kdy přepínač přijme celý datový rámec (tak se nazývá část dat, se kterou přepínač pracuje), zkontroluje kromě zdrojové a cílové MAC adresy také kontrolní součet na konci rámce, tím zajistí, že data jsou neporušená, a pošle je dál. Tak předchází odesílání chybných rámců, avšak má to za následek určité zpoždění.

Jiná metoda zpracování datových rámců je metoda zvaná **cut through**, kdy přepínač odesílá přijatá data ihned, jakmile přijme dostatečnou část rámce, ze které zjistí MAC adresu cílového zařízení. Tato metoda je ve srovnání s předchozí rychlejší, ale může docházet k přeposílání poškozených dat.

Určitým kompromisem je metoda **fragment free**, kdy přepínač počká s odvysíláním přijatého datového rámce do té doby, než přijme prvních 64 bytů, čímž by mělo být omezeno vysílání dat poškozených určitým druhem kolizí v síti.

← ***Switch*** **– přepínač**

Router – směrovač

Směrovač je zařízení pracující na třetí vrstvě OSI modelu. Dělá rozhodnutí, kam data poslat, na základě síťové adresy cílového zařízení. Odděluje různé sítě.

Směrovač si vytváří tabulku s nejlepšími cestami do jemu známých sítí, tzv. **routovací (směrovací) tabulku.** V tabulce je cestám přiřazena určitá hodnota závislá na metrice, s jakou směrovač cesty posuzuje. Podle těchto hodnot pak směrovač vyhodnocuje, která cesta do cílové sítě je nejvýhodnější. Informace o tom, kde která síť leží, získává z okolních směrovačů, které mu informace poskytují v tzv. **směrovacích aktualizacích** (**routovacích updatech**).

V příchozím datovém paketu (to je datová jednotka, se kterou směrovač pracuje) si přečte síťovou adresu cílového zařízení a podle své směrovací tabulky rozhodne, na které rozhraní data přepne a pošle.

Pokud cílovou adresu nezná, posílá data rozhraním, které má přednastaveno pro tento účel.

Na rozdíl od přepínače (na jehož portech jsou připojena zařízení se stejnou adresou sítě a přepínání se děje na základě MAC adresy) se porty směrovače musí vyskytovat v různých sítích (jiné adresy sítí).

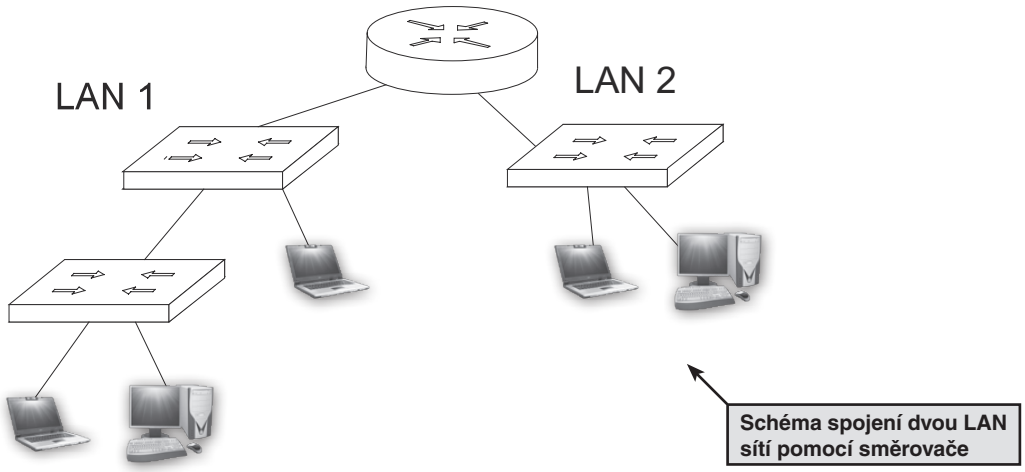


Schéma spojení dvou LAN sítí pomocí směrovače

Pro směrování se používá ve většině případů **protokol IP** (***Internet Protocol***). Tento protokol nezajišťuje spolehlivost doručení ani doručení datových paketů ve správném pořadí, to zaručuje protokol vyšší vrstvy (**TCP** – angl. ***Transmition Control Protocol***), který má tuto kontrolu na starosti.

Podrobně bude probráno později.

Příkladem síťové adresy IP je například **192.168.2.5**.

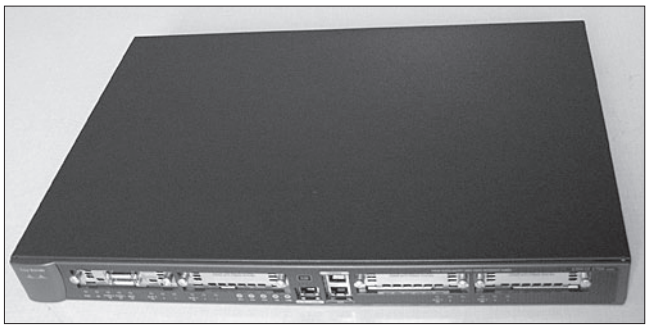
Síťovou adresu počítače můžete zjistit také například příkazem **ipconfig.**

```
Adaptér sítě Ethernet Připojení k místní síti:

   Přípona DNS podle připojení . . . :
   Adresa IPv4 . . . . . . . . . . . : 192.168.0.101
   Maska podsítě . . . . . . . . . . : 255.255.255.0
   Výchozí brána . . . . . . . . . . : 192.168.0.1
```

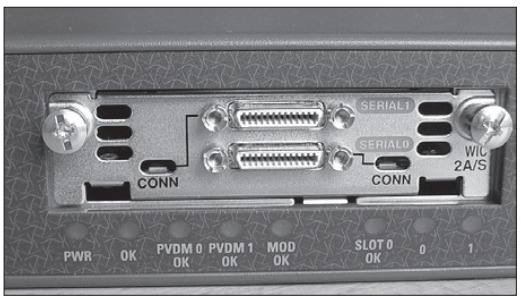
Výpis síťového nastavení příkazem *ipconfig*

Na obrázku má počítač síťovou IP adresu **192.168.0.101**.

Velmi podrobně bude o IP adresách, maskách, bránách pojednáno později.



Router – směrovač



Sériové rozhraní – pro spojení do WAN

Ethernetové, konzolové a AUX rozhraní

4. Typy sítí, Extranet, Intranet

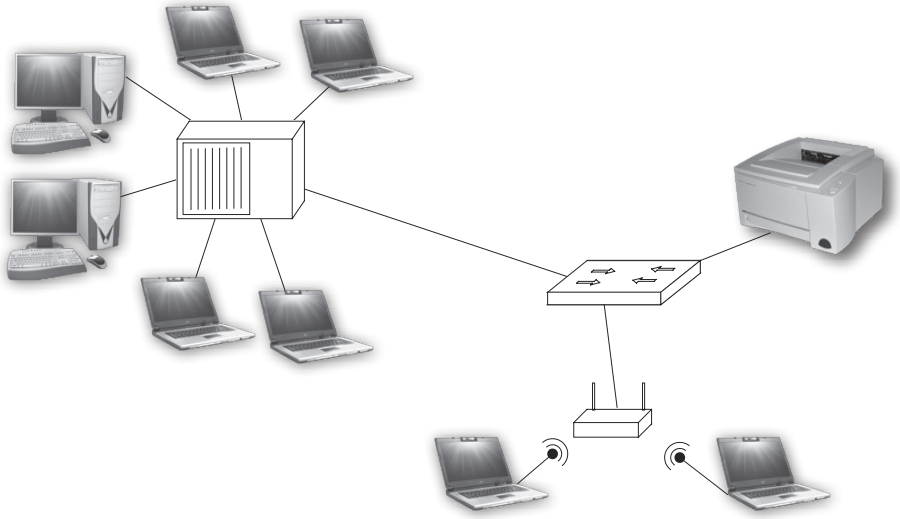
LAN

LAN (***Local Area Network***) – lokální síť

Tato síť se skládá z koncových zařízení, počítačů s jejich síťovými kartami, vstupních a výstupních zařízení, přenosových médií a dalších síťových zařízení, pomocí kterých jsou počítače a další koncová zařízení propojeny.

Umožňuje sdílení dokumentů, tiskáren a zprostředkovává lokální komunikaci.

Vyskytuje se na určitém vymezeném prostoru. Uživatelé této sítě mají umožněno vzájemné spojení většinou pomocí vysokorychlostních linek. V rámci LAN sítě jsou propojena fyzicky blízká zařízení. Přístup k lokálním službám je víceméně neustálý.



Technologie umožňující přenos dat po síti LAN jsou nejčastěji **Ethernet**, **Token Ring** a **FDDI**. Nejčastěji je používána technologie **Ethernet**. *Podrobněji bude probrána později.*

WAN

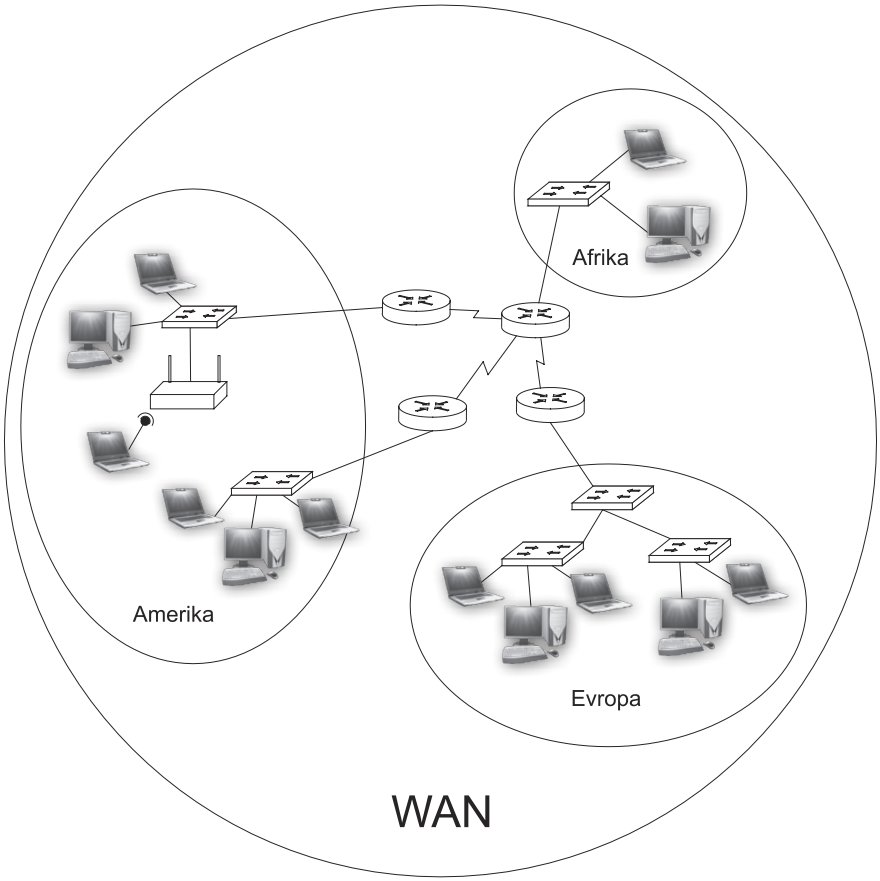
WAN (***Wide Area Network***) – rozlehlá síť

Tato síť spojuje jednotlivé lokální sítě, umožňuje uživatelům v těchto sítích vzájemnou komunikaci, mnohdy velmi vzdálenou. Propojené lokální sítě mohou být geograficky od sebe velmi daleko. Komunikace uživatelů napříč rozlehlou sítí probíhá v reálném

čase. Uživatelé mohou využívat vzájemně poskytované služby, jako například přístup k webovým stránkám nebo přenos souborů.

Technologie, které se často pro přenos dat v rozlehlých sítích používají, jsou **ISDN** (již poměrně zastaralé), **DSL** (má více variant), **Frame Relay**, **vytáčené spojení modemy**, linky typu **E1**, **E3** a další.

Příkladem WAN sítě je například síť **Internet**.

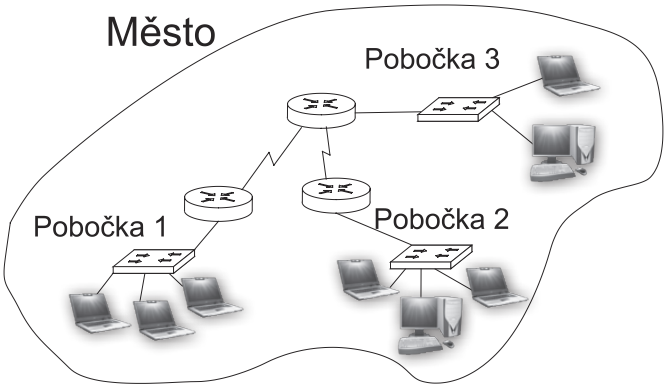


MAN

MAN (***Metropolitan Area Network***) – městská síť

Tato síť spojuje jednotlivé lokální sítě, které se nacházejí v geograficky blízké oblasti větší než LAN a menší než WAN, přibližně na území města.

Může spojovat například jednotlivé pobočky podniku na území města pomocí zabezpečených vyhrazených linek.



SAN

SAN (***Storage Area Network***) – síť úložišť

Je to síť specializovaná na přenos velkých množství dat. Data, která se přenášejí mezi jednotlivými úložišti a servery, nezatěžují jiné běžně používané linky.

Tato síť je zaměřená na výkon a dostupnost.

Intranet

Termínem **Intranet** můžeme označit webové služby dostupné oprávněným uživatelům v rámci LAN pouze z počítačů v této síti.

Extranet

Termínem **Extranet** můžeme označit webové služby podniku, které jsou dostupné i vnějším uživatelům, například obchodním partnerům. Uživatelé musí mít uživatelská jména a přístupová hesla.

5. Síťové modely

Důvodem, proč se k popisování dění na síti používají síťové modely, je snaha o zobecnění, nadhled a snadnější vysvětlení principů fungování zařízení. Pokud výrobci dodržují standardy vycházející ze síťových modelů, jejich zařízení jsou schopna spolupracovat.

Na síťových modelech lze snadno zobrazit a vysvětlit, jak síťová komunikace funguje.

Nejznámějšími modely jsou modely **ISO/OSI** a **TCP/IP**.

V praxi se ve vývoji internetu více využíval model TCP/IP. Pod záštitou modelu OSI se vyvinulo několik protokolů, které se velmi rozšířily. Ve vývoji nových protokolů určených pro nové sítě se pokračuje i dále.

OSI model je podrobnější a na jeho jednotlivých vrstvách lze dobře vysvětlit, jak protokoly fungují a jak vrstvy vzájemně spolupracují.

Síťový model ISO/OSI

Tento model byl vypracován mezinárodní organizací pro normalizaci ISO za účelem sjednocení a standardizace počítačových sítí a za účelem vypracování norem pro propojování různých systémů. V jednotlivých vrstvách sedmivrstvého OSI modelu se popisují funkce síťových zařízení a protokoly těchto vrstev.

Podrobně bude popsáno dále.

7. Aplikační vrstva
6. Prezentační vrstva
5. Relační vrstva
4. Transportní vrstva
3. Síťová vrstva
2. Spojová vrstva
1. Fyzická vrstva

Síťový model TCP/IP

Tento první síťový vrstvový model byl vytvořen v 70. letech 20. století. Obsahuje soubor komunikačních protokolů, které se používají na internetu a v ostatních sítích.

Jeho název pochází ze základních dvou protokolů typických pro tento model – **TCP** (***Transmition Control Protocol***) a **IP** (***Internet Protocol***).

Je to model čtyřvrstvý.

4. Aplikační vrstva
3. Transportní vrstva
2. Internetová vrstva
1. Vrstva síťového rozhraní

Aplikační vrstva zajišťuje koncové zobrazení dat uživateli spolu s kódováním.

Transportní vrstva zajišťuje komunikaci vzdálených zařízení napříč sítí a spolehlivý přenos dat.

Internetová vrstva zajišťuje nejlepší cestu dat k cíli.

Vrstva síťového rozhraní zajišťuje přístup dat na síť, kontroluje zařízení a síťová média na síti.

Proces úpravy dat pro přenos a jejich zpětná rekonstrukce

Zjednodušeně a zkráceně lze přenos dat po síti popsat následovně:

- Na počátku se vytvoří data určená pro přenos pomocí koncové aplikace.
- Data se rozdělí na části a zapouzdří, přidají se k nim další doprovodné informace, aby byla schopna dorazit do cílového zařízení a příslušné aplikace.
- Data připravená v podobě binárního kódu se odešlou síťovou kartou na přenosové médium, během cesty projdou přes další síťová zařízení, na která narazí. Každé z těchto zařízení vyhodnocuje příslušné přidané informace a posílá data dále k cíli.
- Ve chvíli, kdy data dorazí k cílovému zařízení, jsou přijata síťovou kartou cílového zařízení.
- V cílovém zařízení se z dat odstraní přebytečné informace nutné k přenosu po síti, jednotlivé díly se poskládají do původního pořadí a přenesou se do cílové aplikace.

Datové jednotky vrstev modelu TCP/IP

V aplikační vrstvě se vytvoří data, která se mají poslat přes síť k cílovému síťovému zařízení.

Aplikační vrstva předává data do nižší, transportní vrstvy.

V transportní vrstvě se k datům přidává transportní hlavička, z dat se vytvářejí tzv. **segmenty**.

Z transportní vrstvy jsou data posílána do nižší, internetové vrstvy.

V internetové vrstvě se k datovému segmentu přidá síťová hlavička s informacemi o síťových adresách zdrojového a cílového zařízení, ze segmentu se vytváří datový **paket**.

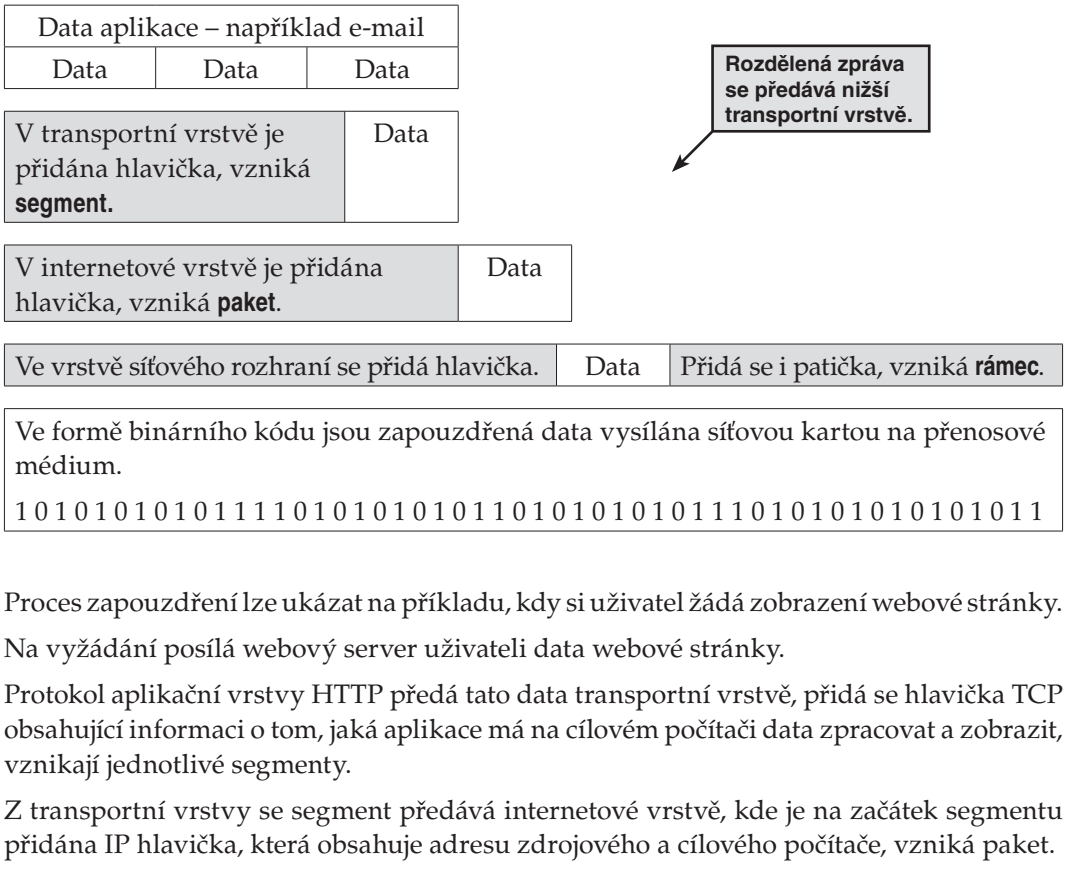
Z internetové vrstvy jsou pakety posílány do nižší vrstvy, do vrstvy síťového rozhraní.

Ve **vrstvě síťového rozhraní** se z paketu vytváří **datový rámec**, k paketu se přidávají na začátek a na konec další informace.

Celý tento proces přidávání informací k datům se nazývá **zapouzdření**.

Na závěr se data ve formě jedniček a nul vyšlou síťovou kartou na síť.

Přenos sítí



Proces zapouzdření lze ukázat na příkladu, kdy si uživatel žádá zobrazení webové stránky.

Na vyžádání posílá webový server uživateli data webové stránky.

Protokol aplikační vrstvy HTTP předá tato data transportní vrstvě, přidá se hlavička TCP obsahující informaci o tom, jaká aplikace má na cílovém počítači data zpracovat a zobrazit, vznikají jednotlivé segmenty.

Z transportní vrstvy se segment předává internetové vrstvě, kde je na začátek segmentu přidána IP hlavička, která obsahuje adresu zdrojového a cílového počítače, vzniká paket.

Z internetové vrstvy je paket předán vrstvě síťového rozhraní, kde je celý paket obalen dalšími informacemi. Na začátek se přidá hlavička obsahující fyzickou MAC adresu zdrojového a cílového zařízení (tyto MAC adresy se mění průchodem sítí, vždy jsou to MAC adresy dvou zařízení na aktuální lokální síti, která si data předávají). Na konec se přidá patička, která obsahuje kontrolní údaj umožňující zjistit, zda data dorazila do cíle v nezměněné podobě.

Nakonec jsou připravená data vyslána ve formě jedniček a nul síťovou kartou na síťové médium (jakým způsobem, závisí na typu média a síťové karty).

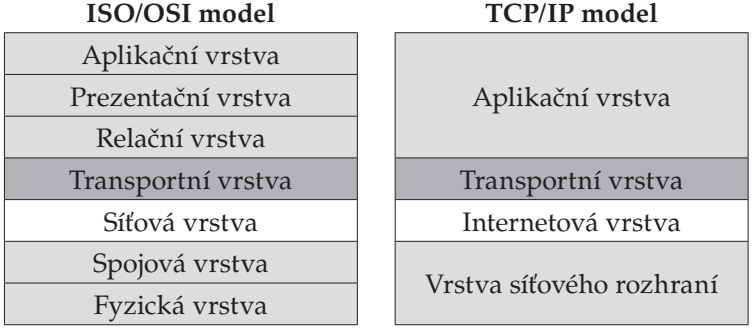
Data v této podobě cestují sítí. Ve chvíli, kdy příchozí data narazí na zařízení typu přepínač, jsou rozbalena do té míry, aby byly pro přepínač čitelné informace o MAC adresách a kontrolní údaje dokládající, zda je rámec neporušen. Tyto informace jsou aktualizovány a data jsou v binární podobě opět vyslána na síť.

Pokud data narazí na směrovač, jsou rozbalena do té míry, aby směrovač mohl přečíst z paketu údaje o IP adresách zdroje a cíle a mohl se rozhodnout, jakému svému rozhraní data pošle. Pak data opět zapouzdří, aktualizuje přidané údaje v rámci a výsledná data vyšle zvoleným rozhraním na síťové médium.

Jakmile signál dorazí k cílovému zařízení, jsou postupně odstraněny všechny přidané údaje – hlavička a patička rámce, hlavička paketu, ze segmentu se přečte a odstraní přidaná informace o aplikaci, která má data zpracovat a zobrazit, a data se ve správném pořadí spojí do původní podoby a předají cílové aplikaci.

Ve výsledku cílový počítač zobrazí požadovanou webovou stránku.

Porovnání síťových modelů



Protokoly popisované TCP/IP modelem je možné popsat i v OSI modelu ve větších podrobnostech. Protokoly a funkce odpovídající v modelu TCP/IP vrstvě síťového rozhraní, jsou dále rozděleny do dvou vrstev v modelu OSI, do **vrstvy spojové** a **fyzické**. Zde je možné detailně popsat jejich odlišnosti.

Obě vrstvy – **síťová** v OSI modelu a **internetová** v TCP/IP modelu – se zabývají způsobem směrování dat po síti směrem k cílovému zařízení. Obě vrstvy popisují IP protokol, který se směrováním zabývá.

Transportní vrstvy v obou modelech se zabývají rozdělením jednotlivých současně probíhajících datových přenosů mezi zdrojovým a cílovým zařízením. Tyto vrstvy popisují, jak si počítače potvrzují předaná data, jak se chovají v případě chybného přenosu, jak data rozdělují do segmentů, které následně posílají a rekonstruují v původním pořadí. Protokoly této vrstvy jsou **TCP** (***Transmition Control Protocol***) a **UDP** (***User Datagram Protocol***).

Protokoly a funkce aplikační vrstvy v TCP/IP modelu jsou detailněji rozděleny do tří vrstev modelu OSI, do **vrstvy relační**, **prezentační** a **aplikační**.

OSI model – princip přenosu a přenosové protokoly

V další části bude popisována síťová komunikace pomocí detailnějšího modelu OSI.

Aplikační vrstva	Umožňuje aplikacím na obou stranách přenosu spolupracovat. Protokoly této vrstvy jsou **FTP, TFTP, DNS, DHCP, SMTP, POP3, SSH** a další.
Prezentační vrstva	Převádí data do tvaru čitelného pro aplikaci, tento tvar může být různý na obou stranách přenosu. Zajišťuje kódování a konverzi dat do podoby čitelné v cílovém zařízení. Provádí kompresi dat tak, aby je cílové zařízení mohlo dekomprimovat. Kryptuje data tak, aby nebyla čitelná v síťových mezičláncích, ale až v cílovém zařízení.
Relační vrstva	Zajišťuje a synchronizuje přenos mezi relačními vrstvami obou stran, vytváří, obnovuje a ukončuje relaci mezi protistranami. Protokoly této vrstvy jsou **NetBIOS, Apple Talk, SSL**.

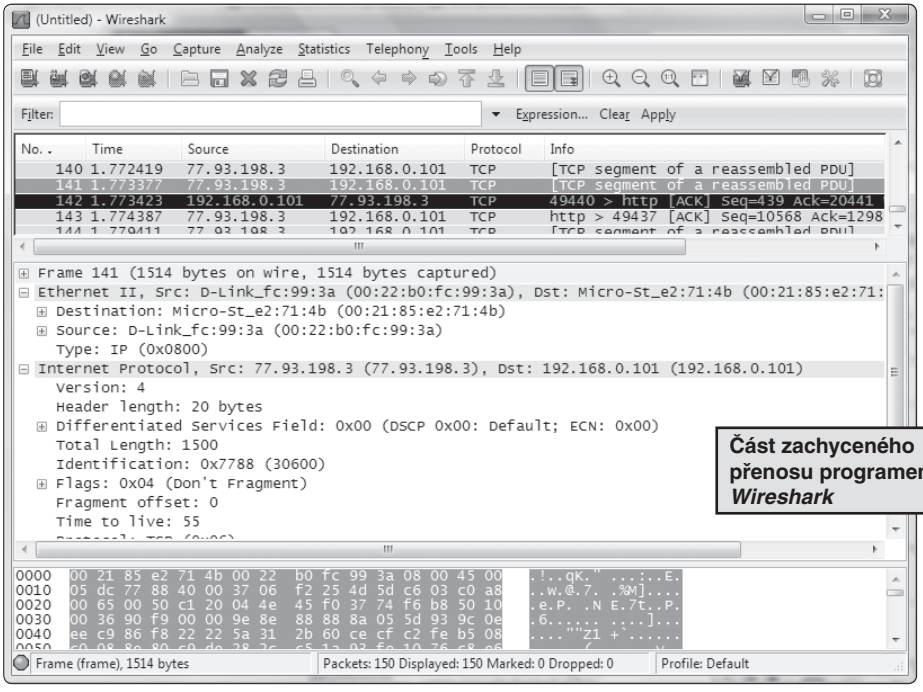
Tabulka pokračuje na následující straně.

Transportní vrstva	Obsahuje údaje o zdrojovém a cílovém portu, čímž je umožněno více současných přenosů. Porty slouží k identifikování procesu (aplikace), který má daná data zpracovat. Informace o portech jsou nutné, protože běžně mezi počítači probíhá více datových přenosů (například zobrazování webové stránky, posílání e-mailu, hraní síťové hry…). Tato vrstva kontroluje kvalitu přenosu dat. Protokoly této vrstvy jsou TCP a UDP.
Síťová vrstva	Obsahuje údaje o zdrojové a cílové síťové adrese. Tato informace je nutná především tehdy, jestliže síťový přenos probíhá mezi různými oddělenými lokálními sítěmi. Na hranici každé lokální sítě je směrovač (nebo podobné zařízení), který prozkoumá síťové adresy uvedené v paketu, data opět zapouzdří do rámce a odesílaná data přepne na rozhraní, jež vede k cíli. Na této vrstvě pracují směrovače. Nejznámější protokol této vrstvy je protokol IP, další jsou protokoly ICMP a ARP.
Spojová vrstva	Obsahuje údaje o zdrojové a cílové fyzické adrese. Tyto adresy jsou zodpovědné za doručení rámce v oblasti lokální sítě. Pokud jsou data posílána mimo lokální síť, jsou na hraničním zařízení – typicky směrovači – informace o fyzických adresách vyměněny za nové. Rámec vždy obsahuje údaje o MAC adresách zdrojového a cílového zařízení v aktuální lokální síti, ve které se data pohybují. Na této vrstvě pracují přepínače, mosty a síťové karty.
Fyzická vrstva	Zabývá se synchronizací a časováním bitů posílaných na síť tak, aby bylo možné data odeslat požadovanou přenosovou rychlostí zvolenou technologií.

Zachytávání síťové komunikace

K zachytávání síťové komunikace jsou vhodné programy **Ethereal** nebo **Wireshark**. V názorné podobě zobrazí zachycená data, včetně čísel portů, IP adres, MAC adres, protokolů a dat. Programy ke stažení můžete nalézt na adresách http://www.ethereal.com/ a http://www.wireshark.org/.

Na následujícím obrázku je vidět část přenosu sítí zachyceného programem **Wireshark**, kdy zdrojový počítač požádal o zobrazení webové stránky www.computermedia.cz.



Část zachyceného přenosu programem *Wireshark*

Ve výpisu (ten je příliš dlouhý na to, aby se celý vešel do obrázku) je vidět zdrojová a cílová MAC adresa počítače a směrovače na lokální síti, který tvoří bránu do internetu, a síťová adresa počítače a směrovače. Dále je vidět, že na síti běží protokol IP verze 4, že protokol transportní vrstvy je TCP, aplikační protokol je HTTP, a další spousta informací.

6. Aplikační vrstva a její protokoly

Aplikační vrstva slouží jako prostředník mezi síťovou aplikací a přenosem po síti.

Aplikační vrstva používá protokoly implementované do uživatelských aplikací a procesů.

Nejznámějšími protokoly v této vrstvě jsou protokoly **DNS – překlad internetových jmen na IP adresy**, **HTTP – přenos a zobrazení webových stránek**, **SMTP – přenos e-mailů**, **Telnet – vzdálený přístup přes síť k cílovému zařízení**, **FTP – přenos souborů mezi dvěma systémy**, **DHCP – dynamické přidělování síťových nastavení síťovému zařízení**.

Zdrojové a cílové zařízení spolu komunikují během síťového přenosu pomocí protokolů aplikační vrstvy, které si musí odpovídat, aby komunikace byla úspěšná.

Protokoly definují přesná pravidla pro komunikaci síťových zařízení, specifikují, jaká a jak formátovaná data se sítí posílají.

Spojení klient–server

Klient je zařízení s příslušným softwarem, které uživatel použije k získání potřebných dat ze serveru.

Server je zařízení, které poskytuje požadované služby a data. Typicky je to počítač, který má uložena nejrůznější data, například obrázky, webové stránky, videa, hudbu, databáze apod., a na vyžádání klientem je poskytuje. V případě tiskového serveru posílá požadavek klienta na tisk příslušné tiskárně.

Na serveru běží na pozadí určitý proces nazývaný **démon,** který čeká na požadavek od klienta a ve chvíli, kdy nějaký požadavek dorazí, jej vyřizuje.

Klient vyvolá požadavek a server na něj odpoví. Může požadovat zadání ověřovacích údajů, jako jsou uživatelské jméno a heslo.

Většinou data putují od serveru ke klientovi – tzv. **download**, ale může to být i naopak, například při ukládání dat na server – tzv. **upload**.

Příkladem spojení **klient–server** může být zobrazení webové stránky uložené na serveru. Klient požádá o zobrazení určité stránky a server mu příslušná data pošle. Zaslaná data se pak v příslušné aplikaci – v internetovém prohlížeči – zobrazí.

Server běžně vyřizuje celou řadu požadavků od klientů současně a jednotlivá spojení se nesmí promíchat.

Spojení typu peer-to-peer

Počítače v síti typu **peer-to-peer** sdílí soubory, tiskárny, obecně zdroje, bez pomoci pověřeného serveru.

Podle toho, který počítač zdroje žádá a který je poskytuje, se v aktuální chvíli mění role počítačů, jeden je chvíli v roli klienta a druhý v roli serveru. Ve stejné chvíli může od počítače, který je klientem v jednom spojení, požadovat zdroje jiný počítač, a pak je tento počítač také v roli serveru.

Síť typu klient–server a peer-to-peer

V sítích typu **peer-to-peer** fungují jednotlivé počítače jako rovnocenní partneři. Každý může zvolit data, která bude sdílet s ostatními. Přístup ke svým zdrojům mohou vázat na použití hesla.

Neexistuje centrální místo, odkud by se poskytovaly sdílené zdroje. Nevýhodou může být, že pokud všichni jednotliví uživatelé nezálohují svá data, mohou o ně v případě problému se systémem přijít.

V případě centrálního serveru, který poskytuje sdílené zdroje, může administrátor serveru zajistit průběžné zálohování důležitých dat. Uživatelé mohou ukládat svá data do svých vyhrazených prostor na serveru, který se postará o jejich zálohu.

Server bývá po stránce výkonu i připojení na síť uzpůsoben k poskytování zdrojů ostatním klientským počítačům, takže se ho toto poskytování tolik nedotýká jako v případě, kdy sdílená data poskytuje běžný počítač v síti typu **peer-to-peer**. Běžný počítač toto poskytování pozná na svém aktuálním výkonu citelněji.

U sítí typu **peer-to-peer** není zvolen žádný centrální administrátor, který by síť spravoval. Každý si zajišťuje poskytování a správu sám, není potřeba žádný serverový operační systém, stačí běžný operační systém podporující sdílení zdrojů po síti.

Síť typu peer-to-peer lze docela snadno spravovat v případě malého počtu zapojených počítačů, ale jestliže síť začne více růst, může nastat problém. Pak je vhodnějším řešením síť typu **server–klient**, kdy se o vše stará jeden administrátor na centrálním serveru. Uživatelé mají na serveru své účty a přihlašují se k serveru svým uživatelským jménem a heslem.

Server může poskytovat nejen data, soubory, ale i přístup k tiskárně, aplikacím, službám. Dokáže vyřizovat požadavky od více klientů současně.

Možnou nevýhodou serveru je jeho vyšší cena, potřeba mít znalého administrátora, který se postará o sdílení dat, uživatelské účty, zálohování a další bezpečnost. V případě výpadku serveru přijdou klientské počítače o přístup ke svým datům uloženým na serveru do doby opravy nebo náhrady.

Porty protokolů HTTP, DNS, FTP, SMTP, POP, DHCP, Telnet

Pomocí čísel portů v transportní vrstvě se identifikují aplikace a služby na zdrojovém a cílovém zařízení.

HTTP obvykle přistupuje na port 80, DNS na port 53, SMTP na port 25, POP na port 110, Telnet na port 23, DHCP na porty 67 a 68, FTP na porty 20 a 21.

HTTP – Hypertext Transfer Protocol

Protokol HTTP zajišťuje přenos požadovaných dat z webového serveru do počítače klienta a určuje způsob komunikace serveru s klientem.

Klientský počítač zažádá o zobrazení webové stránky pomocí internetového prohlížeče. Tento požadavek se pošle k webovému serveru, který odpoví odesláním požadované stránky. V klientském počítači se pak zaslaná data zobrazí v internetovém prohlížeči příslušným způsobem.

Uživatel zadá do internetového prohlížeče jmennou adresu (např. www.ivasp.info/index.php). Část obsahující jméno serveru (zde například www.ivasp.info) se pomocí DNS překladu převede na číselnou síťovou adresu, která je použita pro spojení se serverem. Po zjištění číselné síťové adresy serveru dojde k jeho kontaktování a vyžádání souboru (zde například index.php). Server zašle klientovi HTML kód, který se ve výsledku zobrazí v internetovém prohlížeči.

Webová adresa se označuje jako **URL** (***Uniform Resource Locator***) – je to jednoznačná adresa v síti Internet.

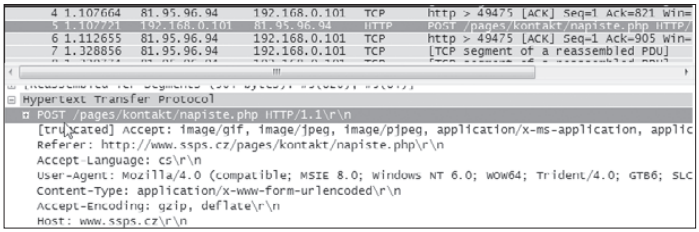
Základní tvar URL

URL adresa má obecný tvar: http://www.cokoliv.cz/adresar/index.html

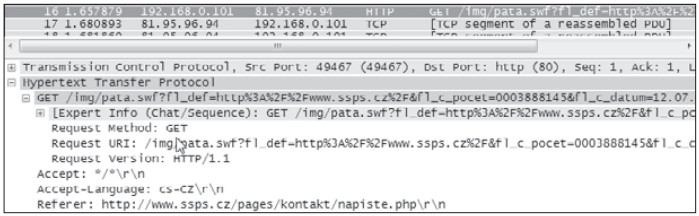
- http – protokol komunikace
- www.cokoliv.cz – plně specifikované doménové jméno (jméno serveru), také může být použita IP adresa serveru (např. **194.210.50.100**)
- adresar – adresářová cesta na zvoleném serveru
- index.html – volaný soubor v tomto adresáři

Metody, které používá **HTTP** protokol, jsou **GET** a **POST**. Metoda **GET** je využita, když si klient vyžádá zaslání webové stránky serverem. Metoda **POST** se používá v případě, že klient posílá data na server, například když vyplní formulář na webové stránce a odešle jej na server nebo odesílá přes webové rozhraní soubor na server.

Zachycení přenosu dat v případě stahování dat ze serveru i odesílání dat na server je vidět na následujících dvou obrázcích.



Ukázka zachycení odeslání dat z formuláře webové stránky na server – HTTP použil metodu POST.



Ukázka zachycení přijímání dat ze serveru – HTTP použil metodu GET.

HTTP nepatří mezi protokoly se zabezpečeným přenosem. Data se přenášejí v prostém textu a kdekoliv po cestě jsou odposlechnutelná!

Pro šifrovaný přenos webových stránek po síti se používá **protokol HTTPS** – ***Hypertext Transfer Protocol Secure***.

DNS – Domain Name System

DNS je služba pracující na principu **klient–server**. Slouží k překladu jmenných názvů na odpovídající IP adresy. DNS zajistí tento překlad ve chvíli, kdy jej nějaká služba nebo aplikace v počítači potřebuje.

Síťová zařízení jsou identifikovatelná svou síťovou adresou. Pro snadnější práci a pamatování adres jsou zařízením přiřazena i jména.

DNS slouží k překladu jmenných adres na jejich číselné ekvivalenty.

DNS překlady se běžně používají při přístupu k webovým stránkám, protože uživatel si snadněji zapamatuje jmennou adresu než IP adresu.

Lépe se pamatuje například adresa www.seznam.cz než **77.75.76.3**.

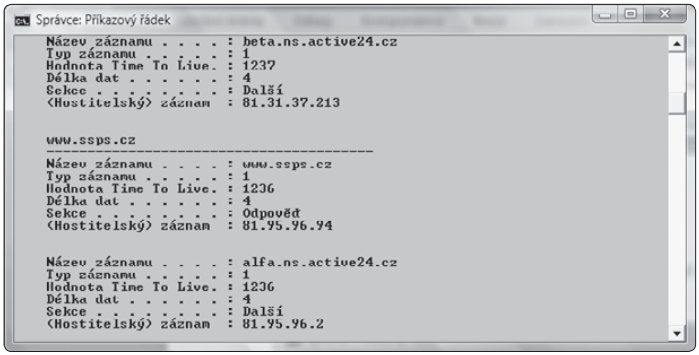
DNS překlady jsou uloženy na síti DNS serverů, které si tyto seznamy aktualizují a předávají. Ve chvíli, kdy počítač požádá například o webovou stránku, ale nezná překlad jmenné adresy na síťovou, zašle požadavek na tento překlad svému DNS serveru, který mu odešle odpověď. Pokud ji DNS server sám nezná, přepošle požadavek dalšímu

DNS serveru, který překlad zajistí (nebo dále přepošle). Po získání odpovědi již počítač přistupuje k webové stránce pomocí získané číselné adresy.

Informace o zjištěných DNS překladech si DNS server udržuje ve své cache paměti, a pokud příště přijde dotaz na stejný překlad, už se nedotazuje jiných DNS serverů, ale použije překlad uložený ve své paměti.

Stejně si provedené DNS překlady pamatuje i koncový počítač, a tak se příště již na DNS server nemusí obracet.

Obsah paměti (na OS Windows), ve které se ukládají DNS překlady, můžete zobrazit příkazem **ipconfig /displaydns**.



Výpis DNS překladů z paměti počítače příkazem *ipconfig /displaydns*

S oprávněními správce lze paměť s uloženými DNS překlady vymazat příkazem **ipconfig /flushdns**.

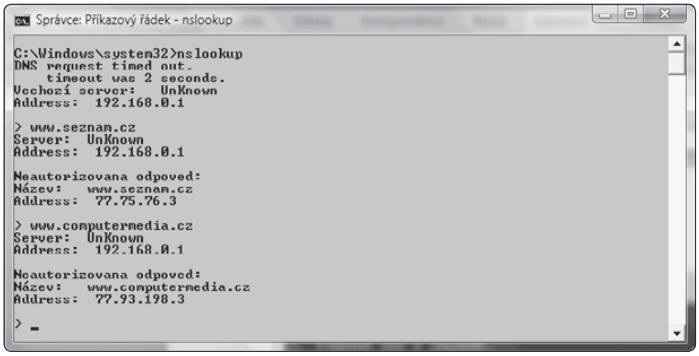
Počítač má ve své síťové konfiguraci uloženo, na jaký DNS server se má obracet. Tuto adresu většinou dostane od svého poskytovatele internetového připojení.

V případě, že počítač leží na lokální síti a jeho přístup k internetu mu zprostředkovává směrovač stejně jako přeposílání DNS požadavků na DNS server ležící na internetu, stačí, aby měl počítač nastaven jako DNS server lokální rozhraní směrovače, ke kterému je připojen. Směrovač pak může zajistit přeposlání DNS požadavku venkovnímu DNS serveru a zpětné přeposlání výsledku počítači.

Ručně lze zjistit překlad jmenné adresy na číselnou adresu pomocí příkazu **nslookup**.

Po napsání příkazu **nslookup** se zobrazí informace o nastaveném DNS serveru a pak lze napsat jmennou adresu, kterou je třeba přeložit. Po stisknutí klávesy **Enter** se jmenná adresa přeloží na číselnou IP adresu.

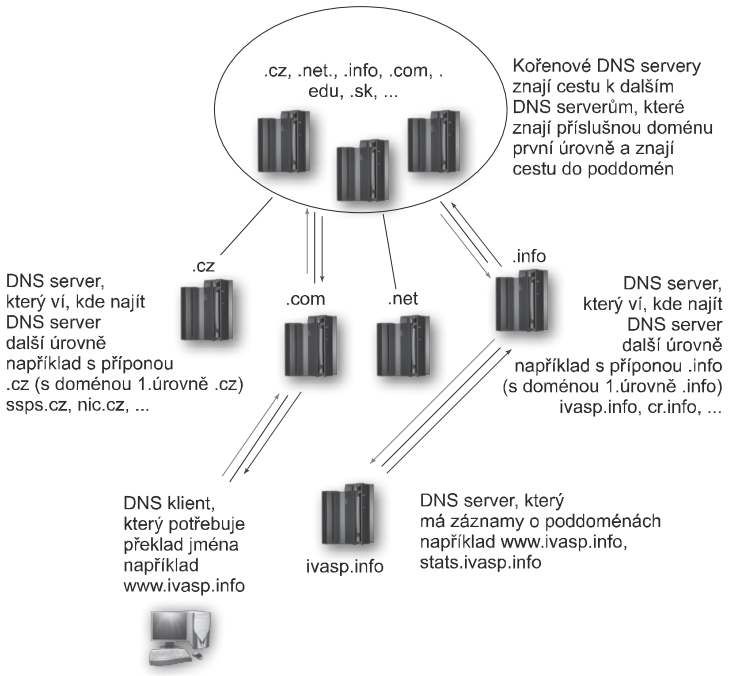
DNS servery mají stromovou strukturu. Například ve jmenné adrese mail.abcdefghij.cz je nejvyšší doména **cz,** doména druhé úrovně je **abcdefghij** a doména třetí úrovně je **mail.**



DNS servery nejvyšší úrovně mají ve svých záznamech informaci o doménách prvního řádu (**cz**, **com**, **edu**, **info**…).

Pro získání informací o doménách druhého řádu se server nejvyšší úrovně obrací k DNS serveru nižší úrovně, který v sobě shromažďuje informace o doménách druhého řádu, jež příslušejí určité doméně prvního řádu (například a.cz, b.cz, c.cz, abcdefghij.cz…).

Tento DNS server předá požadavek na specifikaci domény 3. řádu dalšímu podřízenému DNS serveru, který má informace o doménách třetího řádu pro příslušnou doménu druhého řádu (například mail.abcdefghij.cz, www.abcdefghij.cz, sprava.abcdefghij.cz…).



Pokud počítač potřebuje přistupovat k některé jmenné adrese na webu, požádá o překlad svůj DNS server. Pokud ten nezná překlad, pošle dotaz ke svému nadřízenému DNS serveru. A tak to jde dál, třeba až ke kořenovému DNS serveru, jenž ví, který jeho podřízený DNS server má informace o příslušné adrese, viz předchozí obrázek. Adresy a jejich překlady si DNS server ukládá do své paměti, takže v průběhu síťového provozu se naučí jednotlivé překlady jmen na síťové adresy a nadřízené DNS servery dotazuje méně.

POP, SMTP a IMAP

POP – *Post Office Protocol*, **SMTP** – *Simple Mail Transfer Protocol*

Tyto protokoly využívá pro své fungování e-mail. E-mail (zkratka elektronické pošty) je aplikace typu **klient–server**.

Uživatel vytvoří zprávu pomocí e-mailového klienta (například **Microsoft Outlook**, **Mozilla Thunderbird**, **The Bat**…). E-mailový klient slouží k přijímání, vytváření a odesílání elektronické pošty.

K odeslání e-mailu slouží protokol SMTP, ke stažení e-mailu ze serveru většinou protokol POP.

SMTP slouží nejen k odeslání e-mailu z klienta na poštovní server, ale také k posílání e-mailu mezi poštovními servery při jeho doručování do cílové poštovní schránky. Od doby svého vzniku v 80. letech 20. století prošel mnoha změnami a vylepšeními.

SMTP server poslouchá obvykle na portu 25. Pokud chce klient poslat e-mail, připojuje se k SMTP serveru na tomto portu.

Po vytvoření zprávy ji e-mailový klient pošle svému poštovnímu serveru a ten se rozhodne, co s ní udělá. Pokud má příjemce svou schránku na stejném poštovním serveru, uloží e-mail do jeho schránky. Pokud ne, odešle e-mail sítí na cílový poštovní server.

Chce-li uživatel získat zprávu z poštovního serveru, spojí se jeho e-mailový klient se svým poštovním serverem a zprávu ze své poštovní schránky stáhne.

Spojení za účelem stažení e-mailu z poštovní schránky probíhá pomocí protokolu POP. **Klient se obvykle připojí k portu 110**.

Pokud uživatel nemá na počítači e-mailového klienta, může se většinou se svou poštovní schránkou na poštovním serveru spojit přes internetový prohlížeč.

Jestliže doručení e-mailu do cílové poštovní schránky selže, může to být například z důvodu, že adresát nebo cílový poštovní server vůbec neexistují (pak se odesílateli zasílá zpráva o nedoručení), nebo proto, že je cílový poštovní server dočasně nedostupný (v tom případě by měla být poštovní zpráva po určitou dobu uložena na odesílajícím poštovním serveru, který bude po určitou dobu opakovaně zkoušet e-mail doručit).

Podobně jako se pro stažení e-mailové zprávy ze serveru používá protokol POP, je možné využít i **protokol IMAP** – ***Internet Message Access Protocol***.

IMAP potřebuje trvalé on-line spojení se serverem. Se složkami v poštovní schránce a s e-maily pracuje na straně serveru. Na počítač se stahují ze serveru jen hlavičky zpráv a celá zpráva se stahuje až v případě, že ji chce uživatel číst.

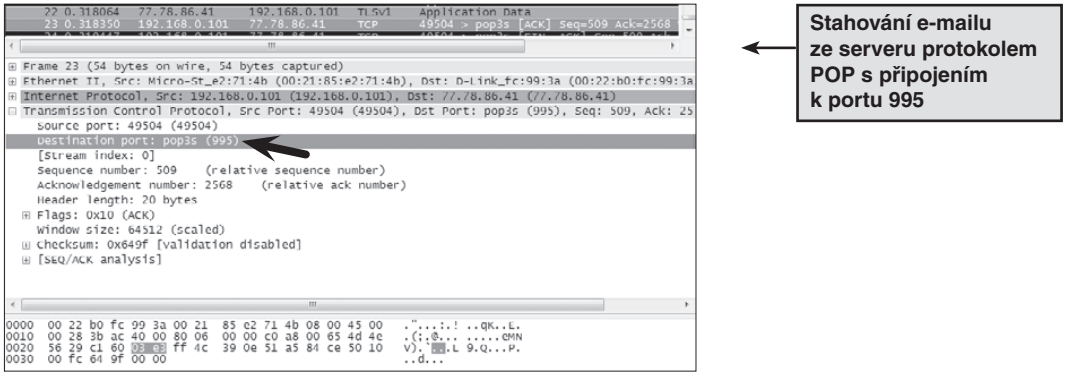
IMAP server poslouchá na portu 143.

IMAP umožňuje přistupovat k poštovní schránce na serveru z více klientů najednou. Informace o stavu zprávy, zda by přečtena, smazána, nebo zda na ni bylo odpovězeno, se uchovávají na straně serveru, takže jsou viditelné všem připojeným klientům.

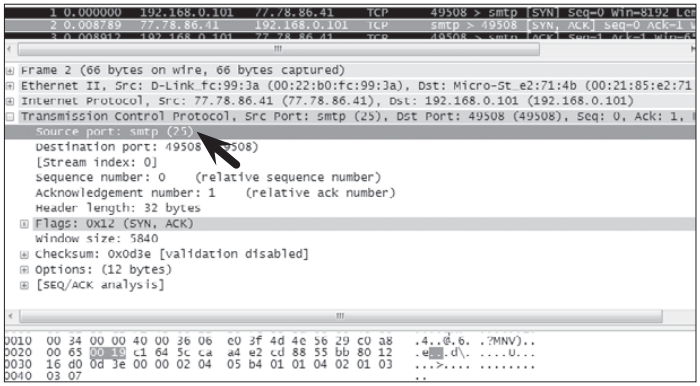
V případě protokolu POP není umožněna práce se schránkou jako v případě protokolu IMAP.

IMAP umožňuje prohledávat zprávy přímo na serveru, aniž by je klient musel předtím stáhnout do počítače uživatele.

Na následujícím obrázku je zachycena část přenosu pomocí protokolu POP. Připojení bylo provedeno k vyhrazenému portu **995**.



Na následujícím obrázku je zachyceno odeslání e-mailu klientem na server. Byl použit protokol SMTP a připojení bylo uskutečněno k portu **25**.



Odeslání e-mailu klientem na server pomocí SMTP k portu 25

FTP

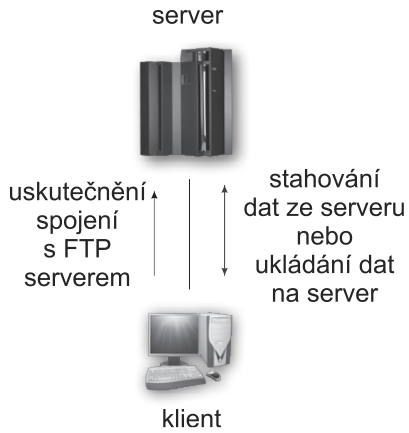
FTP – *File Transfer Protocol*

Tento protokol slouží ke stahování a nahrávání souborů z uživatelského počítače na server a naopak.

Je to aplikační protokol typu **klient–server**.

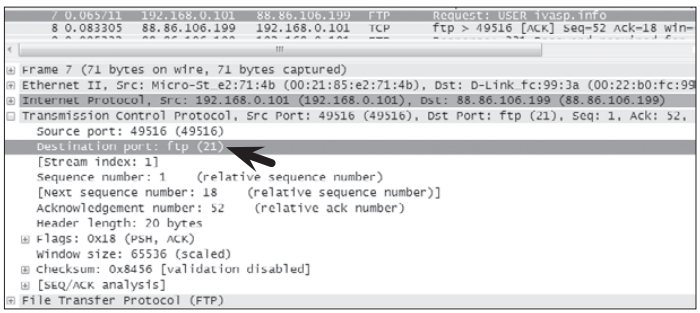
Na začátku se vytvoří spojení mezi klientem a serverem. Obvykle se vyžaduje přihlášení pomocí uživatelského jména a hesla. Po přihlášení získá uživatel nastavená oprávnění pro přístup k souborům na serveru.

Po vytvoření spojení je možné stahovat a nahrávat soubory ze serveru a naopak.



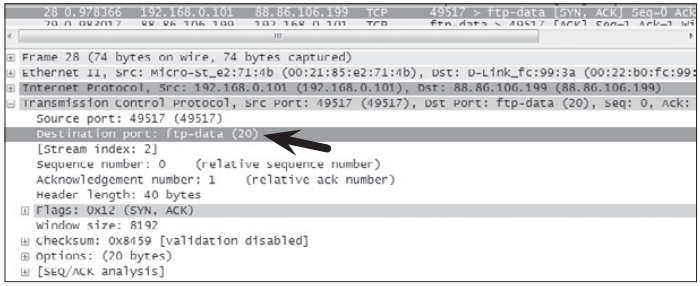
Pro vytvoření spojení se klient připojuje k portu 21. Pro transport souborů se pokaždé připojuje k portu 20.

Na následujícím obrázku je vidět zachycení části připojovacího procesu klienta k FTP serveru, probíhá připojování k portu **21**.



Připojování klienta pomocí protokolu FTP k portu 21

Na následujícím obrázku je vidět část zachyceného přenosu při stahování souboru pomocí FTP ze serveru. Probíhá připojování k cílovému portu **20**.



Stahování souboru z FTP serveru, připojení se uskutečňuje k portu 20

DHCP

DHCP – ***Dynamic Host Configuration Protocol***

Tato služba umožňuje síťovým zařízením, například počítačům, získávat z DHCP serveru síťová nastavení, jako například IP adresu, masku podsítě, adresu brány, adresu DNS serverů atd.

Pokud má počítač ve svém síťovém nastavení uvedeno, že má získávat síťové parametry pomocí DHCP, pak po spuštění vysílá žádost na DHCP server, aby mu poskytl potřebné údaje.

DHCP server, který žádost zaslechne, propůjčí počítači nějakou IP adresu z rozsahu, který má definován. Pošle počítači informace o všech potřebných síťových nastaveních a počítač si vše automaticky nastaví.

Ve větších sítích, kde by bylo obtížné všem počítačům nastavovat síťová nastavení napevno, je výhodné mít spuštěn DHCP server, který vše obstará automaticky. Pomocí DHCP často poskytuje nastavení svým klientům i poskytovatel internetu.

Nevýhodou může být horší kontrolovatelnost připojovaných zařízení a v případě nedostupnosti DHCP serveru i problém s nastavením síťových parametrů a nedostupnost sítě.

Ve chvíli, kdy počítač potřebuje zaslat DHCP nastavení, posílá do sítě vyhledávací dotaz typu broadcast (přijmou jej všechna zařízení na lokální síti), tzv. **DHCP DISCOVER paket**, který slouží ke zjištění, jaké jsou na síti DHCP servery.

DHCP servery, které tento dotaz přijmou, odešlou počítači síťová nastavení (IP adresu, bránu, masku podsítě, adresu DNS serverů, dobu zápůjčky IP adresy), tzv. **DHCP OFFER paket**.

Protože počítač může dostat více nabídek od různých DHCP serverů, musí všem dát najevo, jakou nabídku přijal. Odešle do sítě broadcast s informací, jakou nabídku od jakého DHCP serveru přijal. Jde o tzv. **DHCP REQUEST paket**.

Pokud nabídka serveru stále platí, odešle počítači potvrzující zprávu, tzv. **DHCP ACKNOWLEDGEMENT**. Pokud nabídka již není v platnosti, například proto, že odezva počítače trvala příliš dlouho a DHCP server již přidělil zmíněnou IP adresu jinému počítači, DHCP server odešle negativní potvrzení, tzv. **DHCP NEGATIVE ACKNOWLEDGEMENT**. Pak musí celý proces začít od začátku.

Protože síťová nastavení jsou počítači propůjčována jen na určitou dobu, je nutné před skončením platnosti zápůjčky vyslat serveru znovu požadavek na přidělení těchto nastavení – odeslat nový **DHCP REQUEST.** Server zkontroluje, zda tím nedojde k duplikaci síťových nastavení v síti, a pokud je vše v pořádku, potvrdí zápůjčku a prodlouží její platnost.

Na obrázku je část výpisu příkazu **ipconfig /all**. Je zde vidět, jaká síťová nastavení DHCP server počítači přidělil a dokdy jsou platná.

```
Adaptér sítě Ethernet Připojení k místní síti:

   Přípona DNS podle připojení . . . :
   Popis . . . . . . . . . . . . . . : Realtek RTL8168C(P)/8111C(P) Family PCI-E
Gigabit Ethernet NIC (NDIS 6.0)
   Fyzická Adresa. . . . . . . . . . : 00-21-85-E2-71-4B
   Protokol DHCP povolen . . . . . . : Ano
   Automatická konfigurace povolena  : Ano
   Adresa IPv4 . . . . . . . . . . . : 192.168.0.101(Preferované)
   Maska podsítě . . . . . . . . . . : 255.255.255.0
   Zapůjčeno . . . . . . . . . . . . : 12. července 2009 9:40:15
   Zápůjčka vyprší . . . . . . . . . : 18. srpna 2145 18:06:52
   Výchozí brána . . . . . . . . . . : 192.168.0.1
   Server DHCP . . . . . . . . . . . : 192.168.0.1
   Servery DNS . . . . . . . . . . . : 192.168.0.1
   Primární server WINS. . . . . . . : 192.168.0.1
   Rozhraní NetBios nad protokolem TCP/IP. . . . . . . . : Povoleno
```

Síťová nastavení přidělená DHCP serverem

Gnutella

Zajímavým protokolem fungujícím na principu **peer-to-peer** je **protokol Gnutella**. Aplikace na něm založená slouží ke stahování a nabízení souborů po síti. Každý účastník si buduje vlastní seznam dalších počítačů, které nabízejí své zdroje, a tak vzniká síťová struktura účastníků.

Když některý účastník chce najít určitý soubor ke stažení, pošle vyhledávací dotaz svému seznamu účastníků. Ti mají své seznamy dalších účastníků a takto se dotaz rozšiřuje. Následuje nabídka od počítačů, které daný soubor mají.

Počítač, který vyhledávání zahájil, pak zadá požadavek na stažení nabízeného souboru ze zvoleného umístění.

Ke spojení účastníků se nepoužívá žádný centrální server.

Telnet

Telnet slouží ke vzdálenému přístupu k síťovému zařízení. Vytváří spojení, které se chová tak, jako by byl počítač k vzdálenému zařízení připojen fyzicky.

Používá se textové rozhraní.

Spojení vytvořené pomocí Telnetu se nazývá **virtuální terminál**.

Tento protokol definuje způsob vytváření a ukončení spojení, předepisuje, jaké příkazy lze používat.

Aby bylo možné se spojit se zvoleným síťovým zařízením, které během spojení bude fungovat jako server, musí na něm běžet serverová služba Telnet démon.

Server naslouchá na portu 23.

Na připojovaném koncovém zařízení musí běžet klientská aplikace **Telnet**. Na operačním systému Windows lze **Telnet** spustit z příkazového řádku.

Připojení pomocí Telnetu je potřeba na serveru chránit heslem, aby nedocházelo k nežádoucím spojením. Takto připojený uživatel má stejné možnosti, jako kdyby seděl fyzicky přímo u tohoto cílového zařízení.

Data posílaná po síti během spojení Telnet nejsou nijak šifrovaná, běží po síti v podobě prostého textu. Pro šifrovaný přenos lze použít např. **protokol SSH**.

SSH

SSH – ***Secure Shell***

Pod tímto pojmem se skrývá jak protokol, tak i aplikace. Protokol vznikl jako náhrada Telnetu, který data posílal v nezabezpečené formě. SSH používá šifrování přenášených dat a kontrolu integrity dat. Oba účastníci spojení projdou fází ověřování.

Pomocí příkazového řádku lze data mezi propojenými počítači přenášet bezpečným způsobem.

K bezpečnému přenosu lze s využitím protokolu SSH použít **protokol SCP** (***Secure Copy***) nebo komplexnější **protokol SFTP** (***Secure File Transfer Protocol***).

Programem s grafickým rozhraním využívajícím tyto bezpečné protokoly je například **WinSCP**.

7. Transportní vrstva

Úloha transportní vrstvy

Transportní vrstva má za úkol rozdělovat data získaná z vyšších vrstev do segmentů, rozdělovat jednotlivé souběžné datové přenosy patřící pod různé aplikace, označovat jednotlivé segmenty čísly portů, zajišťovat složení příchozích segmentů ve správném pořadí do původní datové zprávy, vyžádat si data, která nepřišla v pořádku, omezovat rychlost posílání dat podle možností protistran.

Základními dvěma protokoly pracujícími na této vrstvě jsou protokoly TCP a UDP.

Segmentace dat a zpětné spojení segmentů

Data, která přicházejí z vyšších vrstev, jsou dělena do částí nazývaných segmenty. Segmenty jsou očíslovány. Na konci své cesty jsou segmenty opět složeny ve správném pořadí do datového toku, který se posílá do vyšších vrstev.

Pokud některý segment nedorazí nebo dorazí poškozen, je odesílatel kontaktován a tento segment je požadován.

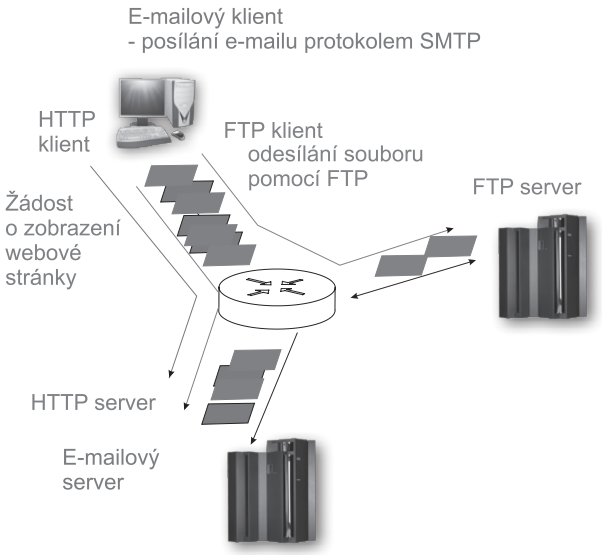
Díky segmentaci dat je možné uskutečňovat současně více spojení, která se díky malým segmentům mohou prolínat. Kdyby se data posílala vcelku, pak by jiná aplikace, která chce posílat data, musela čekat, až skončí aktuální spojení.

Označování dat pro cílovou aplikaci

Aby cílový počítač věděl, do jaké aplikace nebo procesu má zaslaná data předat, musí být v segmentech příznak identifikující cílovou aplikaci. Tímto příznakem je číslo, které se označuje jako **port**.

Rozdělení vícenásobných komunikací

Protože jednotlivé počítače mohou mít navázáno více spojení s protějšky, je potřeba tyto souběžné komunikace nějak rozdělit, aby se jejich data vzájemně nemíchala. Identifikování různých konverzací se děje pomocí čísel, jimiž se segmenty označují. Tato čísla se nazývají porty. Segment se označí číslem cílového portu, který identifikuje aplikaci nebo proces v cílovém zařízení, v němž se mají data zpracovat. Dále se do segmentu zapíše i číslo zdrojového portu, aby při odpovědi protějšku bylo možné identifikovat zdrojový proces nebo aplikaci a nedocházelo k promíchání konverzací.



Spolehlivost přenosu

Jednou z úloh transportní vrstvy je zajistit spolehlivý přenos, aby každý odeslaný datový segment dorazil do svého cíle.

To je zajišťováno pomocí rozdělení dat do jednotlivých očíslovaných segmentů, následným potvrzováním příchozích dat a opětovným zasláním nepotvrzených dat.

Kvůli této kontrole – potvrzování a opětovnému zaslání – přibývá dat na síti, a dochází tak ke zpomalení přenosu.

Podle typu aplikace můžete zvolit, zda je nutné zajistit spolehlivý přenos, zda opravdu všechna data musí dorazit bezchybně do cíle i za cenu zpomalení přenosu. Některé aplikace toto zajištění spolehlivosti nevyžadují, potřebují rychlejší přenos za cenu nespolehlivosti.

Aplikace jako například webové stránky a e-mail používají spolehlivý přenos, jinak by docházelo ke zkreslení obsahu.

Jiné aplikace, jako například přenos videa, nepotřebují 100% spolehlivý přenos, nevadí jim, když část dat nedorazí do cíle, protože na celkový přenos to nemá výrazný vliv.

Spolehlivý přenos zajišťuje protokol TCP, nespolehlivý pak protokol UDP.

TCP

TCP – ***Transmition Control Protocol***

Tento protokol je hlavním protokolem transportní vrstvy, která se po něm i jmenuje.

Je to protokol spojově orientovaný – na začátku přenosu se vytvoří oboustranné spojení mezi komunikujícími stranami, které zaručuje spolehlivé doručení segmentů ve správném pořadí.

Pro zajištění spolehlivosti přenosu přidává TCP do hlavičky segmentu více údajů než UDP, celkem 20 bytů.

V hlavičce segmentu jsou údaje o zdrojovém portu, cílovém portu, číslo sekvence, číslo potvrzení, okno (slouží k regulaci rychlosti posílaných dat), kontrolní součet, ukazatel důležitosti a další volby. Následují zasílaná data.

TCP segment	
Zdrojový port (2 byty)	Cílový port (2 byty)
Číslo sekvence (4 byty)	
Číslo potvrzení (4 byty)	
Délka hlavičky, rezervováno, příznaky (2 byty)	Okno (2 byty)
Kontrolní součet (2 byty)	Ukazatel důležitosti (2 byty)
Další volby (0–4 byty)	
Data	

TCP coby spolehlivý přenosový protokol používají aplikace jako webové prohlížeče, e-mail, aplikace pro přenos souborů.

Ukázka komunikace

Na serveru na portu 80 poslouchá HTTP server. Klientský počítač pošle na server žádost o zobrazení webové stránky. Cílový port této žádosti je 80. Zdrojový port žádosti je dynamicky přidělen zdrojovým počítačem, například má přiděleno číslo 48 152. Server odpoví zasláním požadovaných dat na cílový port klienta **48 152**, zdrojový port odeslaných dat je nyní **80**.

Zahájení spojení

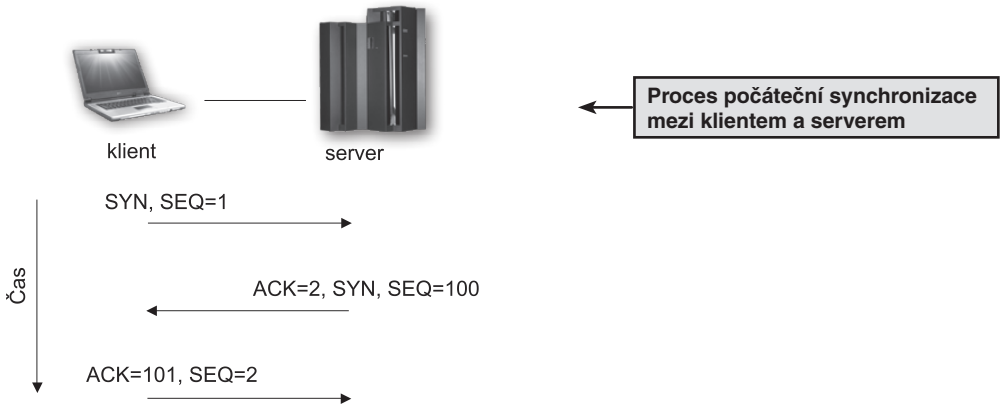
Zahájení i ukončení spojení prochází procesem vzájemných návrhů a potvrzování.

Zahájení spojení prochází procesem nazývaným anglicky **three–way–handshake** neboli „třícestné potřesení rukou“.

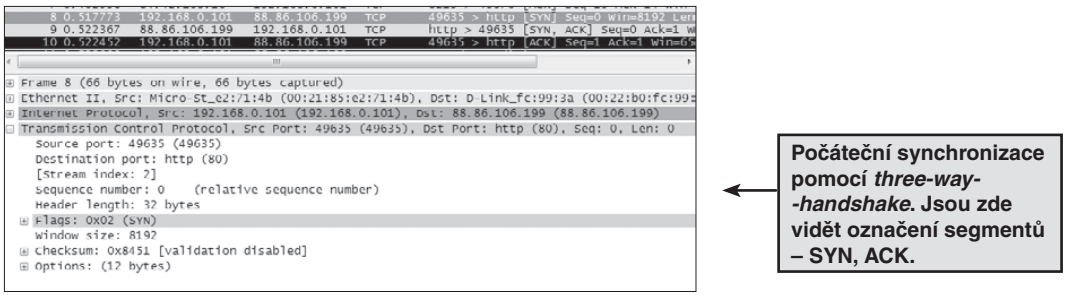
Spočívá v tom, že klient pošle serveru synchronizační segment s označením **SYN** jako žádost o synchronizaci. Tento segment obsahuje úvodní číslo sekvence **SEQ**.

Server odpoví, že na tuto žádost přistupuje, odešle segment s označením **SYN** a **ACK** (***acknowledgement*** – potvrzení žádosti, **SYN** – žádost o synchronizaci), **ACK** nastaví na hodnotu o jedna vyšší, než bylo zaslané číslo sekvence (očekává další sekvenci), **SYN** nastaví na vlastní číslo sekvence **SEQ**.

Klient uzavře tuto synchronizaci zasláním potvrzujícího segmentu s označením **ACK**, jehož hodnota je o jedna vyšší, než bylo číslo sekvence zaslané serverem. Dále pošle sekvenci s požadovaným číslem. Tím je spojení navázáno a může začít přenos.

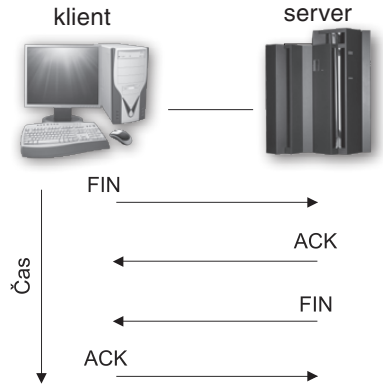


ACK označuje potvrzení, **SYN** je synchronizační údaj, **SEQ** je číslo sekvence.



Ukončení spojení

Proces ukončení spojení má čtyři fáze. Počítač, který chce spojení ukončit, vyšle segment s žádostí o ukončení – **FIN.**



Počítač na druhé straně potvrdí, že s ukončením souhlasí, pošle segment s označením **ACK.** Poté pošle ještě další segment s označením **FIN.**

První počítač potvrdí přijetí segmentu s označením FIN zasláním segmentu s označením **ACK.**

Všechny tyto segmenty, které jsou součástí komunikace potřebné k ukončení spojení, také obsahují svá označení pomocí čísel sekvence.

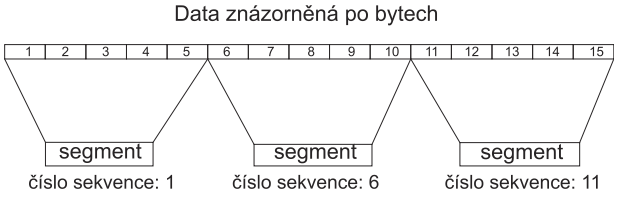
Potvrzování přijetí segmentů

Po navázání spojení cílové zařízení vždy potvrdí přijetí doručených dat. Odesílací zařízení pak posílá další data. Pokud do zdrojového zařízení nedorazí potvrzení o přijetí odeslaných dat do určité doby, vyšle tato data znovu.

V TCP segmentu se kromě ukazatele **ACK** (potvrzení), **SYN** (synchronizace) a **FIN** (ukončení) mohou objevit i ukazatele **URG** (urgentní, obsahuje důležitá data), **PSH** (***push***, význam není ustálený, často znamená, že segment nese data, která se mají předat aplikaci) a **RST** (reset spojení, odmítnutí spojení).

Jak už bylo řečeno výše, každý TCP segment nese ve své hlavičce číslo, které označuje jeho pořadí mezi ostatními segmenty proto, aby mohly být segmenty v cíli poskládány ve správném pořadí. Toto číslo se nazývá **číslo sekvence** – označuje se **SEQ.**

Číslo sekvence označuje počet bytů, které byly již během tohoto spojení odeslány, plus **1**.



Protější zařízení, které data přijímá, potvrzuje zdrojovému zařízení přijetí pomocí ukazatele **ACK**, který nastaví na počet přijatých bytů plus jedna. Například je-li číslo sekvence 6 a počet bytů v segmentu je 5, bude cílové zařízení požadovat zaslání dalšího segmentu s číslem sekvence 11. Nastaví **ACK=11**. Tím zdrojovému zařízení potvrdí, že v pořádku přijalo předchozí data.

Je zřejmé, že kdyby probíhalo potvrzování každého segmentu, byla by přenosová linka neúměrně zatížená. Proto se segmenty posílají po skupinách a potvrzuje se přijetí celé skupiny. Cílové zařízení potvrdí přijetí tohoto souboru segmentů zasláním segmentu s parametrem **ACK** nastaveným na počet všech přijatých bytů plus **1**.

Opakované odeslání nedoručených dat

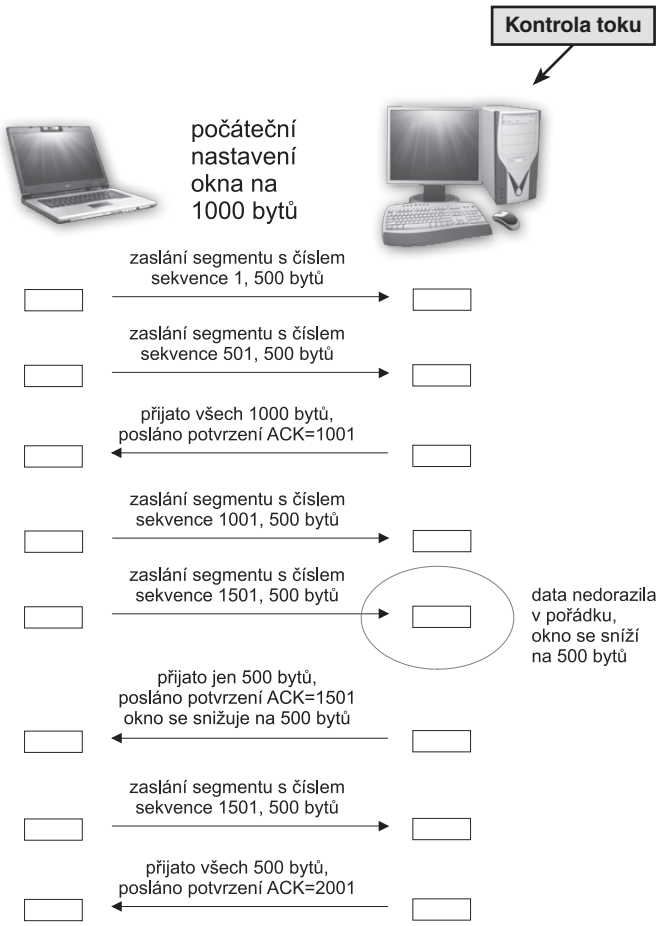
Cílový počítač potvrdí zdrojovému počítači jen spojitou řadu úspěšně přijatých segmentů. Jestliže v sadě segmentů zasílaných pospolu některé chybí nebo jsou poškozeny, například jsou v pořádku segmenty s čísly sekvencí 0–200 a pak až segmenty s čísly sekvencí 500–1000, cílový počítač potvrdí zdrojovému počítači přijetí jen první úspěšné části, tj. pošle potvrzení s nastaveným parametrem **ACK=201**. Zdrojový počítač pak bude znovu posílat segmenty s čísly sekvencí začínajících 201 a dál.

Dalším opatřením proti chybám je automatické odesílání segmentů, na které nepřišlo potvrzení do určité doby. Kopii odeslaného segmentu si zdrojový počítač po určitou dobu ponechává v paměti, po odeslání spouští časovač, a pokud do určité doby nedostane od protistrany potvrzení o přijetí, automaticky segment odesílá znovu.

Kontrola toku

K optimalizaci přenosu slouží další parametr TCP segmentu – tzv. **okno**. Jeho velikost se stanoví na počátku spojení během procedury **three-way-handshake**. Je to číslo, které udává, kolik bytů dat je možné poslat bez potvrzení přijetí. Tímto mechanismem můžete optimálně využít přenosovou kapacitu linky a předcházet zahlcením linky a síťových zařízení optimalizací velikosti okna.

Velikost okna se automaticky mění podle toho, zda jsou data přijímána bez problémů. Pak můžete okno zvětšit, zvýšit počet přijímaných dat bez potvrzování. Sníží se tím počet potvrzení, a tím se sníží i zahlcování linky těmito potvrzeními. Naopak, dochází-li na síti ke ztrátám nebo poškozením přenášených dat,



velikost okna se sníží, čímž se sníží rychlost posílání dat na síť. Zdrojový počítač bude muset čekat na potvrzení o přijetí menší sady segmentů, než pošle další data.

UDP

UDP – ***User Datagram Protocol***

Tento protokol je využíván v aplikacích, které nepotřebují spolehlivý přenos, kterým nevadí, když se část dat ztratí, poškodí nebo přijdou v jiném pořadí. Nevyžadují opětovné zaslání těchto chybných a ztracených dat, případně o opětovné zaslání požádají samy. Doručování datagramů v jiném pořadí, než v jakém byly vyslány na síťové médium, bývá zapříčiněno tím, že každý datagram může do svého cíle dorazit jinou cestou. Každá cesta může být jinak dlouhá, jinak rychlá, mohou se vyskytovat zdržení z důvodu zahlcení síťových prvků.

V cílovém zařízení se příchozí data seřadí v pořadí, v jakém přišla. Pokud aplikace potřebuje, aby byla data správně seřazena do původní podoby, musí si toto seřazení ohlídat sama.

UDP se na rozdíl od TCP nezabývá kontrolou toku, číslováním sekvencí, skládáním příchozích dat do patřičného pořadí nebo opětovným zasláním nedoručených nebo poškozených dat.

Tento protokol používají pro přenos například DNS, přenos hlasu pomocí **VoIP**, video streaming, některé on-line hry, **směrovací protokol RIP, SNMP, DHCP, TFTP**.

UDP je označován za **bezestavový** – což znamená, že cílové zařízení nezasílá zdroji prostřednictvím UDP protokolu zprávy o doručení..

Označuje se také za **nespojovaný**, což znamená, že před zahájením odesílání dat se protějšky nedomlouvají na potřebných parametrech, ale ve chvíli, kdy aplikace potřebuje data vysílat, začne je ihned vysílat.

Datové jednotky vytvářené protokolem UDP se nazývají **datagramy**.

V hlavičce datagramu jsou uvedeny **doprovodné informace** – **zdrojový port** (může být vynechán, protože není vyžadováno, aby cílová aplikace odpovídala odesílateli na přijetí datagramu), **cílový port** (identifikující cílovou aplikaci), **délka** a **kontrolní součet** (ten je možné vynechat, ale nevynechává se).

Celkové doprovodné informace v datagramu přidávají k datům maximálně 8 bytů.

UDP datagram	
Zdrojový port (2 byty)	Cílový port (2 byty)
Délka (2 byty)	Kontrolní součet (2 byty)
Data	

Pokud jsou v datagramu uvedeny oba porty (příchozí i odchozí), pak při odpovědi protějšku se tyto porty zamění.

Porty

Port je šestnáctibitové číslo sloužící k identifikování zdrojové a cílové aplikace, mezi kterými probíhá spojení. Na šestnácti bitech lze vytvořit 2^{16} různých čísel, což jsou čísla 0–65 535.

Zdrojový port slouží k identifikaci aplikace nebo služby na zdrojovém počítači, cílový port slouží k identifikaci cílové aplikace na cílovém počítači.

Porty se používají jak pro protokol TCP, tak i pro UDP.

Aby počítač, který je v roli klienta a spojuje se se serverem, věděl, jaké má být číslo cílového portu, musí vědět, jaké aplikaci jaký port obvykle přísluší, nebo mít nastaveno, jaký port má pro přístup k cílové aplikaci použít.

Například chce-li se webový prohlížeč na klientském počítači spojit s webovým serverem, přistupuje na cílový port **80**, pokud není nastaveno jinak. Aby server věděl, kam má odpovědět, je tato komunikace opatřena i číslem zdrojového portu, který je přidělen na klientském počítači. Toto číslo je náhodné jednoznačné číslo vyšší než 1023.

Porty jsou rozděleny do základních tří skupin

- **Dobře známé porty** – **0** až **1 023**

  Tato čísla jsou přidělena konkrétním aplikacím a procesům, například HTTP, POP, SMTP, Telnet. Při přístupu k těmto aplikacím na serveru je obecně známé, na jaké porty přistupovat.

  Na většině systémů mohou být použity pouze pro systémové procesy nebo programy spuštěné uživatelem s vysokým oprávněním.

- **Registrované porty** – **1 024** až **49 151**

  Tyto porty je možné dynamicky přidělovat jako odchozí port klientských aplikací.

  Lze je přidělovat aplikacím a procesům spuštěných uživatelem s běžnými oprávněními.

- **Dynamické porty** – **49 152** až **65 535**

  Tyto porty jsou přidělovány dynamicky aplikaci, která inicializuje síťové spojení.

Čísla portů přiřazuje a spravuje organizace **IANA** – ***Internet Assigned Numbers Authority***. Tato organizace spravuje i celou řadu jiných náležitostí spojených s internetem, jako je správa IP adres a kořenových DNS domén.

Na adrese http://www.iana.org/assignments/port-numbers získáte kompletní přehled o číslech portů.

Některé aplikace mohou komunikovat přes oba protokoly – **TCP** i **UDP**, například **DNS**, **SNMP** (***Simple Network Management Protocol***).

Soket

Jednoznačná kombinace síťové adresy IP a čísla portu se označuje jako **soket**. Například když klientský počítač přistupuje na cílový počítač s adresou **192.168.1.21** na port **80**, **soket** jako kombinace IP adresy a portu je číslo **192.168.1.21:80**.

Netstat

Ke zjištění, jaká spojení jsou zrovna aktivní, slouží příkaz **netstat**. Spustí se z příkazového řádku.

Ve výpisu lze najít lokální i vzdálenou adresu jednotlivých spojení a porty použité k identifikaci jednotlivých spojení. Je zde vidět také aktuální stav spojení.

```
C:\Windows\system32>netstat

Aktivní připojení

  Proto  Místní adresa          Cizí adresa            Stav
  TCP    192.168.0.101:49576    r4z16:8219             NAVÁZÁNO
  TCP    192.168.0.101:49591    mg-in-f125:5222        NAVÁZÁNO
```

Výpis spojení příkazem netstat

Stavy spojení

- **navázáno** (**established**) – spojení je navázáno a výměna dat může probíhat
- **naslouchání** (**listen**) – lokální spojení čeká na požadavky ze vzdáleného zařízení
- **time-wait** – lokální spojení čeká určitou dobu po odeslání požadavku na ukončení spojení předtím, než spojení ukončí
- **close-wait** – spojení je ukončeno, ale čeká na požadavek o ukončení od lokálního uživatele
- **syn-sent** – lokální spojení čeká na odpověď poté, co odeslalo požadavek na spojení
- **syn-received** – lokální spojení čeká na schválení potvrzení žádosti o spojení

8. Síťová vrstva

Úloha síťové vrstvy

Síťová vrstva zajišťuje doručení jednotlivých částí zprávy do cílového zařízení, které může být ve vzdálené síti.

Datová jednotka vznikající v síťové vrstvě se nazývá **paket**.

Zajišťuje adresování, zapouzdření dat přijatých z transportní vrstvy do paketu, směrování a následné rozbalení paketu.

Adresování spočívá v tom, že každému síťovému zařízení je přiřazena síťová adresa IP, pomocí níž lze části zprávy – pakety – směrovat do cílového zařízení.

Zapouzdření do paketu spočívá v tom, že k datovému segmentu získanému z vyšší vrstvy se přidá hlavička obsahující IP adresu lokálního zdrojového zařízení a IP adresa cílového zařízení. Adresa cílového zařízení se pak použije pro doručení paketu do cíle. Jakmile je paket připraven, je předán nižší vrstvě, která zajistí další úpravu dat, a následně se data vyšlou na síť.

Směrování je úloha, kterou vykonávají síťová zařízení zvaná routery – směrovače. Podle svých směrovacích tabulek rozhodují, kam pošlou přijatý paket. Během své cesty může paket projít mnoha různými sítěmi, které jsou vzájemně propojeny směrovači. Ty zajišťují směrování dat po cestě od zdroje k cíli. Směrovače provádějí analýzu přijatého paketu jen z hlediska zjištění cílové adresy, obsahem segmentu se nezabývají.

Rozbalení paketu nastává až ve chvíli, kdy dorazí do svého cílového zařízení. Tam je z paketu odstraněna hlavička obsahující informace o síťových IP adresách zdrojového a cílového zařízení a získaný segment je předán transportní vrstvě pro další zpracování.

Protokoly síťové vrstvy

Mezi nejrozšířenější protokoly síťové vrstvy patří **IP** – ***Internet Protocol***, konkrétněji **IPv4** a **IPv6**.

Další jsou **IPX** – ***Internetwork Packet Exchange*** (používaný v sítích **Novell NetWare**), **AppleTalk** (vytvořený firmou **Apple**, původně vyvinutý pro počítače **Macintosh**) a **DECNet** (vytvořený firmou **Digital Equipment Corporation**).

Protokol IPv4

Verze 4 protokolu IP je v současnosti nejrozšířenější verzí protokolu IP.

IPv4 byl vyvinut se záměrem doručovat pakety co nejúspěšněji, bez zajišťování kontroly toku a doručování (to zajišťuje – je-li třeba – protokol vyšší vrstvy, obvykle TCP).

Tento protokol je označován za **nespojovaný** – nevytváří si žádné spojení předtím, než uskuteční vyslání paketů – a považován za **nespolehlivý**, protože v zájmu co nejrychlejšího doručení neprovádí žádnou kontrolu doručení a toku dat (to zajišťuje vyšší vrstva), data doručuje do cíle s **nejvyšším úsilím**. Funguje **nezávisle na síťovém médiu**, po kterém jsou data přenášena.

Nespojovaná služba

Nespojovaná služba nebo protokol znamená, že si zdroj před vysláním dat nevytváří s protějškem žádné kontrolní spojení, takže nemá kontrolu nad tím, zda data přišla, ani nedává protějšku na vědomí, že se chystá vysílat. V hlavičce paketu se nevyskytují žádná políčka pro tuto kontrolu, pro navázání spojení. Pokud dochází ke ztrátám dat během přenosu, může kontrolu kvality zajistit protokol vyšší vrstvy – TCP.

Nespolehlivá služba, doručení s nejvyšším úsilím

Protože se tento protokol snaží o co nejrychlejší doručení, nezabývá se dalším potvrzováním úspěšného přijetí. V hlavičce paketu se již nevyskytují další kontrolní políčka, dat k přenosu je proto méně a zahlcení sítě nenastává tak často, jako by nastávalo, kdyby se tato další přídavná data přenášela a oboustranná kontrola probíhala.

Veškerou kontrolu toku a doručení dat má na starosti transportní vrstva, případně vyšší vrstvy.

Nezávislost na přenosovém médiu

V IP paketu není žádná informace o tom, na jaké médium budou data posílána. Paketu je jedno, zda bude vysílán po metalických rozvodech, optických kabelech, nebo bezdrátově.

Vysílání na médium má na starosti nižší, spojová vrstva, ta se zabývá přípravou na vysílání.

Jediné, čím se z hlediska vysílání na síť síťová vrstva zabývá, je maximální velikost datové jednotky, kterou lze na médium vysílat. Tuto informaci dostává síťová vrstva od spojové vrstvy. Někdy je nutné paket během přenosu rozdělit na menší díly (tzv. fragmentace). Provádí ji směrovač, který pakety přesměrovává mezi sítěmi. Některá síťová média potřebují menší pakety.

Struktura paketu IPv4

<table>
<tr><th colspan="2">1. Byte</th><th>2. Byte</th><th colspan="2">3. Byte</th><th>4. Byte</th></tr>
<tr><td>Verze</td><td>IHL</td><td>Typ služby</td><td colspan="3">Délka paketu</td></tr>
<tr><td colspan="3">Identifikace</td><td>Příznak</td><td colspan="2">Umístění fragmentu</td></tr>
<tr><td colspan="2">TTL</td><td>Protokol</td><td colspan="3">Kontrolní součet hlavičky</td></tr>
<tr><td colspan="6">Zdrojová IP adresa</td></tr>
<tr><td colspan="6">Cílová IP adresa</td></tr>
<tr><td colspan="5">Volby</td><td>Výplň</td></tr>
</table>

Verze

Verze je číslo určující verzi IP protokolu, zde verze **4**.

IHL

IHL (***Internet Header Length***) je délka hlavičky, počítáno po 32 bitech (4 bytech). Toto číslo indikuje, kde začnou vlastní data uzavřená v paketu. Minimální povolená délka hlavičky je 5.

Typ služby

Typ služby – toto označení umožní posílání paketů sítí podle priorit. Některé pakety je potřeba posílat častěji než jiné, aby byla zachována kvalita přenosu, například spojitý tok videa nebo hlasu.

Délka paketu

Délka paketu určuje délku paketu včetně hlavičky paketu a dat uvnitř. Délku počítá po bytech. Délka paketu je 16bitové číslo, které umožňuje hodnoty **0–65 535**. Takto dlouhé pakety by ale byly pro přenos nevhodné. Síťová zařízení jsou nastavena tak, aby přijímala **pakety dlouhé maximálně 576 bytů**. Delší pakety je možné posílat jen výjimečně a jen pokud se na tom vysílací a přijímací stanice dohodnou.

Identifikace

Identifikace je 16bitové číslo, které vkládá odesílatel, aby pomohl při rekonstrukci fragmentovaného segmentu. Je to unikátní číslo identifikující fragment v rámci paketu.

Příznak (flags)

Příznak je 3bitové číslo, které udává informace o fragmentování dat. Spolupracuje na interpretaci informace o fragmentování spolu s následujícím polem – **Umístění fragmentu** (***Fragment offset***).

Třetí bit v tomto poli značí hodnotu **MF** (***More Fragments***), může nabývat hodnot **0** nebo **1**.

Je-li hodnota **MF** (***More Fragments***) nastavena na hodnotu 1, znamená to, že fragment není posledním v souboru fragmentů. K určení pozice tohoto fragmentu v rekonstruovaném paketu slouží hodnota uvedená v následujícím poli – **Umístění fragmentu** (***Fragment offset***).

Je-li hodnota **MF** nastavena na hodnotu **0** a v následujícím poli **Umístění fragmentu** (***Fragment offset***) není **0**, pak se tento fragment umístí na konec rekonstruovaného paketu.

Je-li hodnota **MF** nastavena na **0** a v poli **Umístění fragmentu** (***Fragment offset***) je také **0**, znamená to, že paket není fragmentován.

Druhý bit v tomto poli značí hodnotu **DF** (***Don´t fragment***), nabývá hodnot **0** nebo **1**.

Je-li **DF** nastaveno na **1**, pak **není povoleno fragmentování paketu**.

Je-li **DF** nastaveno na **0**, pak **je paket možné fragmentovat**.

Pokud nastane situace, kdy směrovač potřebuje paket fragmentovat kvůli přenosové lince, která vyžaduje kratší pakety, a parametr **DF = 1** (fragmentace se nepovoluje), pak směrovač takový paket zahodí. **První bit** je **vždy nastaven** na **0**.

Fragment offset – Umístění fragmentu

Toto 13bitové číslo určuje umístění fragmentu mezi ostatními při rekonstrukci fragmentovaného paketu. Umístění se určuje po bytech. První fragment má hodnotu **0**.

TTL

TTL (***Time to Live***) určuje maximální dobu v sekundách, po kterou se může paket vyskytovat na síti během své cesty k cíli. Je-li hodnota nastavena na **0**, paket musí být zahozen. Číslo TTL je vždy sníženo o **1**, jakmile paket projde přes směrovač. Začíná se na výchozí přednastavené hodnotě, například 15. Toto opatření slouží k tomu, aby sítí donekonečna neběhaly nedoručitelné pakety, čímž by došlo k zahlcení přenosových linek a síťových zařízení. Tento problém může nastat v případě směrovacích smyček, které by v síti neměly být, ale přesto mohou nastat.

Protokol

Toto 8bitové číslo informuje o datech, jež paket nese, a o protokolu vyšší vrstvy, který je v přenosu použit. Jeho informace jsou zabaleny uvnitř paketu. Tím je síťové vrstvě na cílovém zařízení umožněno předat data správnému protokolu ve vyšší vrstvě. Například číslo 06 indikuje protokol TCP, hodnota 17 protokol UDP.

Kontrolní součet hlavičky

Toto číslo je výsledkem určité kontrolní výpočetní operace provedené na hlavičce paketu. Při změně nějaké hodnoty v hlavičce, například při snížení hodnoty TTL, je kontrolní součet hlavičky přepočítán.

Zdrojová IP adresa

Je to 32bitové číslo identifikující zdrojové síťové zařízení.

Cílová IP adresa

Je to 32bitové číslo identifikující cílové síťové zařízení.

Volby

Tato hodnota je volitelná, může obsahovat určité možnosti zabezpečení. Používá se málo.

Výplň

Tato hodnota je volitelná, doplňuje délku hlavičky paketu do násobku 32bitové délky.

Rozdělení síťových zařízení do skupin

Bylo by velmi nepraktické mít všechny počítače v jedné obrovské síti, proto se počítače dělí do malých celků.

Je snadnější rozdělit velkou síť do menších celků, do podsítí, a spravovat tyto podsítě, nastavit si adresování v sítích, zajistit jejich bezpečnost a výkon.

Počítače se do podsítí zařazují na základě určitých společných faktorů, jako je **fyzické umístění**, **příslušnost k různým vlastníkům** (zajištění bezpečnosti) anebo **účel**, který plní.

Podsíť můžete vytvořit z počítačů například v jedné budově, na určitém poschodí a podobně. Pokud daná skupina počítačů využívá přístup ke stejnému síťovému softwaru, je dobré umístit jej do stejné sítě s těmito počítači, aby provoz nenarušoval chod ostatních sítí.

Pro vhodné navržení sítě je potřeba vědět, jak bude tato síť vytěžována. Pokud na síti budou uživatelé, kteří budou pracovat nejčastěji na svých lokálních stanicích a jen občas budou přistupovat k síti, například za účelem občasné návštěvy webových stránek, stačí jim síť s menší datovou propustností než v případě, že se po síti budou pravidelně přepravovat velké soubory dat, s nimiž zaměstnanci budou pracovat. Pokud ve firmě existují obě tyto skupiny uživatelů, musíte pro každou z nich navrhnout síť splňující jejich nároky.

Jestliže má firma více oddělení a za každé je zodpovědná jiná osoba nebo skupina osob, je vhodné vytvořit pro každé oddělení vlastní podsíť, na níž lze snadněji zajistit bezpečnost v rámci těchto oddělených podsítí. Sítě lze mezi sebou propojit, ale za jejich správu, údržbu a bezpečnost zodpovídá jasně specifikovaný správce. Přístup mezi jednotlivými sítěmi lze monitorovat, určité aktivity povolovat nebo zakázat. Přístup na internet mohou mít všechny podsítě, nebo jen zvolené.

Připomeňme zde již dříve zmiňované vysílání typu **broadcast**. Je to odesílání zpráv, které jsou doručeny všem počítačům v rámci jedné podsítě. Každý počítač se musí doručenou zprávou zabývat a případně ji zpracovat nebo na ni odpovědět. Hranicí pro broadcast je až směrovač, který broadcasty automaticky nepřeposílá do jiné sítě. Na počítačích běží různé programy a služby, které používají broadcast ke svému fungování. Je tedy zřejmé, že pokud je na síti hodně účastníků, je těchto broadcastů také hodně, a tím síť zatěžují. Rozdělíte-li počítače do více podsítí, na jejichž hranicích je směrovač, bude interní provoz v síti opravdu interní a nebude zatěžovat jiné podsítě. Každá podsíť je samostatnou **broadcast doménou**.

Hierarchické adresování

V síti se používá **hierarchické adresování**. Pro představu si lze fungování hierarchického adresování v síti představit jako zasílání dopisu z jednoho státu do druhého. Nejprve se doručování týká doručení do daného státu, posléze do města, obvodu a až nakonec adresátovi. Adresování v síti je provedeno podobným způsobem. Směrovače směrují pakety pouze na základě adresy sítě, nezajímají se o koncové zařízení, ke kterému data proudí. Jakmile se paket dostane do příslušné sítě, lokální zařízení jako například přepínač již doručení k cílovému počítači zajistí.

Jak se zjistí adresa podsítě, do které počítač patří?

Počítač má nastavenu nějakou IP adresu a masku.

Například **IP adresa** je **192.168.1.102** a **maska** je **255.255.255.0**. Adresa podsítě se získá jako výsledek logické operace **AND** provedené na IP adrese a masce.

V našem případě je adresa podsítě **192.168.1.0**.

Část identifikující síť je **192.168.1** a část identifikující koncové zařízení je **102**.

Všechna zařízení ve stejné síti mají stejnou první část IP adresy – část identifikující síť. Síť je možné rozdělit na více podsítí tak, že část identifikující síť je rozšířena o několik dalších bitů na úkor části pro koncová zařízení. Celá IP adresa je 32bitová.

Podrobně bude tato problematika probrána později.

Brána

Bránou je obvykle směrovač, který v síti funguje jako hraniční síťové zařízení, jako spojovací článek mezi vnitřní a vnější sítí. Nastavení IP adresy, masky podsítě a výchozí brány můžete provést manuálně přímo v síťovém nastavení počítače nebo je získat z DHCP serveru.

Nastavení síťové adresy, masky podsítě a výchozí brány lze zjistit příkazem **ipconfig**.

Všechny počítače v jedné podsíti mívají nastavenu stejnou adresu výchozí brány. Brána je v téže podsíti jako počítače.

```
Adaptér sítě Ethernet Připojení k místní síti:

   Přípona DNS podle připojení . . . :
   Adresa IPv4 . . . . . . . . . . . : 192.168.0.101
   Maska podsítě . . . . . . . . . . : 255.255.255.0
   Výchozí brána . . . . . . . . . . : 192.168.0.1
```

Výpis IP adresy, masky a brány příkazem ipconfig

Směrování

Když potřebuje počítač komunikovat s jiným počítačem, který není na stejné síti, posílá svá data svému hraničnímu směrovači – výchozí bráně – a ten se postará o další doručení.

Směrovač dělá směrovací rozhodnutí s každým paketem, který do něj dorazí.

Pokud je cílový počítač připojen na jiném rozhraní stejného směrovače jako zdrojový počítač, pak jsou data ihned přesměrována k cíli. Pokud cílová síť není připojena přímo ke směrovači, směrovač přepošle data na další směrovač, který vede k cíli.

Směrovač si vede **směrovací tabulku** (**routovací tabulka**, ***routing table***), v níž má uvedeny cesty do jednotlivých sítí i s metrikou (podle ní může posoudit, která cesta je vhodnější).

Dostane-li paket směrovaný do sítě, kterou má ve své směrovací tabulce, přepne paket na příslušné rozhraní vedoucí do cílové sítě. Pokud dostane paket, jejž má směrovat do sítě, kterou nemá ve své směrovací tabulce, přepne tento paket na rozhraní, jež má definované jako přednastavenou cestu právě pro tento účel. Jestliže taková přednastavená cesta neexistuje a směrovač narazí na síť, kterou nemá ve své směrovací tabulce, pak paket zahodí.

Během své cesty nejsou data v paketu nijak měněna nebo zkoumána. Na směrovači se v síťové vrstvě vždy zkoumá pouze hlavička paketu, ve které se mění jen údaje jako TTL a kontrolní součet hlavičky. Předtím než směrovač může začít zkoumat paket, musí nižší spojová vrstva rozbalit rámec a předat paket do vyšší, síťové vrstvy. Jakmile se směrovač rozhodne, kam paket poslat, předá jej nižší spojové vrstvě, ta přidá informace patřící do rámce a data se vyšlou zvoleným rozhraním.

Směrovač ve své směrovací tabulce vede informace o přímo připojených i vzdálených sítích. Počítače v přímo připojených sítích mají jako svou výchozí bránu nastavenou IP adresu rozhraní směrovače, ke kterému jsou připojeny.

Statické a dynamické směrování

Cesty do vzdálených sítí se buď směrovač naučí dynamicky pomocí směrovacích protokolů od ostatních směrovačů, nebo je můžete staticky manuálně na směrovači nastavit. Výchozí cesta se na směrovači může nastavit staticky manuálně.

Směrovací protokoly, pomocí kterých si směrovače vzájemně předávají informace o sítích a pomocí kterých jsou data směrována, jsou například **protokoly RIP** (***Routing Information Protocol***), **IGRP** (***Interior Gateway Routing Protocol***), **EIGRP** (***Enhanced Interior Gateway Routing Protocol***), **OSPF** (***Open Shortest Path First***).

Pokud jsou na směrovačích povoleny dynamické směrovací protokoly, pak aktualizace informací o stavu sítě probíhají automaticky. To je výhoda oproti statickému směrování v tom ohledu, že při změně v síti dochází u dynamických protokolů k automatické

aktualizaci a neprodlužuje se doba, po kterou by na směrovačích existovaly neaktuální směrovací tabulky. To by způsobovalo nesprávné směrování a možnou ztrátu dat. Aktualizace údajů o síti probíhají v pravidelných intervalech anebo při změnách v síti. Směrovač pak rozešle aktualizované údaje ostatním směrovačům v síti, aby si mohly aktualizovat své informace o síti.

Nevýhodou dynamického směrování je, že během rozesílání aktualizací sítě ostatním směrovačům v síti dochází k přidávání provozu na síť a chvilkově může docházet k zahlcení linky. Následně pak musí směrovače přepočítat údaje ve své směrovací tabulce, což jim také na chvíli zabere čas a prostředky.

Způsoby směrování se mohou kombinovat, mohou být nastaveny statické cesty, přednastavená cesta a na směrovači může být zapnut dynamický směrovací protokol.

Ve směrovací tabulce je u jednotlivých sítí uvedeno rozhraní, kterým se lze do dané sítě dostat, a metrika, jež je závislá na druhu směrovacího protokolu. Obecně udává výhodnost dané cesty.

Pokud se cesta do cílové sítě ve směrovací tabulce nenachází a směrovač má definovanou výchozí cestu, pošlou se data touto výchozí cestou. Není-li výchozí cesta definována, data se zahodí.

Během času se informace o sítích mohou na směrovačích aktualizovat, takže různé pakety jedné komunikace mohou běžně cestovat do cíle různými cestami.

Výpis směrovací tabulky na směrovači může vypadat například takto:

```
RouterA#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
  D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
  N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
  E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
  i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
  * - candidate default, U - per-user static route, o - ODR
  P - periodic downloaded static route

  Gateway of last resort is not set
  10.0.0.0/24 is subnetted, 1 subnets
  C 10.0.0.0 is directly connected, Serial0/0
  S 192.168.2.0/24 is directly connected
  O 172.17.0.0/16 [110/1563] via 172.16.0.1, 00:05:16, Serial0/1
  D 172.17.0.0/16 [90/20514560] via 172.16.0.1, 00:00:55, Serial1/1
```

V seznamu jsou vidět sítě, které jsou přímo připojeny (***directly connected***).

Jsou zde i vzdálené sítě, u nichž je uvedeno rozhraní, přes které se k síti lze dostat (via …), způsob získání informace o síti (**S** – manuálně zadána, **C** – přímo připojená, tj. automaticky se vloží do směrovací tabulky, **O** – naučená prostřednictvím dynamického směrování přes protokol **OSPF**, **D** – naučená prostřednictvím dynamického směrování přes protokol **EIGRP**).

U dynamicky nastavených cest jsou vidět jejich metriky – údaje v závorkách **[ ]**.

Výpis směrovací tabulky na počítači

Na počítači se také vytváří ze síťového provozu směrovací tabulka, kterou lze vypsat příkazem **netstat –r**.

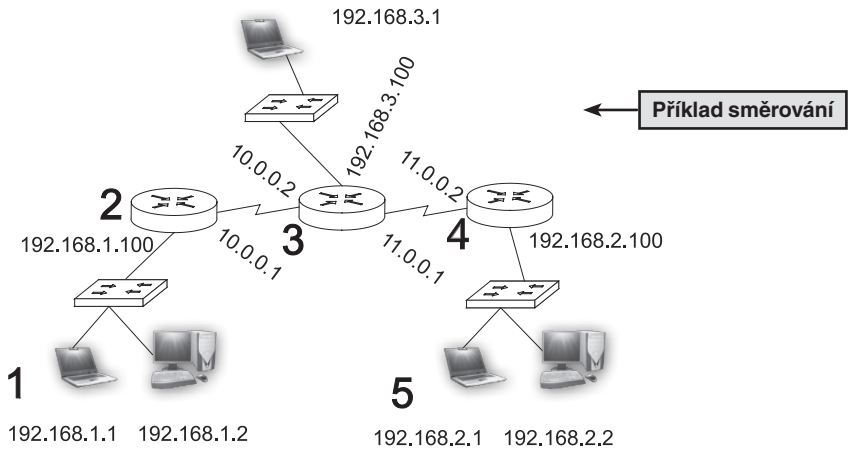
Na obrázku je vidět přednastavená cesta reprezentovaná prvním řádkem výpisu. Ke každé síti, která není lokální, se lze dostat přes bránu (zde **192.168.0.1**).

Lokálnímu rozhraní zvanému **loopback** se přiřazuje adresa **127.0.0.1** nebo jakákoliv ze sítě **127.0.0.0**. Toto rozhraní používají některé interní procesy.

```
IPv4 Směrovací tabulka
===========================================================================
Aktivní směrování:
         Cíl v síti   Síťová maska          Brána       Rozhraní Metrika
          0.0.0.0          0.0.0.0    192.168.0.1   192.168.0.101     20
        127.0.0.0        255.0.0.0         Propojené       127.0.0.1    306
        127.0.0.1  255.255.255.255         Propojené       127.0.0.1    306
  127.255.255.255  255.255.255.255         Propojené       127.0.0.1    306
      192.168.0.0    255.255.255.0         Propojené   192.168.0.101    276
    192.168.0.101  255.255.255.255         Propojené   192.168.0.101    276
    192.168.0.255  255.255.255.255         Propojené   192.168.0.101    276
        224.0.0.0        240.0.0.0         Propojené       127.0.0.1    306
        224.0.0.0        240.0.0.0         Propojené   192.168.0.101    276
  255.255.255.255  255.255.255.255         Propojené       127.0.0.1    306
  255.255.255.255  255.255.255.255         Propojené   192.168.0.101    276
===========================================================================
Trvalé trasy:
  Žádné
```

Výpis směrovací tabulky příkazem *netstat -r*

Ukázka směrování



Příklad směrování

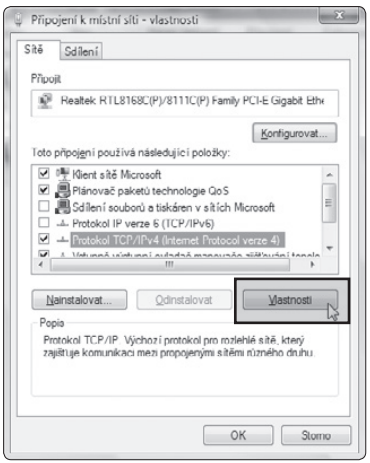
1. Počítač s IP adresou **192.168.1.1** posílá data počítači **192.168.2.1**. Protože cílová adresa není ve stejné síti, posílá data své výchozí bráně, aby zajistila směrování.
2. Směrovač, jehož rozhraní **192.168.1.100** je výchozí bránou zdrojového počítače, vyhodnotí, že cílová síť není k němu přímo připojená, a pošle data následujícímu sousedícímu směrovači na rozhraní **10.0.0.2**, jehož rozhraní má nastaveno jako svou výchozí bránu.
3. Směrovač, na jehož rozhraní **10.0.0.2** přišel paket od předchozího směrovače, zjistí, že cílová síť k němu není přímo připojená, a tak pošle paket následujícímu sousedícímu směrovači na rozhraní **11.0.0.2**, jehož rozhraní má nastaveno jako svou výchozí bránu.
4. Směrovač, na jehož rozhraní **11.0.0.2** přišel paket od předchozího směrovače, zjistí, že cílová síť k němu je připojená, a pošle paket cílovému počítači **192.168.2.1**.

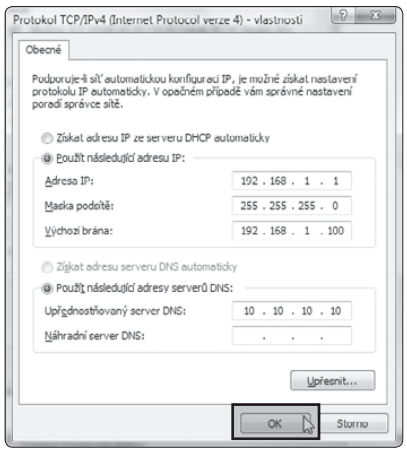
9. Síťové adresy a převody

Síťové nastavení na počítači

Přístup k těmto nastavením se může lišit v závislosti na operačním systému.

V OS Windows Vista se k síťovému nastavení dostanete přes **Ovládací panely** (klasické zobrazení) ▸ **Centrum sítí a sdílení** ▸ **Spravovat síťová připojení**. Zde zvolte přes pravé tlačítko myši **Vlastnosti** zvoleného síťového připojení.





Adresu IP, masku podsítě, výchozí bránu a DNS servery můžete nastavit manuálně napevno nebo si tyto údaje nechat přidělit pomocí DHCP (zaškrtnutím volby **Získat adresu IP ze serveru DHCP automaticky**).

V OS Windows XP se k síťovému nastavení dostanete přes **Ovládací panely** ▸ **Síťová připojení**. Zde zvolte přes pravé tlačítko myši **Vlastnosti** zvoleného síťového připojení.

APIPA

APIPA je **funkce automatického přidělování soukromých adres IP**. Počítač si přidělí sám IP adresu, pokud má nastaveno přidělování pomocí DHCP a na síti nenajde žádný DHCP server.

Pro přidělení adres je stanoven rozsah IP adres **169.254.0.0–169.254.255.255**, maska podsítě 255.255.0.0. Tyto adresy se nesměrují. Pokud směrovač dostane paket směřující do takové sítě, zahodí jej. Rozsah IP adres stanovil úřad **IANA** (***Internet Assigned Numbers Authority***).

Po přidělení takové adresy může počítač komunikovat jen s počítači v lokální síti, které mají nastavenu IP adresu také pomocí funkce **APIPA**.

Někdy může toto přidělení nastat nechtěně (například počítač nebyl ve chvíli startování připojen k síti a nedošlo ke kontaktování DHCP serveru, pak je potřeba se tohoto nastavení zbavit. Po opravení fyzického spojení se sítí můžete docílit přidělení nových síťových nastavení pomocí DHCP serveru buď restartováním počítače, nebo použitím příkazu **ipconfig /release** a **ipconfig /renew**.

Struktura IP adresy verze 4

IP adresa verze 4 je 32bitové číslo. V této podobě s ní pracují síťová zařízení. Pro snadnější pamatování a čtení je zobrazena v lidsky čitelné podobě ve formě čtyř čísel z dekadické soustavy oddělených tečkami. Binární číslo je rozděleno do čtyř skupin po osmi číslicích, tzv. **oktetů**.

Příklad

Binárně: 11000000101010000001100100000001

Dekadicky: **192.168.25.1**

IP adresa se skládá z části, jež je společná všem počítačům na jedné podsíti (tzv. adresa sítě), a dále z části, která identifikuje jednotlivá zařízení.

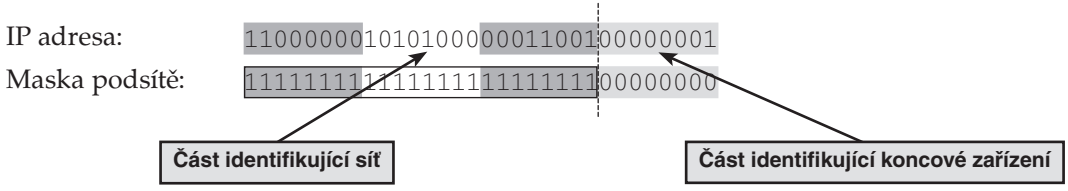
V předchozím příkladu lze rozdělit adresu na část identifikující síť – **192.168.25** a část identifikující koncové zařízení – **1**.

Která část identifikuje síť a která koncové zařízení, můžete zjistit pomocí masky podsítě.

Maska podsítě je také 32bitové číslo, které zleva začíná jedničkami a pokračuje nulami. Jedničky v masce říkají, které bity jsou v IP adrese součástí adresy sítě, nuly pak určují, které bity v IP adrese jsou součástí adresy koncového zařízení.

V předchozím příkladu byla maska číslo 11111111111111111111111100000000 neboli **255.255.255.0**.

Když se napíše v binárním vyjádření pod sebe IP adresa a maska, pak lze snadno určit, která část IP adresy identifikuje síť a je společná všem zařízením v této síti, a která část upřesňuje a jednoznačně identifikuje koncové zařízení.



Číslu 11000000 10101000 00011001 odpovídá dekadicky číslo **192.168.25** – část identifikující síť.

Číslu 00000001 odpovídá dekadicky číslo 1 – část identifikující koncové zařízení.

Dekadická a binární soustava

Následující řádky popisují, co vlastně zápis čísla znamená.

Číslo abc v dekadické soustavě znamená $a \cdot 10^2 + b \cdot 10^1 + c \cdot 10^0$. Symboly abc jsou proměnné zastupující číslice 0 až 9.

10^0 je 1.

Například číslo 987 v dekadické soustavě znamená $9 \cdot 10^2 + 8 \cdot 10^1 + 7 \cdot 10^0$.

Číslo abab v binární soustavě znamená $a \cdot 2^3 + b \cdot 2^2 + a \cdot 2^1 + b \cdot 2^0$. Symboly a, b jsou proměnné zastupující číslice 0 až 1.

2^0 je 1.

Například číslo 11001 v binární soustavě znamená $1 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0$.

Spočítá-li se součet, dostaneme číslo 25 v dekadické soustavě. Tím je naznačen převod z binární do dekadické soustavy.

Konverze z dekadické do binární soustavy

Zadané dekadické číslo je potřeba zapsat jako součet mocnin dvojky.

Číslo se rozkládá na mocniny dvojky, začíná se od maximální mocniny dvojky, kterou číslo obsahuje. Zbytek se opět vyjadřuje jako součet mocnin dvojky, začíná se od maximální mocniny. A tak se postupuje až do zbytku 0.

Převeďte číslo 125 na součet mocnin dvojky.

Do 125 se vejde maximální mocnina dvojky $64 = 2^6$.

Zbytek je $125 - 64 = 61$.

Do čísla 61 se vejde maximální mocnina dvojky $32 = 2^5$.

Zbytek je $61 - 32 = 29$.

Do čísla 29 se vejde maximální mocnina dvojky $16 = 2^4$.

Zbytek je $29 - 16 = 13$.

Do čísla 13 se vejde maximální mocnina dvojky $8 = 2^3$.

Zbytek je $13 - 8 = 5$.

Do čísla 5 se vejde maximální mocnina dvojky $4 = 2^2$.

Zbytek je $5 - 4 = 1$.

Do čísla 1 se vejde maximální mocnina dvojky $1 = 2^0$.

Zbytek je $1 - 1 = 0$.

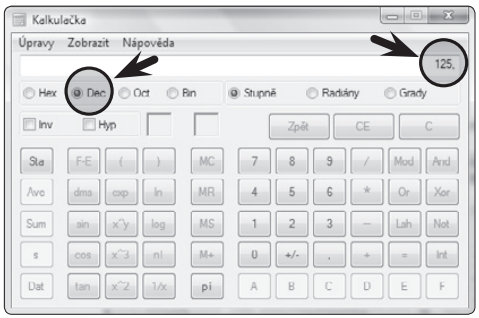
Číslo $125 = 2^0 + 2^2 + 2^3 + 2^4 + 2^5 + 2^6$.

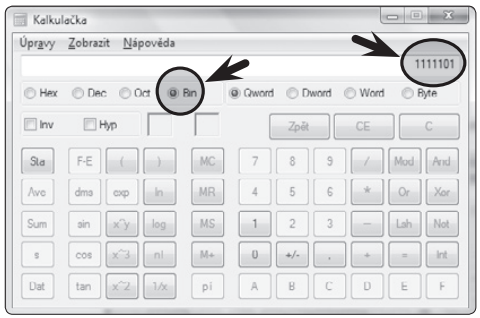
Zápis se seřadí od nejvyšší mocniny a zapíší se i chybějící mocniny jako násobek nuly.

$125 = \mathbf{1}\cdot 2^6 + \mathbf{1}\cdot 2^5 + \mathbf{1}\cdot 2^4 + \mathbf{1}\cdot 2^3 + \mathbf{1}\cdot 2^2 + \mathbf{0}\cdot 2^1 + \mathbf{1}\cdot 2^0$

Nyní se už jen vypíší jedničky a nuly (zvýrazněné výše), vynechají se mocniny dvojky, a tak vznikne zápis čísla v binární soustavě – **1111101**.

Kontrolu lze provést na kalkulačce, která umožňuje převody mezi číselnými soustavami.





IP adresa koncového zařízení, maska podsítě a adresa sítě

Adresa sítě je 32bitové číslo stejně jako IP adresa koncového zařízení. Všechna zařízení v síti mají stejnou adresu sítě. Ve své IP adrese mají stejnou část, která je síťovou částí IP adresy.

Například počítač má nastavenou IP adresu **192.168.25.135**, masku podsítě má **255.255.255.128**.

Adresa sítě se získá z IP adresy a masky jako výsledek logické operace **AND** (**1 AND 1 = 1**, jinak **= 0**).

Binárně zapsáno:

IP adresa:	11000000. 10101000. 00011001. 10000111	
Maska:	11111111. 11111111. 11111111. 10000000	
Adresa sítě:	11000000. 10101000. 00011001. 10000000	Dekadicky: 192.168.25.128

Vzájemná komunikace počítačů v síti

Mohou dva počítače v jedné síti spolu komunikovat (bez pomoci směrovače)? Je-li známa jejich IP adresa a maska podsítě, můžete pomocí logického **AND** zjistit, zda mají oba počítače stejnou adresu sítě. Pokud tomu tak není, nemohou spolu přímo komunikovat. Aby to tak bylo, musí mít stejnou adresu sítě.

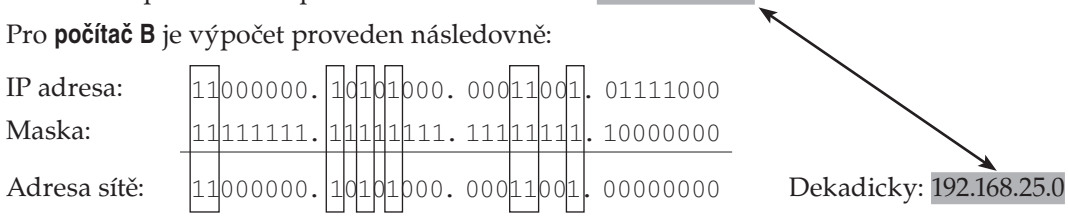
Například **počítač A** má IP adresu 192.168.25.135 a masku podsítě **255.255.255.128**.

počítač B má IP adresu 192.168.25.120 a masku podsítě stejnou – **255.255.255.128**.

Pro zodpovězení otázky, zda mohou oba počítače spolu přímo komunikovat, je potřeba zjistit adresu sítě jednotlivých počítačů.

Počítač A (z předchozího příkladu) má adresu sítě 192.168.25.128.

Pro **počítač B** je výpočet proveden následovně:



Adresy sítí pro počítač **A** a **B** jsou různé, proto spolu počítače nebudou schopny přímo komunikovat.

Unicast, broadcast, multicast

Unicast je vysílání pro určité konkrétní jedno koncové zařízení. Cílová adresa je jednoznačná adresa, například **192.168.1.1**.

Broadcast je vysílání určené všem počítačům v dané lokální síti. Adresa broadcastu (na 3. vrstvě OSI modelu) pro danou síť se skládá z části síťové (ta je stejná pro všechny počítače v dané síti) a části určené pro koncové zařízení, která je v tomto případě vyplněna samými jedničkami (binárně). Tento typ vysílání se používá například v případě, že počítač potřebuje přiřadit adresu síťové vrstvy k adrese spojové vrstvy, proto obesílá všechny počítače v síti, jelikož síťovou adresu prozatím nezná. Broadcast se většinou neposílá ven ze sítě, ve které vznikl, a tím se chrání přenosová kapacita ostatních linek a nenarušuje se práce jiných síťových zařízení v jiných sítích. Hranicí pro broadcast je obvykle směrovač.

Například **adresa sítě** je **192.168.1.0**, **maska podsítě** je **255.255.255.0**. Všechny počítače v této síti sdílejí stejnou část síťové adresy – **192.168.1**. Zbývající číslo je vyhrazeno pro identifikaci konkrétních koncových zařízení. Jeden počítač může mít na tomto místě jedničku, jiný dvojku atd.

Adresa broadcastu je adresa, která na tomto místě má binárně samé jedničky – 11111111, což odpovídá **dekadicky číslu 255**. Celkem tedy **adresa broadcastu** v síti **192.168.1.0** je **192.168.1.255**.

Všechny počítače v dané síti musí paket s touto adresou přijmout a zabývat se jím. Adresa broadcastu je zvláštní typ IP adresy, která se nepřiděluje síťovým zařízením.

Multicast je vysílání pro určitou skupinu počítačů. Využívají jej například směrovače ke vzájemné výměně svých směrovacích tabulek. Existuje soubor IP adres, které jsou určeny pro vysílání tohoto typu. Jsou to adresy v rozsahu **224.0.0.0**–**239.255.255.255**. Zařízení, které chce být účastníkem multicastu, přijme data s cílovou adresou ze zmíněného rozsahu, i když jeho vlastní IP adresa je jiná.

Zápis adresy sítě pomocí prefixu

Aby bylo možno ve zkratce upřesnit, jaká část adresy v IP adrese je adresou sítě, používá se zápis pomocí prefixu. Za IP adresu se za lomítko napíše číslo, které udává délku síťové části adresy v bitech (zleva).

Například zápis **192.168.12.192/26** říká, že prvních 26 bitů v IP adrese mají všechna zařízení v síti stejných, prvních 26 bitů tedy určuje síťovou část adresy. Ostatních 6 bitů zbývá pro adresy koncových zařízení.

Binární zápis adresy **192.168.12.192/26**:

11000000. 10101000. 00001100. 11000000

Prvních 26 bitů je síťová část adresy.

Tento zápis je možné nahradit zápisem IP adresy a masky. IP adresa je stejná – **192.168.12.192** a maska podsítě je **255.255.255.192** (binárně má maska 26 jedniček a 6 nul – 26 jedniček odpovídá číslu za lomítkem v prefixu).

Na zbývajících šesti bitech je možné vytvářet nejrůznější variace jedniček a nul. Na šesti bitech jich lze vytvořit $2^6 = 64$. Teoreticky by bylo možné mít v této síti 64 síťových zařízení. Nicméně to tak úplně není. Dvě adresy jsou výjimečné – samé nuly nebo samé jedničky na těchto posledních šesti bitech. Samé nuly by vytvořily adresu naprosto shodnou s adresou sítě (nepoužívá se), a samé jedničky by vytvořily adresu broadcastu (také se nepoužívá, broadcast je využíván jiným způsobem).

Příklad

Vypište adresy pro koncová zařízení v síti **192.168.10.240/28**.

Řešení

Zapište si adresu binárně:

11000000. 10101000. 00001010. 11110000 (**192.168.10.240**)

/28 znamená, že prvních 28 bitů je část síťová a na zbývajících 4 bitech lze pomocí variací jedniček a nul získat adresy jednotlivých zařízení.

Tyto variace na posledních čtyřech bitech budou vypadat následovně:

1. `0000` – nepoužije se – je to shodné s adresou sítě
2. `0001`
3. `0010`
4. `0011`
5. `0100`
6. `0101`
7. `0110`
8. `0111`
9. `1000`
10. `1001`
11. `1010`
12. `1011`
13. `1100`
14. `1101`
15. `1110`
16. `1111` – nepoužije se – je to shodné s adresou broadcastu

Když k těmto použitelným 14 variacím přidáte předchozích 28 bitů síťové části adresy, dostanete IP adresy pro koncová zařízení.

1. `11000000. 10101000. 00001010. 11110001` – dekadicky **192.168.10.241**
2. `11000000. 10101000. 00001010. 11110010` – dekadicky **192.168.10.242**
3. `11000000. 10101000. 00001010. 11110011` – dekadicky **192.168.10.243**
4. `11000000. 10101000. 00001010. 11110100` – dekadicky **192.168.10.244**
5. `11000000. 10101000. 00001010. 11110101` – dekadicky **192.168.10.245**
6. `11000000. 10101000. 00001010. 11110110` – dekadicky **192.168.10.246**
7. `11000000. 10101000. 00001010. 11110111` – dekadicky **192.168.10.247**
8. `11000000. 10101000. 00001010. 11111000` – dekadicky **192.168.10.248**
9. `11000000. 10101000. 00001010. 11111001` – dekadicky **192.168.10.249**

10. 11000000. 10101000. 00001010. 11111010 – dekadicky **192.168.10.250**
11. 11000000. 10101000. 00001010. 11111011 – dekadicky **192.168.10.251**
12. 11000000. 10101000. 00001010. 11111100 – dekadicky **192.168.10.252**
13. 11000000. 10101000. 00001010. 11111101 – dekadicky **192.168.10.253**
14. 11000000. 10101000. 00001010. 11111110 – dekadicky **192.168.10.254**

Nepoužité adresy jsou: adresa sítě

11000000. 10101000. 00001010. 11110000 – dekadicky **192.168.10.240**

a adresa broadcastu

11000000. 10101000. 00001010. 11111111 – dekadicky **192.168.10.255**

Příklad

Vypište rozsah adres použitelných pro koncová zařízení, adresu sítě a adresu broadcastu. Síť je dána prefixem **172.12.16.0/21**.

Řešení

Zapište si adresu binárně:

10101100. 00001100. 00010000. 00000000 (**172.12.16.0**)

Prvních 21 bitů je síťová část adresy. Zbývajících 11 bitů je část, na které pomocí všech variací jedniček a nul lze získat všechny adresy pro koncová zařízení. Vynechají se opět samé nuly a samé jedničky (adresa sítě a adresa broadcastu). Celkem na 11 bitech lze vytvořit $2^{11} - 2 = 2\,046$ různých variací použitelných pro koncová zařízení. Nebudou zde samozřejmě všechny vypsány.

Následuje výpis jen některých, při výpisu se zachovává oddělení tečkou mezi jednotlivými byty.

1. 000 . 00000001
2. 000 . 00000010
3. 000 . 00000011
255. 000 . 11111111
256. 001 . 00000000
257. 001 . 00000001
2045. 111 . 11111101
2046. 111 . 11111110

Když se tyto adresní části přidají k síťové části adresy, která je 10101100. 00001100. 00010, vzniknou příslušné IP adresy použitelné pro koncová zařízení.

1. 10101100. 00001100. 00010000 . 00000001 – dekadicky **172.12.16.1**
2. 10101100. 00001100. 00010000 . 00000010 – dekadicky **172.12.16.2**
3. 10101100. 00001100. 00010000 . 00000011 – dekadicky **172.12.16.3**

255. 10101100. 00001100. 00010000 . 11111111 – dekadicky **172.12.16.255**

256. 10101100. 00001100. 00010001 . 00000000 – dekadicky **172.12.17.0**

257. 10101100. 00001100. 00010001 . 00000001 – dekadicky **172.12.17.1**

2045. 10101100. 00001100. 00010111 . 11111101 – dekadicky **172.12.23.253**

2046. 10101100. 00001100. 00010111 . 11111110 – dekadicky **172.12.23.254**

Pro koncová zařízení lze v této síti použít adresy

172.12.16.1 – 172.12.16.255, 172.12.17.0 – 172.12.17.255, 172.12.18.0 – 172.12.18.255, 172.12.19.0 – 172.12.19.255, 172.12.20.0 – 172.12.20.255, 172.12.21.0 – 172.12.21.255, 172.12.22.0 – 172.12.22.255, 172.12.23.0 – 172.12.23.254

Adresa sítě je varianta, kdy v posledních 11 bitech jsou samé nuly.

10101100. 00001100. 00010000 . 00000000 – dekadicky **172.12.16.0**

Adresa broadcastu je varianta, kdy v posledních 11 bitech jsou samé jedničky.

10101100. 00001100. 00010111 . 11111111 – dekadicky **172.12.23.255**

Zjištění síťové a broadcast adresy

Ze znalosti IP adresy počítače a masky podsítě lze určit, jak bude vypadat adresa sítě a adresa broadcastu.

Například IP adresa je **10.15.107.25** a maska podsítě je **255.255.248.0**.

IP adresa i maska se zapíše binárně:

10.15.107.25	00001010 . 00001111 . 01101011 . 00011001	
255.255.248.0	11111111 . 11111111 . 11111000 . 00000000	
adresa sítě	00001010 . 00001111 . 01101000 . 00000000	– **10.15.104.0**
adresa broadcastu	00001010 . 00001111 . 01101111 . 11111111	– **10.15.111.255**

Adresa sítě je výsledek operace **AND** provedené na IP adrese a masce (**1 AND 1 = 1**, **1 AND 0 = 0**, **0 AND 1 = 0**, **0 AND 0 = 0**).

Svislá čára ukazuje, kde je hranice mezi síťovou částí adresy a částí specifikující koncové zařízení.

U adresy broadcastu jsou na posledních 11 bitech samé jedničky.

Příklad

Určete adresu sítě, broadcastu, první, čtvrté a desáté použitelné IP adresy v podsíti, pro kterou znáte prefix **172.27.64.64/26**.

Řešení

Zapište adresu a masku binárně:

172.27.64.64	10101100. 00011011. 01000000. 01000000	– adresa podsítě
/26 (maska)	11111111. 11111111. 11111111. 11000000	
broadcast	10101100. 00011011. 01000000. 01111111	– 172.27.64.127

Zadaná adresa 172.27.64.64 je již adresou podsítě, protože v posledních šesti bitech vyhrazených pro identifikaci koncových zařízení obsahuje samé nuly.

Adresy koncových zařízení:

Na šesti bitech vyhrazených pro koncová zařízení lze získat $2^6 - 2 = 62$ různých použitelných variací.

1. 000001 (dekadicky 1)
2. 000010 (dekadicky 2) atd.
4. 000100 (dekadicky 4)
10. 001010 (dekadicky 10)

Tyto variace se přidají k síťové části 10101100. 00011011. 01000000. 01

1. 10101100. 00011011. 01000000. 01000001
4. 10101100. 00011011. 01000000. 01000100
10. 10101100. 00011011. 01000000. 01001010

Výsledek vyjádříte dekadicky a dostanete výsledek.

První použitelná IP adresa je tedy 172.27.64.65, čtvrtá 172.27.64.68, desátá 172.27.64.74.

Příklad

Určete, zda spolu mohou počítače komunikovat bez použití směrování.

Počítač A – 172.27.24.1/19

Počítač B – 172.27.8.10/20

Počítač C – 172.27.1.192/23

Řešení

Vyjádřete IP adresy počítačů binárně:

A: 10101100. 00011011. 00011000. 00000001

B: 10101100. 00011011. 00001000. 00001010

C: 10101100. 00011011. 00000001. 11000000

Čísla za lomítkem určují počet číslic v binárním zápisu, které patří do síťové části adresy. Ostatní bity v adrese sítě jsou vyplněny nulami.

A/19: 10101100. 00011011. 00000000. 00000000 – adresa sítě počítače **A**

B/20: 10101100. 00011011. 00000000. 00000000 – adresa sítě počítače **B**

C/23: 10101100. 00011011. 00000000. 00000000 – adresa sítě počítače **C**

Všechny počítače mají stejnou adresu sítě – můžete ji napsat i dekadicky, tj. **172.27.0.0**, ale to, že je stejná, je vidět i v binárním zápisu. Z tohoto pohledu mohou počítače vzájemně komunikovat.

Příklad

Počítač má IP adresu **172.27.1.63** a masku **255.255.255.192**. Je to adresa vhodná pro koncové zařízení?

Řešení

IP adresu a masku vyjádřete binárně:

IP adresa: 10101100. 00011011. 00000001. 00111111

Maska: 11111111. 11111111. 11111111. 11000000

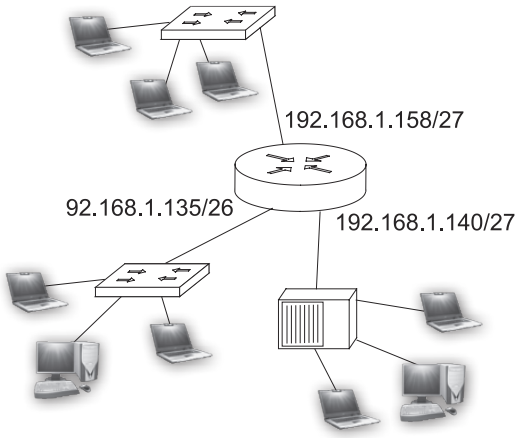
V masce je 26 jedniček, které udávají, že v prvních 26 bitech IP adresy se nachází část identifikující síť a v posledních 6 bitech část identifikující koncové zařízení.

Vidíte, že IP adresa má v posledních 6 bitech samé jedničky. Jak už víte, adresa se samými jedničkami v části identifikující koncové zařízení je vyhrazena pro broadcast.

Taková IP adresa proto není vhodná k adresaci koncového zařízení.

Příklad

Na třech rozhraních směrovače, z nichž každé je výchozí bránou pro jinou síť, byly nakonfigurovány IP adresy **192.168.1.135/26**, **192.168.1.140/27**, **192.168.1.158/27**. Nastane nějaký problém s touto adresací?



Řešení

Protože směrovač slouží ke směrování mezi sítěmi, musí být každé rozhraní směrovače v jiné síti. Ověřte proto, zda se rozhraní směrovače nachází v různých sítích.

192.168.1.135:	11000000. 10101000. 00000001. 10000111
Maska **/26**:	11111111. 11111111. 11111111. 11000000
Adresa sítě:	11000000. 10101000. 00000001. 10000000
192.168.1.140:	11000000. 10101000. 00000001. 10001100
Maska **/27**:	11111111. 11111111. 11111111. 11100000
Adresa sítě:	11000000. 10101000. 00000001. 10000000
192.168.1.158:	11000000. 10101000. 00000001. 10011110
Maska **/27**:	11111111. 11111111. 11111111. 11100000
Adresa sítě:	11000000. 10101000. 00000001. 10000000

Adresy sítí jsou ve všech třech případech naprosto stejné, binárně a samozřejmě i dekadicky (**192.168.1.128**). S takovou adresací nastane problém, protože na směrovači musí mít jednotlivá rozhraní nastaveny takové IP adresy, aby byla v různých sítích. Jinak by nebylo možné provádět směrování.

Směrovač si vede směrovací tabulku, ve které jsou uvedeny sítě a rozhraní, jež do sítí vedou, a podle toho pak rozhoduje, kterým rozhraním zasílaná data poslat. Tady by měl stejnou síť připojenou ke třem různým rozhraním. To by vedlo ke zmatku a nefunkčnosti směrovače.

10. Třídy IP adres, privátní a veřejné adresy, rezervované adresy

Rezervované IP adresy

Na 32 bitech je teoreticky možno vytvořit až 2^{32} = 4 294 967 296 různých IP adres, ale ve skutečnosti se některé adresy nepoužívají. Jak je vidět, tento rozsah by v dnešní době i tak nestačil (nedostatek IP adres se řeší jinými způsoby, které budou popsány později).

Pro komunikaci typu unicast se vynechávají následující adresy:

- **224.0.0.0–239.255.255.255 – používají se pro multicast**
- **240.0.0.0–255.255.255.254 – používají se pro výzkum**

Třídy IP adres

Třída A

Síťová část IP adresy je zastoupena 8 bity, zbývajících 24 bitů je určeno k identifikaci koncových zařízení. Pro zvolenou síťovou adresu je tedy k dispozici 2^{24} různých variací jedniček a nul, což může adresovat přibližně 16 milionů IP adres ve zvolené síti.

Je tedy zřejmé, že sítě s IP adresou třídy **A** jsou vybaveny IP adresami pro obrovské množství koncových zařízení. Obvykle je takový adresní prostor dále dělen do menších pomocí podsíťování.

Na první pohled lze IP adresu třídy **A** poznat tak, že v binárním zápisu vždy začíná nulou.

0sssssss . kkkkkkkk . kkkkkkkk . kkkkkkkk

s – označuje bity vyhrazené síťové části IP adresy, **k** – označuje bity vyhrazené k identifikování koncových zařízení

K úvodní nule lze v síťové části adresy přiřadit doplňujících 7 číslic – jedniček a nul. Pokud by byly doplněny samé nuly, vznikla by adresa začínající dekadicky nulou – **0.x.x.x**.

Adresy začínající nulou se nepoužívají – je to **adresa přednastavené cesty** (***default route***).

Pokud by k úvodní nule byly doplněny samé jedničky, vznikla by adresa začínající dekadicky číslem 127 – **127.x.x.x**.

Tak začíná **adresa loopbacku** – vnitřního rozhraní počítače. Slouží k testování správné konfigurace TCP/IP protokolu na počítači a vnitřních procesů počítače. Ani tato adresa se nepoužívá.

Síťovou část adresy lze tedy doplnit jakoukoliv kombinací jedniček a nul mimo samé jedničky a samé nuly.

00000001–011111110 – dekadicky 1–126

IP adresa třídy A začíná číslem z rozsahu 1–126.

Maska podsítě IP adresy třídy A je 255.0.0.0.

Třída B

Síťová část IP adresy je zastoupena 16 bity, zbývajících 16 bitů je určeno k identifikaci koncových zařízení. Pro zvolenou síťovou adresu je tedy k dispozici 2^{16} různých variací jedniček a nul, což může adresovat přibližně 65 tisíc IP adres ve zvolené síti.

Síť třídy **B** je vhodná pro střední až velké množství počítačů.

Na první pohled lze IP adresu třídy **B** poznat tak, že v binárním zápisu vždy začíná jedničkou a nulou.

10ssssss . ssssssss . kkkkkkkk . kkkkkkkk

s – označuje bity vyhrazené síťové části IP adresy, **k** – označuje bity vyhrazené k identifikování koncových zařízení

K úvodní jedničce a nule lze v síťové části adresy přiřadit doplňujících 14 číslic – jedniček a nul.

Na prvním bytu lze doplnit číslo jakoukoliv kombinací jedniček a nul – od **000000** až po **111111**.

10000000–101111111 – dekadicky **128–191**.

IP adresa třídy B začíná číslem z rozsahu 128–191.

Maska podsítě IP adresy třídy B je 255.255.0.0.

Třída C

IP adresa se skládá z části síťové, která je tu zastoupena 24 bity, zbývajících 8 bitů je určeno k identifikaci koncových zařízení. Pro zvolenou síťovou adresu je tedy k dispozici 2^8 různých variací jedniček a nul, což může adresovat 254 IP adres ve zvolené síti (jak již bylo řečeno dříve, pokud by se část určená k identifikaci koncových zařízení vyplnila binárně samými nulami, vznikla by adresa stejná, jako je adresa celé sítě, a vyplnění samými jedničkami by bylo stejné jako adresa broadcastu v dané síti – proto se tyto dvě adresy pro adresování koncových zařízení nepoužívají).

Síť třídy **C** je vhodná pro malé sítě.

Na první pohled lze IP adresu třídy **C** poznat tak, že v binárním zápisu vždy začíná dvěma jedničkami a nulou.

110sssss . ssssssss . ssssssss . kkkkkkkk

s – označuje bity vyhrazené síťové části IP adresy, **k** – označuje bity vyhrazené k identifikování koncových zařízení

K úvodním dvěma jedničkám a nule lze v síťové části adresy přiřadit doplňujících 21 číslic – jedniček a nul. To dává k dispozici 2^{21} různých sítí.

Na prvním bytu lze doplnit číslo jakoukoliv kombinací jedniček a nul – od **00000** až po **11111**.

11000000–11011111 – dekadicky **192–223**.

IP adresa třídy C začíná číslem z rozsahu 192–223.

Maska podsítě IP adresy třídy C je 255.255.255.0.

Třída D

Tato třída je určena pro vysílání typu **multicast** – vysílání pro předem určenou skupinu zařízení. Pomocí multicastu si například směrovače mohou vyměňovat informace o topologii sítě.

Binárně začíná IP adresa třídy **D** třemi jedničkami a nulou.

1110xxxx . xxxxxxxx . xxxxxxxx . xxxxxxxx

Na prvním bytu lze doplněním zbývajících čtyř bitů jedničkami a nulami získat varianty od **11100000–11101111**, což je dekadicky **224–239**.

IP adresy třídy D začínají číslem z rozsahu 224–239.

Třída E

IP adresy z této třídy se používají pro výzkumné účely. K adresování běžných zařízení se nepoužívají.

Binárně začíná IP adresa třídy **E** čtyřmi jedničkami.

1111xxxx . xxxxxxxx . xxxxxxxx . xxxxxxxx

Na prvním bytu lze doplněním zbývajících čtyř bitů jedničkami a nulami získat varianty od **11110000–11111111**, což je dekadicky **240–255**.

IP adresy třídy E začínají číslem z rozsahu 240–255.

Privátní IP adresy

Pro směrování na veřejné síti se musí používat unikátní, jednoznačné IP adresy. Na veřejné síti se IP adresy nesmí duplikovat. To by vedlo ke konfliktům těchto adres a zmatku ve směrování.

Existuje ale jasně definovaná skupina adres (**privátní adresy**), které se nepoužívají na veřejné síti, ale používají se v sítích privátních, které jsou od veřejné sítě odděleny.

Ve světě může existovat bezpočet privátních sítí, které používají stejné adresy, ale protože jsou pro veřejnou síť skryté, nevadí to. Tyto privátní sítě jsou skryté za nějaké hraniční zařízení, obvykle směrovač, který těmto sítím zprostředkovává přístup k internetu. Počítače takové privátní sítě pak mohou přistupovat k internetu a komunikovat s ostatními počítači veřejné sítě prostřednictvím svého hraničního zařízení. Skrývají se za jeho IP adresou a jejich privátní IP adresa v síťové komunikaci nefiguruje.

V třídě **A** je stanovena privátní adresa **10.x.x.x /8**.

Počítače v této privátní síti mohou mít adresy **10.0.0.1–10.255.255.254** (**10.0.0.0** – adresa sítě, **10.255.255.255** – adresa broadcastu).

V třídě **B** je stanoveno 16 privátní sítí – **172.16.0.0–172.31.0.0 /16**.

Každá z nich má adresní rozsah **x.y.0.0–x.y.255.255** (**x.y.0.0** – adresa sítě, **x.y.255.255** – broadcast).

Tento adresní blok lze shrnout do jednoho zápisu **172.16.0.0/12**.

10101100. 00010000. 00000000. 00000000 – na zbývajících 20 bitech získáte kombinacemi jedniček a nul všechny adresy definované jako privátní **od 172.16.0.0 do 172.31.255.255**.

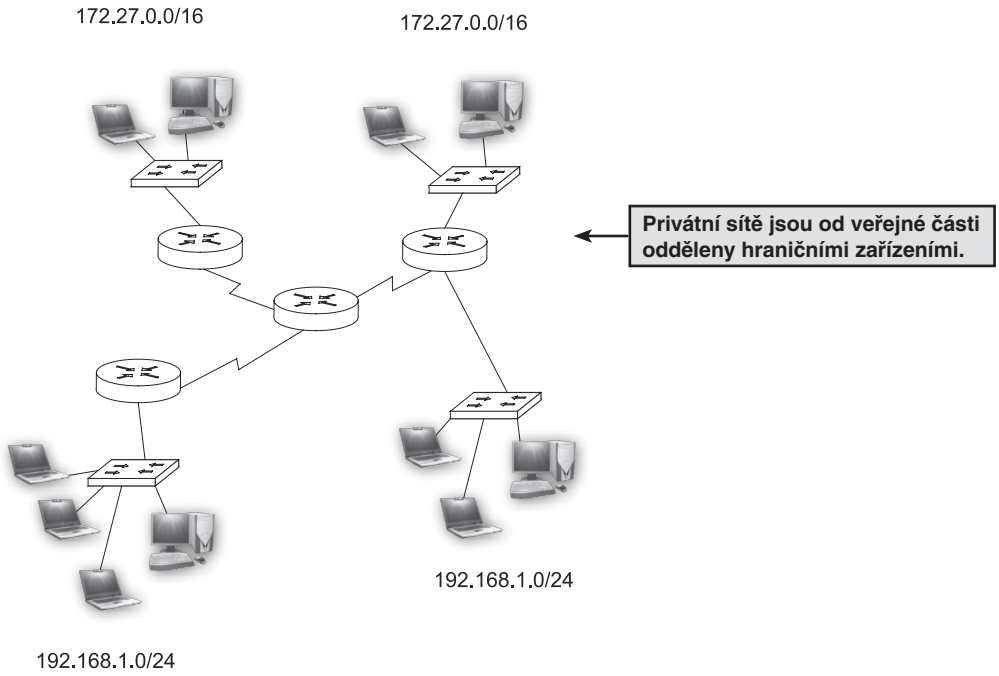
V třídě **C** je stanoveno 256 privátních sítí – **192.168.0.0–192.168.255.0/24**.

Každá z těchto privátních sítí má adresní rozsah **x.y.z.0–x.y.z.255** (**x.y.z.0** – adresa sítě, **x.y.z.255** – broadcast).

Tento rozsah lze shrnout do jednoho zápisu **192.168.0.0/16.**

11000000. 10101000. 00000000. 00000000 – na zbývajících 16 bitech získáte kombinacemi jedniček a nul všechny adresy definované jako privátní **od 192.168.0.0 do 192.168.255.255**.

Na obrázku je znázorněna sada privátních sítí, jejichž adresní rozsahy mohou být stejné, důležité je, aby navenek vystupovaly pod unikátní veřejnou adresou, kterou často zajišťuje hraniční směrovač těchto sítí.



NAT

NAT – ***Network Address Translation***

Jedná se o překlad mezi interní privátní adresou a veřejnou adresou. Jestliže chce počítač v interní privátní síti přistupovat ke zdrojům na veřejné síti, například na internet, musí na hraničním zařízení, kterým je obvykle směrovač, dojít k překladu adres.

Počítač vyšle dotaz směřovaný na vnější zařízení umístěné na internetu. Hraniční směrovač musí zajistit, aby se v posílaných datech nevyskytovala jako odchozí IP adresa privátní adresa počítače, protože odpověď na tento dotaz by nebyla v síti Internet směrována a pakety s privátní adresou cíle by byly zahozeny.

Směrovač nahradí zdrojovou privátní adresu veřejnou IP adresou, která je již v síti Internet směrovatelná.

Směrovač může mít k dispozici jednu nebo více veřejných IP adres, které takto půjčuje a pomocí kterých zaměňuje privátní adresy v komunikaci za tyto veřejné. Obvykle nemá k dispozici tolik veřejných IP adres, jako má privátních IP adres ve své privátní síti. Pak musí jednotlivé komunikace odlišit ještě pomocí přidělování portů. V tom případě se již jedná o **NAT** s podporou **PAT** – ***Port Address Translation***.

Odchozí komunikace a příchozí odpovědi z veřejné sítě jsou pak mapovány pomocí čísel portů, podle nichž směrovač dokáže odlišit a rozhodnout, kterému počítači kterou odpověď poslat.

Přednastavená cesta

Jak již bylo napsáno v části o IP adresách třídy **A**, adresa typu **0.x.y.z** se pro zařízení v síti nepoužívá, je to adresa přednastavené cesty, kterou se posílají data, u nichž síťové zařízení přesně neví, kam je poslat. Jsou to všechny adresy, které lze zapsat souhrnně jako **0.0.0.0/8**.

Anglicky se označuje jako ***default route***.

Správci IP adres

Hlavním správcem IP adres je organizace **IANA** - ***Internet Assigned Numbers Authority*** spolu s **ICANN** - ***Internet Corporation for Assigned Names and Numbers***. Do 90. let 20. století **IANA** spravovala a přidělovala IP adresy sama, ale později tuto činnost delegovala na další společnosti. Zbývající prostor IP adres verze 4 byl rozdělen mezi tyto organizace, tzv. **regionální internetové registry** – **RIR**. Pokud regionální registr potřebuje další adresní prostor, je mu přidělena část z adresního prostoru IP verze 6.

Hlavní registry jsou:

- **AfriNIC** – pro region Afrika
- **APNIC** – pro region Asie a Tichého oceánu
- **ARIN** – pro region Severní Ameriky
- **LACNIC** – pro region Latinské Ameriky a některých karibských ostrovů
- **RIPE NCC** – pro region Evropy, Středního východu a střední Asie

Poskytovatel připojení k internetu

ISP (***Internet Service Provider***) – poskytovatel připojení k internetu

Když se uživatel potřebuje připojit k internetu, požádá některého z poskytovatelů. Ten mu může přidělit potřebný počet veřejných IP adres, pomocí kterých se lze do veřejné sítě připojit.

Pokud přejde k jinému poskytovateli, vrátí předchozímu poskytovateli poskytnuté IP adresy a dostane od nového poskytovatele nové adresy z jeho rozsahu.

Poskytovatelé internetu zajišťují svým zákazníkům většinou i další služby, jako jsou DNS služby, webové služby, e-mail.

Poskytovatelé internetu se dělí podle hierarchického připojení k síti do skupin. Nejvyšší skupina je připojena přímo k páteřní síti, a poskytuje tak svým zákazníkům vysokorychlostní připojení k síti a odpovídající služby, především spolehlivost přístupu k síti.

Druhá skupina získává své připojení od poskytovatele internetu na první úrovni. Nabízí svým zákazníkům obvykle více služeb než poskytovatelé první úrovně. Jsou to například DNS služby, e-mailové servery, webové servery, výroba a údržba webových stránek, internetové obchody, telefonní služby přes internet. Jelikož je druhá skupina připojena k internetu zprostředkovaně pomocí první skupiny, není již rychlost připojení a jeho spolehlivost tak vysoká jako u první skupiny.

Druhá skupina poskytovatelů může podporovat poskytovatele internetu třetí úrovně, kteří se již nezabývají rychlostí ani spolehlivostí připojení, ale jen základním připojením a podporou. I tyto horší služby od poskytovatelů internetu v třetí úrovni jsou často vhodnou volbou vzhledem ke své ceně.

11. IP verze 6

Protokol IP verze 4 umožňuje adresovat přibližně 4 miliardy zařízení. Na světě však dnes existuje již více počítačů, proto byl v 90. letech 20. století vyvinut nástupce protokolu IP verze 4 – **protokol IP verze 6**. Protože se pro úsporu veřejných IP adres používají privátní IP adresy a podsíťování, nejsou ještě všechny IP adresy verze 4 zcela vyčerpány, nicméně registrátoři odhadují vyčerpání adres během pár let.

Při vytváření nového internetového protokolu verze 6 bylo vzato v úvahu hned několik vylepšení.

Protokol IPv6 umožňuje 128bitové adresování. Ve srovnání s protokolem **IPv4**, který nabízí adresování 32bitové, je 128bitové adresování velkým rozšířením. Na 32 bitech lze získat až 2^{32} různých adres, ale na 128 bitech až 2^{128} různých adres. To je oproti přibližně 4 miliardám vzestup na přibližně $3{,}4 \cdot 10^{38}$ (což je nárůst o 29 řádů).

To do budoucna zajistí více než **dostatečné množství IP adres** pro počítače a síťová zařízení na celém světě.

Zjednodušená struktura hlavičky paketu **IPv6** umožňuje snadnější manipulaci a vyhodnocování paketu a kontrolu kvality přenosu pomocí označování paketů určitými příznaky.

Protokol **IPv6 je bezpečnější než IPv4**, protože již ve své struktuře má integrované bezpečnostní prvky a možnost ověřování.

Prozatím zůstává v sítích stále nejrozšířenější protokol **IPv4**, přestože teoreticky již dávno nestačí pokrýt adresní potřeby počítačů na světě. Nedostatek IP adres se řeší převážně pomocí privátních sítí, ve kterých se adresy mohou opakovat a navenek vystupují pod jednou nebo více veřejnými adresami. Dále se adresní prostory úsporně využívají pomocí podsíťování.

V některých sítích je již protokol **IPv6** implementován a do budoucna je připraven stát se následníkem **IPv4**.

Někdy se nová podoba protokolu **IPv6** setkává s kritikou kvůli délce IP adresy. Ta umožňuje adresovat astronomické množství síťových zařízení, které podle kritiků neúměrně přesahuje potřeby. Navíc se tato dlouhá IP adresa oproti relativně krátké IP adrese verze 4 špatně pamatuje a k pojmenování zařízení jsou již třeba jmenné DNS překlady.

Přesto je to schopný následník IP adresy verze 4 a v brzké době bude jednou z mála alternativ jak v sítích adresovat síťová zařízení.

Struktura paketu IPv6

IP paket verze 6 se skládá z 320bitové hlavičky a těla. Hlavička obsahuje verzi, dopravní třídu, pojmenování toku, délku těla, následující hlavičku, limit přeskoků, zdrojovou a cílovou IP adresu.

Verze 6 (4 bity)	Dopravní třída (8 bitů)	Pojmenování toku (20 bitů)
Délka těla (16 bitů)	Následující hlavička (8 bitů)	Limit přeskoků (8 bitů)
Zdrojová adresa (128 bitů)		
Cílová adresa (128 bitů)		

Verze (angl. ***Version***) je číslo verze protokolu IP, je nastavena na 6.

Dopravní třída (angl. ***Traffic class***) určuje prioritu paketu nebo zařazení do určité přepravní třídy. Umožňuje zajistit služby s požadovanou kvalitou.

Pojmenování toku (angl. ***Flow label***) – zde může být specifikováno, zda je s určitými pakety potřeba zacházet přednostněji než s jinými. Tato část se používá pro služby kontroly kvality přenosu. Podle této značky dokáže směrovač poznat, že určitý paket patří k určité skupině paketů, se kterými je potřeba nakládat stejným způsobem. Využití tohoto políčka je stále ve vývoji.

Délka těla (angl. ***Payload length***) označuje délku paketu následujícího po hlavičce, počítáno po bytech.

Následující hlavička (angl. ***Next header***) určuje další vnořený protokol. Označuje typ hlavičky neprodleně následující po hlavičce **IPv6**. Používá stejné hodnoty jako políčka v protokolu **IPv4**.

Limit přeskoků (angl. ***Hop limit***) – ze své výchozí hodnoty je snížen vždy o 1, jakmile projde přes uzel typu směrovač. Jakmile je hodnota na nule, paket se zahodí.

Hlavička neobsahuje kontrolní součet jako paket **IPv4**, tuto kontrolu přenechává spojové vrstvě.

Zdroj a kompletní popis struktury paketu IPv6 v angličtině:
http://www.ietf.org/rfc/rfc2460.txt?number=2460

Zápis IP adresy verze 6

IP adresa verze 6 je 128bitová a pro jednodušší zápis se používá zápis v hexadecimální soustavě. Zapisuje se jako osm skupin čtyř hexadecimálních číslic, skupiny jsou odděleny dvojtečkami. Nuly zleva se mohou vynechávat.

Příklad IP adresy verze 6 je **1001:ab21:cd:af:1112:35:ff:ab28**.

Jestliže je některá skupina číslic složená ze samých nul, můžete ji celou vynechat.

Například: **1fab::23c:ffab::98de::ab02**.

Typy adres

- **Unicast (individuální adresa)** je přidělena jednomu konkrétnímu síťovému rozhraní. Data posílaná na tuto adresu jsou doručena právě jen tomuto síťovému rozhraní.
- **Multicast (skupinová adresa)** slouží k adresování skupiny síťových zařízení. Data posílaná na tuto adresu dorazí všem členům skupiny.
- **Anycast (výběrová adresa)** je nový typ adresy, který označuje skupinu síťových zařízení. Data poslaná na tuto adresu však dorazí jen k nejbližšímu členovi této skupiny. Význam těchto adres je například v rozkládání zátěže, zrychlení doby odezvy od serveru směrem ke klientovi, v ochraně proti útokům majícím za cíl zahltit určitou adresu a ve zmenšení počtu adres, které danou službu poskytují. Využití nachází například v adresování kořenových DNS serverů, kdy se za několika adresami schovává ve skutečnosti mnohem větší počet strojů, které poskytují tyto služby.

Počítače, které jsou součástí jedné sítě, mají stejný začátek adresy podobně jako u IP verze 4. Adresa sítě se vyjadřuje pomocí IP adresy a **prefixu** (**číslo za lomítkem označující počet bitů zleva, které jsou společné všem adresám v dané podsíti**).

Individuální adresy jsou celosvětově unikátní a jsou přidělovány hierarchicky. Poskytovatel internetu dostane přidělen určitý adresní rozsah definovaný IP adresou s určitým prefixem a z této adresy pak vytváří prodloužením prefixu další podsítě. Všechny jeho podsítě mají stejný začátek, což je důležité z hlediska slučování adres se stejným začátkem do jednoho společného zápisu, který si pak vedou směrovače ve svých směrovacích tabulkách. Nemusí si tak vést informace o všech podsítích, stačí jim vést si záznam o této sloučené – agregované – adrese.

Celou adresu lze rozdělit na část identifikující hlavní síť – obvykle 48 bitů, část specifikující podsíť – obvykle 16 bitů a část identifikující síťové rozhraní – posledních 64 bitů.

Problematika IPv6 je velmi obsáhlá a nebude zde dopodrobna probrána.

12. Maska podsítě

Maska podsítě pro protokol **IPv4** je 32bitové číslo, pomocí něhož se rozlišuje, které bity v IP adrese identifikují síť a které síťové zařízení.

Toto číslo se pro přehlednost rozděluje do čtyř skupin po osmi bitech, skupiny se oddělují tečkou. V dekadické podobě se čísla také oddělují tečkou.

Maska se ve své binární podobě skládá ze spojité řady jedniček následované spojitou řadou nul. Jedničky v masce znamenají, že na stejných pozicích se v IP adrese nacházejí identifikátory sítě. Nuly v masce určují, na kterých bitech se v IP adrese nacházejí identifikátory síťového zařízení.

Jestliže je maska číslo například **11111111.00000000.00000000.00000000**, pak prvních osm bitů, na kterých se vyskytují jedničky, říká, že v příslušné IP adrese patří prvních osm bitů síti a zbývajících 24 bitů patří identifikaci síťového zařízení.

Konkrétně – **k IP adrese** například **185.162.1.24** přísluší **maska 255.255.0.0**. Ta značí, že část IP adresy **185.162** patří k síti, ve které se zařízení s touto IP adresou nachází, a část **1.24** identifikuje jednoznačně toto síťové zařízení.

Pokud je maska takto jednoduchá, lze síťovou část IP adresy a část identifikující síťové zařízení okamžitě odhadnout.

Jestliže je maska komplikovanější, je potřeba IP adresu i masku zapsat binárně, a pak je vidět, na kterých bitech v masce jsou jedničky identifikující síťovou část IP adresy a kde jsou nuly, jež identifikují část patřící síťovému zařízení.

Například – **k IP adrese 185.162.152.24** přísluší **maska 255.255.192.0**. Aby bylo možno určit, jak vypadá část identifikující síť (tato část adresy je pro všechny počítače v dané síti společná) a jak část identifikující síťové zařízení, je v tomto případě nutné provést rozpis do binární podoby.

IP adresa: `10111001. 10100010. 10011000. 00011000`

Maska: `11111111. 11111111. 11000000. 00000000`

Je vidět, že v masce jsou jedničky na prvních 18 bitech. Proto se síťová část IP adresy nachází v IP adrese také na prvních 18 bitech.

Síťová část IP adresy tedy obsahuje bity `10111001. 10100010. 10`.

Zbývající bity jsou určeny k jednoznačné identifikaci síťového zařízení – `011000. 00011000`.

Pro vyjádření adresy sítě dekadicky je třeba doplnit v části pro síťová zařízení nuly a výsledné binární číslo převést do desítkové soustavy.

Adresa sítě: `10111001. 10100010. 10000000. 00000000` – **185.162.128.0**.

Masky v třídách IP adres

Pro jednotlivé třídy IP adres se používají přednastavené masky podsítě.

Třída IP adres	Maska podsítě dekadicky	Maska podsítě binárně
Třída A	**255.0.0.0**	11111111.00000000.00000000.00000000
Třída B	**255.255.0.0**	11111111.11111111.00000000.00000000
Třída C	**255.255.255.0**	11111111.11111111.11111111.00000000

Třídy IP adres skýtají různě velký adresní prostor pro síťová zařízení. Ten lze využít buď efektivně, kdy nezbude příliš velké množství nevyužitých adres, nebo neefektivně, kdy těchto adres zůstane k dispozici neúměrně mnoho.

Pokud není potřeba využít adresní prostor efektivně a bez velkého počtu nevyužitých adres, lze použít tyto základní masky podsítě.

Pokud ale adresní rozsah umožňuje adresovat velké množství síťových zařízení, a přitom je třeba přiřadit IP adresu jen malému množství počítačů, je možné masku podsítě rozšířit na úkor části identifikující síťová zařízení, a tím zmenšit počet adresovatelných síťových zařízení.

Tato problematika bude vysvětlena v následující části o podsítích.

Rozšíření masky

Jestliže zvýšíte počet jedniček v binárním vyjádření masky na úkor části pro síťová zařízení, dostanete následující vzorky jedniček a nul, které lze v desítkové soustavě vyjádřit takto:

- 10000000 – **128**
- 11000000 – **192**
- 11100000 – **224**
- 11110000 – **240**
- 11111000 – **248**
- 11111100 – **252**
- 11111110 – **254**

Maska zapsaná prefixem

Jak již bylo zmíněno dříve, k symbolickému znázornění masky podsítě se používá zápis označovaný jako **prefix**. **Prefix je číslo za lomítkem označující zleva počet jedniček v masce podsítě.**

Například – **masku 255.248.0.0** lze prefixem zapsat jako **/13**.

Masku podsítě 255.255.255.192 lze prefixem zapsat jako **/26**.

Adresa sítě

IP adresa definovaná na síťovém zařízení spolu s maskou podsítě určují jednoznačně síť, do které toto síťové zařízení patří.

Například IP adresa síťového zařízení je **192.168.2.3**, maska podsítě je **255.255.255.0**. Maska obsahuje 24 jedniček, což je počet bitů v IP adrese, které identifikují síť tohoto síťového zařízení.

Adresa IP vyjádřena binárně je 11000000. 10101000. 00000010. 00000011. Prvních 24 bitů je 11000000. 10101000. 00000010, tato část identifikuje síť. Zbytek (8 bitů) se vyplní samými nulami a takto společně se vyjádří adresa sítě. 11000000. 10101000. 00000010. 00000000 – **192.168.2.0**.

Zařízení v síti

Každá síť má určitý počet IP adres, které lze přiřadit síťovým zařízením.

Přesný počet lze zjistit z masky podsítě.

Jestliže maska je dána prefixem **/24**, pak pro síťová zařízení zbývá osm bitů, na kterých lze pomocí variant jedniček a nul získávat různé IP adresy. Na osmi bitech lze takto získat $2^8 = 256$ různých variant, které spolu se síťovou částí IP adresy dávají dohromady IP adresy síťových zařízení.

Například – **IP adresa sítě** je **192.168.2.0**, **maska** je dána prefixem **/24**. Zbývá 8 bitů pro síťová zařízení. Na těch lze získat varianty: 00000000, 00000001, 00000010, 00000011, 00000100, …, 11111110, 11111111.

To jsou v desítkové soustavě čísla **0–255**.

Jestliže spojíte síťovou část IP adresy **192.168.2** s těmito čísly, dostanete IP adresy síťových zařízení v této síti.

192.168.2.0, 192.168.2.1, 192.168.2.2, 192.168.2.3, …, 192.168.2.254, 192.168.2.255.

Jak už víte, první a poslední varianta se k adresování síťových zařízení nepoužívá, protože první adresa **192.168.2.0** je shodná s adresou sítě a poslední adresa **192.168.2.255** je shodná s adresou broadcastu.

Logický součin

Pro připomenutí následuje tabulka ukazující, jak vypadá logický součin označovaný jako **AND**.

AND	0	1
0	0	0
1	0	1

Logický součin dvou hodnot je téměř vždy nula, jen v případě **1 AND 1** vychází **1**.

Logický součin se používá v síťové matematice například pro zjištění adresy sítě z IP adresy síťového zařízení a masky podsítě.

Určete adresu sítě

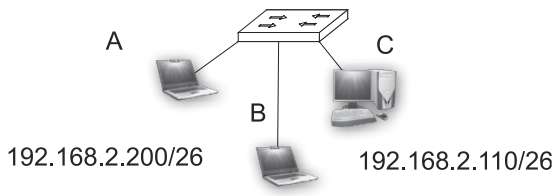
Z IP adresy počítače **172.27.141.25 s maskou** podsítě zadanou prefixem **/20** zjistěte, do jaké sítě počítač patří.

Vyjádřete IP adresu a masku binárně a proveďte logický součin mezi nimi. Logický součin se provádí na úrovni jednotlivých bitů. Je vhodné adresy napsat pod sebe tak, aby bylo vidět, který bit se kterým bitem budou členy operace **AND**.

IP adresa:	10101100. 00011011. 10001101. 00011001
Maska:	11111111. 11111111. 11110000. 00000000
AND:	10101100. 00011011. 10000000. 00000000

Výsledek logického součinu zapsaný na předchozím řádku je adresa sítě, do které počítač patří. V desítkové soustavě lze adresu sítě zapsat jako **172.27.128.0**.

Zjistěte, zda jsou počítače A, B a C ve stejné síti.



A – 192.168.2.200: 11000000. 10101000. 00000010. 11001000

B – 192.168.2.150: 11000000. 10101000. 00000010. 10010110

C – 192.168.2.110: 11000000. 10101000. 00000010. 01101110

Všechny počítače mají masku zadanou prefixem **/26**.

Prvních dvacet šest bitů u všech počítačů označuje síťovou část adresy. Pro získání adresy sítě je třeba doplnit zbývajících šest bitů nulami.

Síť počítače **A**: 11000000. 10101000. 00000010. 11000000

Síť počítače **B**: 11000000. 10101000. 00000010. 10000000

Síť počítače **C**: 11000000. 10101000. 00000010. 01000000

I v tomto binárním vyjádření je vidět, že každá síť je jiná, a to se nezmění ani po převedení čísel sítí do desítkové soustavy.

Síť počítače **A**: **192.168.2.192**

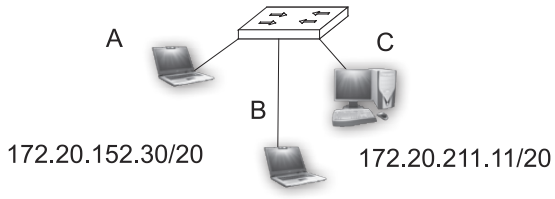
Síť počítače **B**: **192.168.2.128**

Síť počítače **C**: **192.168.2.64**

Každý počítač je v jiné síti, a to i přesto, že jsou připojeny na jeden přepínač (switch). Protože jsou v jiných sítích, nebudou spolu bez pomoci směrovače (routeru) komunikovat.

Cvičení – určení čísel sítí počítačů

Určete, zda jsou počítače **A**, **B** a **C** v jedné síti.



13. Základy vytváření podsítí

Princip

Pomocí podsíťování lze z konkrétního bloku adres z jedné sítě vytvořit několik různých podsítí, z nichž každá bude mít svůj vlastní adresní prostor. Směrovač (router) bude schopen směrovat provoz mezi sítěmi.

Jak už bylo zmíněno dříve, každé rozhraní na směrovači musí mít IP adresu nastavenou tak, aby každá z nich byla v jiné síti, jinak by směrovač nedokázal řídit provoz a přepínat pakety na správná rozhraní do správných sítí.

Podsítě se z dané sítě vytvářejí tak, že se maska podsítě prodlouží o několik bitů na úkor části, která je vyhrazena pro identifikaci síťových zařízení.

Na každém „půjčeném" bitu je možné zapsat buď jedničku, nebo nulu, každým „půjčeným" bitem se zdvojnásobí počet získaných podsítí. Je zřejmé, že každé podsíti zbude méně bitů pro identifikaci síťových zařízení, a tím tedy bude možné vytvořit méně IP adres pro tato zařízení.

Jestliže se maska sítě prodlouží o jeden bit, získáte dvě podsítě, jestliže se prodlouží o dva bity, získáte čtyři podsítě, na třech půjčených bitech lze vytvořit osm podsítí atd.

Rozdělení sítě na jednotlivé podsítě má několik výhod. Jednou z nich je, že adresní prostor se může přizpůsobit počtu počítačů v dané podsíti, a tak se využije lépe, s minimálním plýtváním. Další výhodou je, že pokud se při vytváření podsítí vychází z jedné celkové sítě, která je rozdělena na podsítě, pak vnější síť nemusí znát podrobnosti adresace v podsítích, stačí, když zná adresu hlavní sítě. Paket pak pošle směrovači, na kterém jsou podsítě připojeny, a ten zajistí směrování do podsítí.

Ukázka rozdělení sítě na podsítě

Je potřeba rozdělit přidělený adresní rozsah **192.168.5.0/24** na dvě podsítě. Ke směrovači jsou k jeho dvěma rozhraním připojeny dvě sítě, mezi nimiž je nutné zabezpečit směrování.

Masku je třeba prodloužit o jeden bit – **/25**.

Adresa **192.168.5.0** vyjádřená binárně vypadá takto:

```
11000000. 10101000. 00000101. 00000000
```

S maskou **/24** končí síťová část za třetím bytem:

```
11000000. 10101000. 00000101. 00000000
```

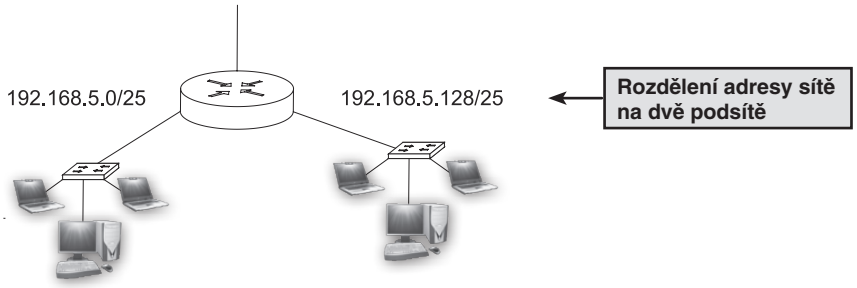
Po prodloužení masky bude síťová část končit až za prvním bitem čtvrtého bytu:

```
11000000. 10101000. 00000101. 00000000
```

Na tomto půjčeném bitu je možné zapsat jedničku nebo nulu:

- 11000000. 10101000. 00000101. 0**0**000000
- 11000000. 10101000. 00000101. 1**1**000000

Jedna síť bude mít adresu **192.168.5.0/25** a druhá **192.168.5.128/25**.



Každá z těchto podsítí má adresní prostor pro 126 síťových zařízení. V IP adrese sítě zbývá sedm bitů pro identifikaci těchto zařízení. Na sedmi pozicích lze získat $2^7 = 128$ různých variant jedniček a nul, ale jak už víte, varianty se samými nulami a samými jedničkami, které jsou adresami sítě a broadcastu, se pro adresování síťových zařízení nepoužívají.

První síť může používat IP adresy **192.168.5.1–192.168.5.126** s maskou **/25**, adresa **192.168.5.0** je adresou sítě, a adresa **192.168.5.127** je adresou broadcastu v této síti.

11000000. 10101000. 00000101. 0**0**000000 – 11000000. 10101000. 00000101. 0**1**111111

Druhá síť může používat IP adresy **192.168.5.129–192.168.5.254** s maskou **/25**. Adresa **192.168.5.128** je adresou této podsítě a adresa **192.168.5.255** je adresou broadcastu v této síti.

11000000. 10101000. 00000101. 1**0**000000 – 11000000. 10101000. 00000101. 1**1**111111

Rozdělení sítě na více podsítí – rozbor adres

Na směrovač je připojeno šest sítí. Administrátor zodpovědný za konfiguraci směrovače dostane přidělen adresní rozsah **172.16.0.0/16**.

Jakou masku podsítě je potřeba použít, aby se do tohoto adresního rozsahu vešlo šest podsítí, mezi nimiž bude směrovač schopen provádět směrování?

Kolik síťových zařízení bude možné v každé podsíti adresovat?

Jaké budou adresy podsítí?

Jaké budou adresní rozsahy těchto podsítí?

Jaká bude adresa broadcastu v jednotlivých podsítích?

Řešení

Víte, že s každým půjčeným bitem z části pro identifikaci síťových zařízení vzroste počet podsítí na dvojnásobek.

Na dvou půjčených bitech by bylo možné provozovat čtyři podsítě, na třech osm, pro šest podsítí je proto třeba půjčit tři bity.

Maska podsítě bude o tři bity delší, než byla původní maska, tj. **/19**.

Číselně vyjádřeno bude maska vypadat následovně:

11111111.11111111.11100000.00000000 – 255.255.224.0

Každé podsíti zbude 32 – 19 = 13 bitů pro adresování svých síťových zařízení.

Na těchto 13 bitech bude možno v každé podsíti získat $2^{13} - 2 =$ **8 190** různých použitelných adres pro jednotlivá síťová zařízení.

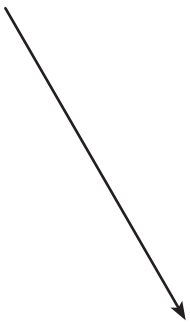
Pro určení jednotlivých IP adres je potřeba pracovat s adresou sítí v binární podobě.

172.16.0.0 /16: 10101100. 00010000. 00000000. 00000000

172.16.0.0 /19: 10101100. 00010000. 000**00000**. 00000000

Na půjčených třech bitech lze získat celkem osm různých variant jedniček a nul:

- 000
- 001
- 010
- 011
- 100
- 101
- 110
- 111



1. adresa podsítě:

   10101100. 00010000.**000**00000. 00000000 – **172.16.0.0 /19**

2. adresa podsítě:

   10101100. 00010000.**001**00000. 00000000 – **172.16.32.0 /19**

3. adresa podsítě:

   10101100. 00010000.**010**00000. 00000000 – **172.16.64.0 /19**

4. adresa podsítě:

10101100. 00010000.011**00000**. 00000000 – **172.16.96.0 /19**

5. adresa podsítě:

10101100. 00010000.100**00000**. 00000000 – **172.16.128.0 /19**

6. adresa podsítě:

10101100. 00010000.101**00000**. 00000000 – **172.16.160.0 /19**

Je vidět, že díky spojitosti číslování se třetí číslo v IP adrese vždy zvyšuje o 32. To lze využít při zobecnění úvah.

Adresní rozsahy podsítí:

Na zbývajících třinácti bitech se mohou pro adresování síťových zařízení jakkoliv kombinovat jedničky a nuly, vynechá se pouze kombinace samých nul (adresa sítě) a samých jedniček (adresa broadcastu).

1. adresa podsítě:

10101100. 00010000.000**00000. 00000000**

1. adresa pro síťové zařízení: 10101100. 00010000.000**00000. 00000001** – **172.16.0.1**
2. adresa pro síťové zařízení: 10101100. 00010000.000**00000. 00000010** – **172.16.0.2**
3. adresa pro síťové zařízení: 10101100. 00010000.000**00000. 00000011** – **172.16.0.3**

Je vidět, a lze to i zobecnit, že první adrese v podsíti odpovídá číslo 1, druhé odpovídá číslo 2 atd. Protože na 13 bitech lze vytvořit 8190 různých adres zařízení, musí se číslo vyšší než 255 rozprostřít přes celých třináct bitů a zasáhnout nejen poslední, ale i předposlední byte.

255. adresa pro síťové zařízení: 10101100. 00010000.000**00000. 11111111** – **172.16.0.255**
256. adresa pro síťové zařízení: 10101100. 00010000.000**00001. 00000000** – **172.16.1.0**
257. adresa pro síťové zařízení: 10101100. 00010000.000**00001. 00000001** – **172.16.1.1**
258. adresa pro síťové zařízení: 10101100. 00010000.000**00001. 00000010** – **172.16.1.2**

číslo 258 vyjádřené binárně a rozložené na posledních třináct bitů

8189. adresa pro síťové zařízení: 10101100. 00010000.000**11111. 11111101** – **172.16.31.253**
8190. adresa pro síťové zařízení: 10101100. 00010000.000**11111. 11111110** – **172.16.31.254**

Adresy v 1. podsíti jsou **172.16.0.1–172.16.0.255**, **172.16.1.0–172.16.1.255**, …, **172.16.31.0 až 172.16.31.254**.

Adresa broadcastu v první podsíti vznikne tak, že všech třináct bitů se vyplní samými jedničkami.

10101100. 00010000.00011111. 11111111 – **172.16.31.255**.

2. adresa podsítě:

10101100. 00010000.00100000. 00000000 – **172.16.32.0 /19**

Jak je vidět, adresa druhé podsítě navazuje plynule na broadcast v první podsíti. A takto to pokračuje i dále, proto není nutné rozepisovat binárně všech šest podsítí, aby mohly být vyjádřeny adresní rozsahy a broadcasty. Stačí se podívat na následující síť – předchozí adresa je adresa broadcastu v předchozí podsíti atd.

Adresní rozsah ve 2. podsíti bude vypadat následovně:

172.16.32.1–172.16.32.255, **172.16.33.0–172.16.33.255**, …, **172.16.63.0–172.16.63.254**

Adresa broadcastu v 2. podsíti bude **172.16.63.255**.

3. adresa podsítě:

10101100. 00010000.01000000. 00000000 – **172.16.64.0 /19**

Adresní rozsah ve 3. podsíti bude vypadat následovně:

172.16.64.1–172.16.64.255, **172.16.65.0–172.16.65.255**, …, **172.16.95.0–172.16.95.254**

Adresa broadcastu v 3. podsíti bude **172.16.95.255**.

4. adresa podsítě:

10101100. 00010000.01100000. 00000000 – **172.16.96.0 /19**

Adresní rozsah ve 4. podsíti bude vypadat následovně:

172.16.96.1–172.16.96.255, **172.16.97.0–172.16.97.255**, …, **172.16.127.0–72.16.127.254**

Adresa broadcastu v 4. podsíti bude **172.16.127.255**.

5. adresa podsítě:

10101100. 00010000.10000000. 00000000 – **172.16.128.0 /19**

Adresní rozsah v 5. podsíti bude vypadat následovně:

172.16.128.1–172.16.128.255, **172.16.129.0–172.16.129.255**, …, **172.16.159.0–172.16.159.254**

Adresa broadcastu v 5. podsíti bude **172.16.159.255**.

6. adresa podsítě:

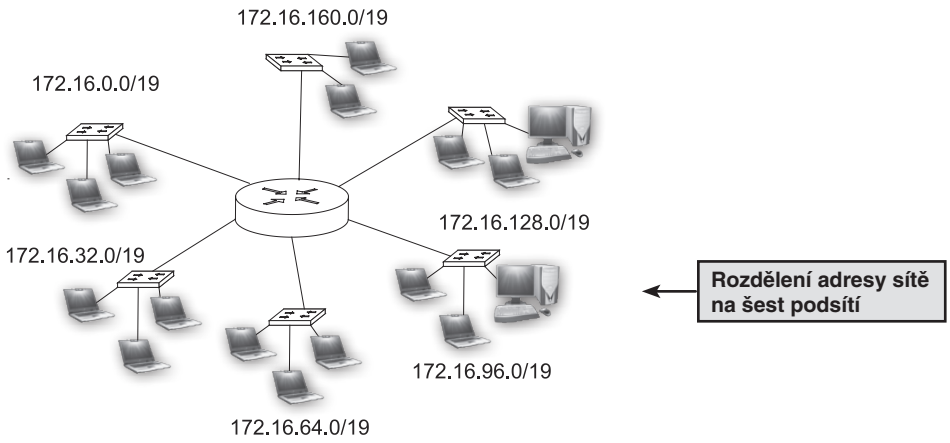
10101100. 00010000.10100000. 00000000 – **172.16.160.0 /19**

Adresní rozsah v 6. podsíti bude vypadat následovně:

172.16.160.1–172.16.160.255, **172.16.161.0–172.16.161.255**, …, **172.16.191.0–172.16.191.254**

Adresa broadcastu v 6. podsíti bude **172.16.191.255**.

Pro lepší představu, kam je až možné zajít v adresaci šesté podsítě, lze nastínit, jak by pokračovala adresace v sedmé podsíti. Číslo na třetím bytu by bylo o 32 vyšší než 160, tj. adresace v sedmé podsíti by začínala adresou **172.16.192.0**. Proto broadcast šesté podsítě je hned předchozí číslo a adresa posledního síťového zařízení v šesté podsíti předchází broadcastu v této síti.



Přizpůsobení podsítí počtu zařízení v síti

Jiný pohled jak rozdělit síť na podsítě vychází z potřeby přizpůsobit velikost podsítě počtu síťových zařízení, která se budou na síti vyskytovat.

Jak už vyplynulo z předchozího, na x bitech v IP adrese, které zůstávají pro jednoznačnou identifikaci síťových zařízení, lze získat $2^x - 2$ použitelných IP adres.

Z této úvahy se odvíjí stanovení počtu bitů, které musí zůstat pro identifikaci určitého počtu síťových zařízení.

Jestliže síť obsahuje přibližně 20 síťových zařízení, bude stačit ponechat pro tato síťová zařízení 5 bitů. Na nich lze získat $2^5 - 2 = 30$ různých IP adres.

Pro síť, která bude obsahovat 500 síťových zařízení, bude potřeba ponechat pro identifikaci těchto zařízení 9 bitů. Na nich lze získat $2^9 - 2 = 510$ různých IP adres.

Spojovací linka mezi dvěma zařízeními, například směrovači, bude potřebovat jen dvě adresy, proto pro jejich identifikaci stačí ponechat 2 bity, na nich lze získat právě dvě různé IP adresy: $2^2 - 2 = 2$.

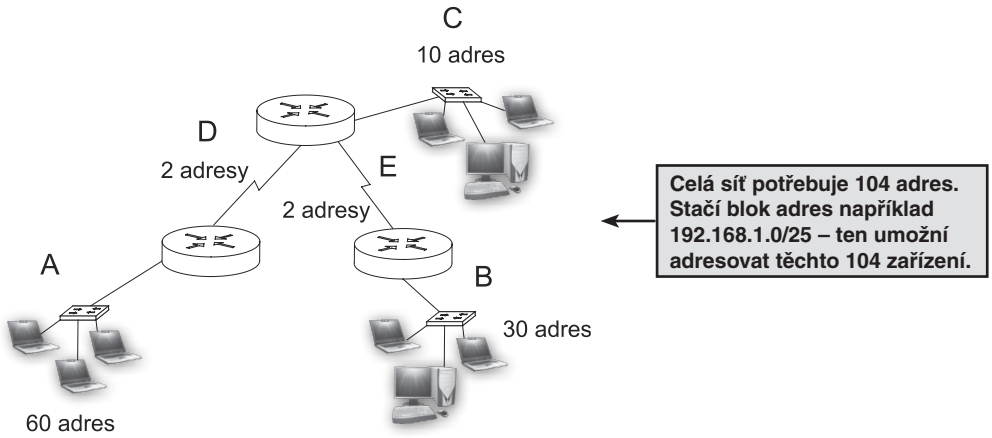
Při vytváření podsítí s různým počtem síťových zařízení se postupuje od největší sítě k nejmenší. Nejprve se přiřadí rozsah největší síti a pak se zbývající adresy dělí mezi další, menší sítě.

Nikdy se nesmí stát, že by se adresní rozsahy překrývaly, vedlo by to k chybám při směrování.

Postup při vytváření podsítí

Nejprve se musí určit, kolik podsítí bude třeba vytvořit a jaký budou mít adresní prostor.

Vyčíslí se jednotlivé adresní rozsahy podsítí. Vychází se od podsítí s největším adresním prostorem a postupuje se k sítím pro nejmenší počet zařízení (například spojovací linky spojující směrovače).

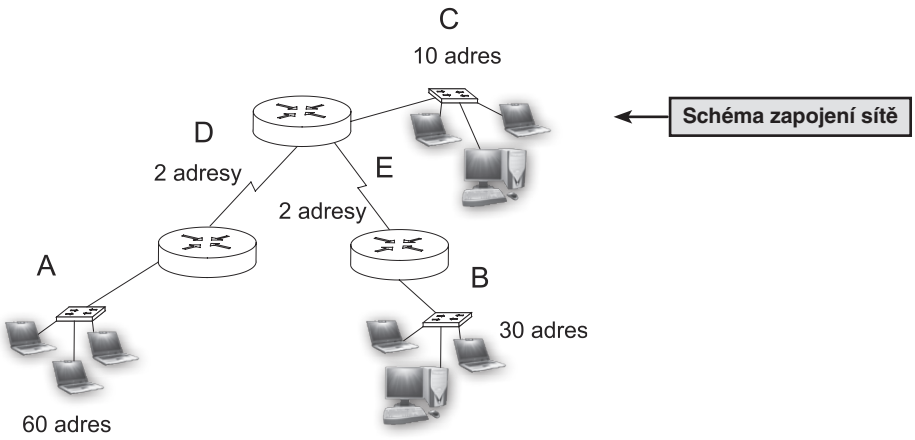


Vytvoření podsítí s různými rozsahy

Příklad

Ve struktuře firemní sítě se nacházejí tři lokální segmenty obsahující počítače a dva segmenty spojující směrovače. Segmenty jsou zapojeny následujícím způsobem.

V segmentu **A** je potřeba adresní prostor pro 60 zařízení, v segmentu **B** pro 30 zařízení, v segmentu **C** pro 10 zařízení, segmenty **D** a **E** jsou pouze spojovací linky mezi směrovači a každá potřebuje dvě IP adresy.



Celkem síť potřebuje adresovat 104 zařízení.

Řešení

Nejbližší vyšší číslo k číslu 104, které je mocninou dvojky, je sedmá mocnina dvojky, $2^7 = 128$.

Ponechte tedy 7 bitů pro adresování koncových zařízení.

Maska základní podsítě bude zapsána prefixem **/25**. Na takové síti s touto maskou byste mohli adresovat 126 různých zařízení, ale všechna by byla ve stejné síti, tudíž by nebylo možné směrování.

Musíte vyjít z nějaké základní IP adresy sítě. Zvolte číslo základní sítě například **192.168.1.0/25**.

Se stanovením čísla podsítě začněte u sítě s největším počtem zařízení – zde je to segment **A**. Ten potřebuje 60 IP adres. Nejbližší vyšší číslo k číslu 60, které je mocninou dvojky, je číslo $64 = 2^6$.

Maska odpovídající 60 IP adresám je zapsána prefixem **/26**. Na posledních 6 bitech (32 – 26 = 6) lze vytvořit 62 (64 – 2) různých IP adres.

Zapište si základní IP adresu **192.168.1.0/25** binárně a vyznačte půjčený bit.

```
11000000. 10101000. 00000001. 00000000
```

Na tomto bitu lze zapsat nulu nebo jedničku. Vzniknou takto dvě různé podsítě s prefixem masky **/26**, kde každá z nich má adresní prostor pro 62 IP adres.

1. podsíť: 11000000. 10101000. 00000001. 00000000 – (**192.168.1.0/26**)
2. podsíť: 11000000. 10101000. 00000001. 01000000 – (**192.168.1.64/26**)

První z těchto adres sítě použijte pro **segment A: 192.168.1.0/26**.

Druhou síť budete dále dělit.

Segment **B** potřebuje 30 IP adres. Takový adresní prostor nabízí maska zapsaná prefixem **/27**. Na zbývajících 5 bitech (32 – 27 = 5) lze získat 30 různých IP adres.

Vyjděte z druhé základní podsítě (11000000. 10101000. 00000001. 01000000) a její masku prodlužte o další bit. Tak vzniknou dvě podsítě, každá z nich bude mít adresní prostor pro 30 IP adres.

1. podsíť: 11000000. 10101000. 00000001. 0**1**0000000 – (**192.168.1.64/27**)
2. podsíť: 11000000. 10101000. 00000001. 0**1**1000000 – (**192.168.1.96/27**)

První z těchto adres použijte pro **segment B: 192.168.1.64/27**.

Druhou síť budete dále dělit.

Segment **C** potřebuje 10 IP adres. Takový adresní prostor nabízí maska zapsaná prefixem **/28**. Na zbývajících 4 bitech (32 – 28 = 4) lze získat 14 různých IP adres.

Vyjděte z druhé, zatím nevyužité podsítě (11000000. 10101000. 00000001. 01100000) a její masku prodlužte o další bit. Masku pak můžete zapsat prefixem **/28**. Půjčeným bitem vzniknou další dvě podsítě, z nichž každá má adresní prostor pro 14 IP adres.

1. podsíť: 11000000. 10101000. 00000001. 0**11**00000 – (**192.168.1.96/28**)
2. podsíť: 11000000. 10101000. 00000001. 0**11**10000 – (**192.168.1.112/28**)

První z těchto adres použijte pro **segment C: 192.168.1.96/28**.

Nakonec potřebujete vytvořit podsítě pro spojovací linky – segmenty **D** a **E**.

Ještě stále máte k dispozici jednu nevyužitou podsíť – poslední v pořadí:
11000000.10101000.00000001.0**11**10000 – (**192.168.1.112/28**).

Abyste mohli vytvořit dvě platné IP adresy pro adresování rozhraní směrovačů na spojovací lince, je třeba ponechat pro toto adresování dva bity. Proto bude maska podsítě pro segmenty **D** a **E** mít prefix **/30** (32 – 30 = 2).

V nevyužité podsíti 11000000. 10101000. 00000001. 0**11**10000 prodlužte masku tak, aby ještě zbyly dva bity pro koncové IP adresy, tedy o dva bity.

Na dalších dvou půjčených bitech můžete variacemi jedniček a nul získat čtyři podsítě, z nichž každá bude mít adresní prostor pro dvě IP adresy.

1. podsíť: 11000000. 10101000. 00000001. 0**11100**00 – (**192.168.1.112/30**)
2. podsíť: 11000000. 10101000. 00000001. 0**11101**00 – (**192.168.1.116/30**)
3. podsíť: 11000000. 10101000. 00000001. 0**11110**00 – (**192.168.1.120/30**)
4. podsíť: 11000000. 10101000. 00000001. 0**11111**00 – (**192.168.1.124/30**)

První z těchto podsítí použijte pro **segment D: 192.168.1.112/30**.

Druhou z těchto malých podsítí použijte pro **segment E: 192.168.1.116/30**.

Jak je vidět, zůstanou k dispozici dvě nevyužité malé podsítě: **192.168.1.120/30** a **192.168.1.124/30**.

<table>
<tr><td colspan="6">192.168.1.0/25 – základní síť</td></tr>
<tr><td rowspan="4">192.168.1.0/26
Segment A</td><td colspan="5">192.168.1.64/26</td></tr>
<tr><td rowspan="3">192.168.1.64/27
Segment B</td><td colspan="4">192.168.1.96/27</td></tr>
<tr><td rowspan="2">192.168.1.96/28
Segment C</td><td colspan="4">192.168.1.112/28</td></tr>
<tr><td>192.168.1.112/30
Segment D</td><td>192.168.1.116/30
Segment E</td><td>192.168.1.120/30
nevyužito</td><td>192.168.1.124/30
nevyužito</td></tr>
</table>

Je evidentní, že podsíťováním se přichází o některé IP adresy – jsou to IP adresy podsítí a broadcastů v těchto podsítích. Tyto adresy již nelze použít pro adresování síťového zařízení. S každou podsítí se ztrácejí dvě adresy. Proto se při vytváření podsítí může stát, že nakonec nezbude dostatečný adresní prostor pro poslední síť. Pak byste museli začít od začátku, a to s maskou o jeden bit kratší než původně.

Určení adres v sítích

Určete, jaké IP adresy lze používat v sítích **A, B, C, D, E** z předchozího příkladu. Určete adresu broadcastu v těchto sítích.

Síť A: 192.168.1.0/26

Prefix **/26** znamená, že maska je prodloužena na 26 bitů.

Rozepište IP adresu sítě binárně:

`11000000. 10101000. 00000001. 00000000` – **192.168.1.0/26**

Na posledních šesti bitech lze variacemi jedniček a nul získat celkem 62 různých použitelných IP adres. Samé nuly by vytvořily adresu shodnou s adresou sítě – nepoužívá se, samé jedničky jsou adresou broadcastu v této síti.

1. IP adresa `11000000. 10101000. 00000001. 00000001`
2. IP adresa `11000000. 10101000. 00000001. 00000010`
3. IP adresa `11000000. 10101000. 00000001. 00000011`

62. IP adresa `11000000. 10101000. 00000001. 00111110`

IP adresy použitelné v této síti pro adresování síťových zařízení jsou **192.168.1.1** až **192.168.1.62**.

Adresa broadcastu v této síti: 11000000. 10101000. 00000001. 00111111 – **192.168.1.63**.

Síť B: 192.168.1.64/27

Prefix **/27** znamená, že maska je prodloužena na 27 bitů.

Rozepište IP adresu sítě binárně:

11000000. 10101000. 00000001. 01000000 – **192.168.1.64/27**

Na posledních pěti bitech lze variacemi jedniček a nul získat celkem 30 různých použitelných IP adres. Samé nuly by vytvořily adresu shodnou s adresou sítě – nepoužívá se, samé jedničky jsou adresou broadcastu v této síti.

1. IP adresa 11000000. 10101000. 00000001. 01000001
2. IP adresa 11000000. 10101000. 00000001. 01000010
3. IP adresa 11000000. 10101000. 00000001. 01000011

30. IP adresa 11000000. 10101000. 00000001. 01011110

IP adresy použitelné v této síti pro adresování síťových zařízení jsou **192.168.1.65** až **192.168.1.94**.

Adresa broadcastu v této síti: 11000000. 10101000. 00000001. 01011111 – **192.168.1.95**.

Síť C: 192.168.1.96/28

Prefix **/28** znamená, že maska je prodloužena na 28 bitů.

Rozepište IP adresu sítě binárně:

11000000. 10101000. 00000001. 01100000 – **192.168.1.96/28**

Na posledních čtyřech bitech lze variacemi jedniček a nul získat celkem 14 různých použitelných IP adres. Samé nuly by vytvořily adresu shodnou s adresou sítě – nepoužívá se, samé jedničky jsou adresou broadcastu v této síti.

1. IP adresa 11000000. 10101000. 00000001. 01100001
2. IP adresa 11000000. 10101000. 00000001. 01100010
3. IP adresa 11000000. 10101000. 00000001. 01100011

14. IP adresa 11000000. 10101000. 00000001. 01101110

IP adresy použitelné v této síti pro adresování síťových zařízení jsou **192.168.1.97** až **192.168.1.110**.

Adresa broadcastu v této síti: 11000000. 10101000. 00000001. 0110**1111** – **192.168.1.111**.

Síť D: 192.168.1.112/30

Prefix **/30** znamená, že maska je prodloužena na 30 bitů.

Rozepište IP adresu sítě binárně:

11000000. 10101000. 00000001. 01110000 – **192.168.1.112/30**

Na posledních dvou bitech lze variacemi jedniček a nul získat celkem 2 různé použitelné IP adresy. Samé nuly by vytvořily adresu shodnou s adresou sítě – nepoužívá se, samé jedničky jsou adresou broadcastu v této síti.

1. IP adresa 11000000. 10101000. 00000001. 01110001
2. IP adresa 11000000. 10101000. 00000001. 01110010

IP adresy použitelné v této síti pro adresování síťových zařízení jsou **192.168.1.113** a **192.168.1.114**.

Adresa broadcastu v této síti: 11000000. 10101000. 00000001. 01110011 – **192.168.1.115**.

Síť E: 192.168.1.116/30

Prefix **/30** znamená, že maska je prodloužena na 30 bitů.

Rozepište IP adresu sítě binárně:

11000000. 10101000. 00000001. 01110100 – **192.168.1.116/30**

Na posledních dvou bitech lze variacemi jedniček a nul získat celkem 2 různé použitelné IP adresy. Samé nuly by vytvořily adresu shodnou s adresou sítě – nepoužívá se, samé jedničky jsou adresou broadcastu v této síti.

1. IP adresa 11000000. 10101000. 00000001. 01110101
2. IP adresa 11000000. 10101000. 00000001. 01110110

IP adresy použitelné v této síti pro adresování síťových zařízení jsou **192.168.1.117** a **192.168.1.118**.

Adresa broadcastu v této síti: 11000000. 10101000. 00000001. 01110111 – **192.168.1.119**.

Shrnutí výsledků

Síť	IP adresa sítě	Maska podsítě	Broadcast	Adresy v síti
A	192.168.1.0	255.255.255.192	192.168.1.63	192.168.1.1– 92.168.1.62
B	192.168.1.64	255.255.255.224	192.168.1.95	192.168.1.65–192.168.1.94
C	192.168.1.96	255.255.255.240	192.168.1.111	192.168.1.97–192.168.1.110
D	192.168.1.112	255.255.255.252	192.168.1.115	192.168.1.113–192.168.1.114
E	192.168.1.116	255.255.255.252	192.168.1.119	192.168.1.117–192.168.1.118

14. Testování síťové vrstvy

Ping

Ping – zkratka z angl. ***Packet InterNet Groper***

Zápis: **ping** IP adresa nebo **ping** doménové jméno

Pokud je parametrem příkazu **ping** doménové jméno, je nejprve kontaktován DNS server, který zajistí překlad tohoto jména do číselné podoby.

Pomocí příkazu **ping** lze testovat spojení mezi dvěma síťovými zařízeními, na nichž je nakonfigurován protokol TCP/IP.

Ping používá protokol ICMP, který je součástí třetí vrstvy OSI modelu, síťové vrstvy.

ICMP – ***Internet Control Message Protocol***

Zdrojový počítač vysílá určitý druh zprávy **protokolu ICMP** – tzv. ***Echo Request*** (požadavek odezvy). Jakmile cílový počítač tento požadavek přijme, odesílá jiný druh zprávy protokolu ICMP, tzv. ***Echo Reply*** (odeslání odezvy).

Mezi odesláním požadavku na odezvu a navrácením odezvy se měří čas, který se následně zobrazí na výstupu příkazu **ping**.

Pomocí tohoto příkazu lze změřit výkonnost sítě. Pokud zdrojový počítač čeká na odezvu příliš dlouho, déle, než je jeho přednastavená hodnota pro čekání, vyhodnotí to, jako že žádnou odpověď nedostal.

Po zadání příkazu **ping** je odesláno několik požadavků odezvy, po návratu odpovědi se zobrazí celková statistika odeslaných paketů, minimální, maximální a průměrný čas odezvy.

```
C:\Users\Iva>ping 81.95.96.94

Příkaz PING na 81.95.96.94 - 32 bajtů dat:
Odpověď od 81.95.96.94: bajty=32 čas=6ms TTL=58
Odpověď od 81.95.96.94: bajty=32 čas=4ms TTL=58
Odpověď od 81.95.96.94: bajty=32 čas=4ms TTL=58
Odpověď od 81.95.96.94: bajty=32 čas=4ms TTL=58

Statistika ping pro 81.95.96.94:
Pakety: Odeslané = 4, Přijaté = 4, Ztracené = 0 (ztráta 0%),
Přibližná doba do přijetí odezvy v milisekundách:
    Minimum = 4ms, Maximum = 6ms, Průměr = 4ms
```

Ukázka výstupu příkazu *ping*

Ping na vnitřní rozhraní

Pro zjištění, zda je na počítači správně nakonfigurován protokol TCP/IP, slouží **ping** na vnitřní rozhraní, tzv. loopback, jehož adresa je **127.0.0.1**.

Ping na vnitřní rozhraní testuje funkčnost tří dolních vrstev OSI modelu.

```
C:\Users\Iva>ping 127.0.0.1

Příkaz PING na 127.0.0.1 - 32 bajtů dat:
Odpověď od 127.0.0.1: bajty=32 čas < 1ms TTL=128
Odpověď od 127.0.0.1: bajty=32 čas < 1ms TTL=128
Odpověď od 127.0.0.1: bajty=32 čas < 1ms TTL=128
Odpověď od 127.0.0.1: bajty=32 čas < 1ms TTL=128

Statistika ping pro 127.0.0.1:
Pakety: Odeslané = 4, Přijaté = 4, Ztracené = 0 (ztráta 0%),
Přibližná doba do přijetí odezvy v milisekundách:
    Minimum = 0ms, Maximum = 0ms, Průměr = 0ms
```

Ping na vnitřní rozhraní

Ping na výchozí bránu

Někdy je potřeba otestovat spojení mimo lokální síť. Pokud nefunguje, je možné ověřit, zda je vůbec dostupná výchozí brána, která spojení s vnější sítí zajišťuje.

Jestliže je **ping** na výchozí bránu úspěšný, znamená to, že toto spojení správně funguje.

Pokud **ping** na výchozí bránu nefunguje, může to znamenat problém buď s výchozí bránou, nebo se zdrojovým počítačem, případně se spojením mezi nimi. To lze zjistit provedením příkazu **ping** na jiný počítač, který se nachází ve stejné lokální síti jako výchozí počítač.

Jestliže **ping** na jiný počítač v lokální síti projde bez problémů, je pravděpodobně problém na straně směrovače, který funguje jako výchozí brána. Pak je vhodné ověřit, zda je směrovač k lokální síti správně připojen, zda běží, případně jej restartovat, a nepomůže-li to, následně prověřit jeho konfiguraci.

```
C:\Users\Iva>ipconfig

Konfigurace protokolu IP systému Windows


Adaptér bezdrátové sítě LAN Bezdrátové připojení k síti:

   Stav média  . . . . . . . . . . . : odpojeno
   Přípona DNS podle připojení . . . :

Adaptér sítě Ethernet Připojení k místní síti:

   Přípona DNS podle připojení . . . :
   Adresa IPv4 . . . . . . . . . . . : 192.168.0.101
   Maska podsítě . . . . . . . . . . : 255.255.255.0
   Výchozí brána . . . . . . . . . . : 192.168.0.1
```

Ověření konfigurace počítače pomocí příkazu *ipconfig*

Někdy může být směrovač nastaven tak, že na výzvu k odezvě příkazu **ping** vůbec neodpovídá, což je jeho ochrana proti určitému typu útoků v síti. Přesto může být směrovač funkční.

Aby mohl počítač komunikovat se směrovačem, musí být oba na stejné lokální síti (z IP adresy to již umíte ověřit), musí být správně propojeny a směrovač nesmí mít nastavena bezpečnostní opatření, která by bránila počítači v přístupu.

```
C:\Users\Iva>ping 192.168.0.1

Příkaz PING na 192.168.0.1 - 32 bajtů dat:
Odpověď od 192.168.0.1: bajty=32 čas=2ms TTL=64
Odpověď od 192.168.0.1: bajty=32 čas=1ms TTL=64
Odpověď od 192.168.0.1: bajty=32 čas=1ms TTL=64
Odpověď od 192.168.0.1: bajty=32 čas=1ms TTL=64

Statistika ping pro 192.168.0.1:
Pakety: Odeslané = 4, Přijaté = 4, Ztracené = 0 (ztráta 0%),
Přibližná doba do přijetí odezvy v milisekundách:
    Minimum = 1ms, Maximum = 2ms, Průměr = 1ms
```

Ověření spojení s výchozí bránou

Pokud se podaří získat odezvu z výchozí brány na příkaz **ping** a dále na nějaké zařízení mimo lokální síť, pak je zřejmé, že směrovač funguje jako výchozí brána dobře, zajišťuje směrování a zdrojový počítač je správně nastaven.

```
C:\Users\Iva>ping www.ssps.cz

Příkaz PING na www.ssps.cz [81.95.96.94] - 32 bajtů dat:
Odpověď od 81.95.96.94: bajty=32 čas=3ms TTL=58
Odpověď od 81.95.96.94: bajty=32 čas=5ms TTL=58
Odpověď od 81.95.96.94: bajty=32 čas=4ms TTL=58
Odpověď od 81.95.96.94: bajty=32 čas=4ms TTL=58

Statistika ping pro 81.95.96.94:
Pakety: Odeslané = 4, Přijaté = 4, Ztracené = 0 (ztráta 0%),
Přibližná doba do přijetí odezvy v milisekundách:
    Minimum = 3ms, Maximum = 5ms, Průměr = 4ms
```

Ověření spojení s počítačem na vnější síti

Fakt, že **ping** na určité síťové zařízení nefunguje, nemusí znamenat problémy v síti. Z bezpečnostních důvodů bývá odezva na příkaz **ping** někdy zakázána.

Některé parametry příkazu ping

Parametr	Popis	Ukázka zápisu
-t	Příkaz **ping** s tímto parametrem testuje cílové zařízení neustále, dokud není přerušen uživatelem. Pro přerušení testování a zobrazení statistiky stiskněte kombinaci kláves **Ctrl+Break.** Pro ukončení testování stiskněte kombinaci kláves **Ctrl+C.**	**ping www.ssps.cz –t**
-n číslo	Po zadání příkazu **ping** následovaného tímto parametrem bude odeslán zadaný počet požadavků na odezvu. Například **–n 10** odešle 10 požadavků na odezvu.	**ping www.ssps.cz –n 10**

-i TTL	Jestliže za příkaz **ping** napíšete tento parametr, omezíte počet průchodů uzly. Je-li zadané číslo TTL příliš malé, může být paket na některém směrovači po cestě zahozen a nedorazit k cíli.	**ping www.ssps.cz –i 9**

```
C:\Users\Iva>ping www.ssps.cz -t

Příkaz PING na www.ssps.cz [81.95.96.94] - 32 bajtů dat:
Odpověď od 81.95.96.94: bajty=32 čas=4ms TTL=58
Odpověď od 81.95.96.94: bajty=32 čas=7ms TTL=58
Odpověď od 81.95.96.94: bajty=32 čas=7ms TTL=58
Odpověď od 81.95.96.94: bajty=32 čas=4ms TTL=58
Odpověď od 81.95.96.94: bajty=32 čas=4ms TTL=58
Odpověď od 81.95.96.94: bajty=32 čas=4ms TTL=58
Odpověď od 81.95.96.94: bajty=32 čas=8ms TTL=58
Odpověď od 81.95.96.94: bajty=32 čas=4ms TTL=58
Odpověď od 81.95.96.94: bajty=32 čas=4ms TTL=58

Statistika ping pro 81.95.96.94:
Pakety: Odeslané = 9, Přijaté = 9, Ztracené = 0 (ztráta 0%),
Přibližná doba do přijetí odezvy v milisekundách:
    Minimum = 4ms, Maximum = 8ms, Průměr = 5ms
Control-C
^C
```

Nekonečné testování cílového počítače, je použit parametr *-t*

```
C:\Users\Iva>ping www.ssps.cz -n 10

Příkaz PING na www.ssps.cz [81.95.96.94] - 32 bajtů dat:
Odpověď od 81.95.96.94: bajty=32 čas=3ms TTL=58
Odpověď od 81.95.96.94: bajty=32 čas=4ms TTL=58
Odpověď od 81.95.96.94: bajty=32 čas=4ms TTL=58
Odpověď od 81.95.96.94: bajty=32 čas=5ms TTL=58
Odpověď od 81.95.96.94: bajty=32 čas=4ms TTL=58
Odpověď od 81.95.96.94: bajty=32 čas=4ms TTL=58
Odpověď od 81.95.96.94: bajty=32 čas=5ms TTL=58
Odpověď od 81.95.96.94: bajty=32 čas=8ms TTL=58
Odpověď od 81.95.96.94: bajty=32 čas=5ms TTL=58
Odpověď od 81.95.96.94: bajty=32 čas=3ms TTL=58

Statistika ping pro 81.95.96.94:
Pakety: Odeslané = 10, Přijaté = 10, Ztracené = 0 (ztráta 0%),
Přibližná doba do přijetí odezvy v milisekundách:
    Minimum = 3ms, Maximum = 8ms, Průměr = 4ms
```

Pomocí parametru *–n 10* dojde k odeslání deseti požadavků na odezvu.

```
C:\Users\Iva>ping www.ssps.cz -i 7

Příkaz PING na www.ssps.cz [81.95.96.94] - 32 bajtů dat:
Odpověď od 194.50.100.236: Hodnota TTL vypršela při přechodu.
Odpověď od 194.50.100.236: Hodnota TTL vypršela při přechodu.
Odpověď od 194.50.100.236: Hodnota TTL vypršela při přechodu.
Odpověď od 194.50.100.236: Hodnota TTL vypršela při přechodu.

Statistika ping pro 81.95.96.94:
Pakety: Odeslané = 4, Přijaté = 4, Ztracené = 0 (ztráta 0%),

C:\Users\Iva>ping www.ssps.cz -i 8

Příkaz PING na www.ssps.cz [81.95.96.94] - 32 bajtů dat:
Odpověď od 81.95.96.94: bajty=32 čas=5ms TTL=58
Odpověď od 81.95.96.94: bajty=32 čas=6ms TTL=58
Odpověď od 81.95.96.94: bajty=32 čas=3ms TTL=58
Odpověď od 81.95.96.94: bajty=32 čas=3ms TTL=58

Statistika ping pro 81.95.96.94:
Pakety: Odeslané = 4, Přijaté = 4, Ztracené = 0 (ztráta 0%),
Přibližná doba do přijetí odezvy v milisekundách:
    Minimum = 3ms, Maximum = 6ms, Průměr = 4ms
```

Za parametrem *-i* je zadáno příliš malé číslo, aby mohl paket doputovat až k cíli. Je zahozen některým směrovačem po cestě.

Zde je za parametrem *-i* zadáno dostatečně velké číslo, aby mohl paket doputovat až k cíli.

Tracert

Pomocí příkazu **tracert** (trace route) lze zjistit cestu od zdrojového počítače k cílovému. Ve výpisu příkazu **tracert** se objeví všechny uzly (směrovače), kterými pakety příkazu **tracert** prošly.

Po zadání příkazu **tracert** dojde k odeslání **paketu ICMP Echo Request** směrem k cílovému počítači.

```
C:\Users\Iva>tracert www.ssps.cz

Výpis trasy k www.ssps.cz [81.95.96.94]
s nejvýše 30 směrováními:

  1     1 ms    < 1 ms    < 1 ms  info.mladotova.cz [212.27.205.129]
  2     1 ms     1 ms     1 ms  v10.bytovedruzstvoinstart.cpe.vol.cz [212.27.198
.57]
  3     3 ms     3 ms     7 ms  v50.b1.opa.prg.vol.cz [195.250.133.1]
  4     3 ms     3 ms     3 ms  v302.c1.opa.prg.vol.cz [195.250.141.153]
  5     4 ms     4 ms     4 ms  v124.bb3.prg2.vol.cz [212.20.124.42]
  6     3 ms     4 ms     4 ms  ge3-1.tr1.prg2.vol.cz [212.20.124.62]
  7     5 ms     6 ms     3 ms  nix4-ge.active24.cz [194.50.100.236]
  8     4 ms     3 ms     8 ms  uvirt7.active24.cz [81.95.96.94]
```

Ve výpisu jsou vidět uzly, kterými paket směrem k cíli prošel, a doby odezev těchto uzlů.

Ve výpisu se objeví jednotlivé doby odezev jednotlivých uzlů. Pokud je některá doba odpovědi od průchozích uzlů příliš dlouhá, může to znamenat přetížení daného uzlu a problémy v pokračování.

Pokud dojde ke ztrátě paketu, zobrazí se to ve výpisu pomocí hvězdičky *****.

RTT (***Round Trip Time***) je doba, za kterou paket dorazí k následujícímu uzlu a zpět.

```
C:\Users\Iva>tracert www.gts.cz

Výpis trasy k apollo.fg.cz [81.95.101.194]
s nejvýše 30 směrováními:

  1     1 ms     1 ms     1 ms  info.mladotova.cz [212.27.205.129]
  2     2 ms     1 ms     1 ms  v10.bytovedruzstvoinstart.cpe.vol.cz [212.27.198
.57]
  3     3 ms     3 ms     3 ms  v50.b1.opa.prg.vol.cz [195.250.133.1]
  4     9 ms    21 ms    12 ms  v302.c1.opa.prg.vol.cz [195.250.141.153]
  5     4 ms     4 ms     5 ms  v124.bb3.prg2.vol.cz [212.20.124.42]
  6     3 ms     3 ms     4 ms  ge3-2.tr1.prg2.vol.cz [212.20.124.64]
  7     4 ms     4 ms     4 ms  nix4-ge.active24.cz [194.50.100.236]
  8     *        *        *     Vypršel časový limit žádosti.
  9     *        *        *     Vypršel časový limit žádosti.
 10     *        *        *     Vypršel časový limit žádosti.
 11     *        *        *     Vypršel časový limit žádosti.
 12     *        *        *     Vypršel časový limit žádosti.
 13     *        *        *     Vypršel časový limit žádosti.
 14     *        *        *     Vypršel časový limit žádosti.
 15     *
```

Ztracené pakety jsou označené *.

Pokud příkaz **tracert** nedokázal dosáhnout cíle, je možné z výpisu určit, na kterém uzlu cesta paketu skončila.

V hlavičce paketu, který je poslán příkazem **tracert**, se vyskytuje políčko **TTL** (***Time to Live***), které udává, kolika uzly může paket směrem k cíli projít.

Při každém průchodu přes směrovač se toto políčko sníží o 1 a ve chvíli, kdy dosáhne nuly, je paket zahozen.

Směrovač, který paket zahodí kvůli přesáhnutí povolené hodnoty TTL, pošle zdrojovému počítači zprávu o zahození paketu z důvodu přesáhnutí TTL – zprávu **ICMP Time Exceeded**.

Na začátku je TTL nastaven na hodnotu **1**. Při průchodu paketu prvním směrovačem sníží směrovač hodnotu TTL na nulu a pošle zdrojovému počítači chybovou zprávu.

Tracert si pak zvýší hodnotu TTL o jedna a pošle dotaz znovu. Pokud toto nastavení TTL ještě nestačí a dojde k zahození, opět dojde k následnému zvýšení TTL atd., dokud paket nedosáhne cíle. Toto zvyšování probíhá automaticky až k přednastavenému maximu, které má zabránit, aby paket neběhal donekonečna po síti.

Jakmile paket dosáhne cíle, cílový počítač odešle odpověď o doručení (**ICMP Echo Reply**).

Toto postupné zvyšování TTL a odesílání chybových zpráv směrovači na síti zdrojovému počítači slouží k vystavění seznamu uzlů, kterými paket prochází při cestě k cíli.

Protože lze pomocí příkazu **tracert** zmapovat topologii sítě, což může být některými záškodníky zneužito, je odezva na příkaz **tracert** na síťových zařízeních často vypnuta administrátorem.

ICMP

ICMP – zkratka z angl. ***Internet Control Messaging Protocol***

Používá ho někdy protokol IP pro zasílání kontrolních a chybových zpráv z přenosu paketu. Často ale posílání vzkazů pomocí ICMP protokolu není z bezpečnostních důvodů povoleno.

Využívají jej například programy **ping** a **tracert**.

Vzkazy protokolu ICMP mohou obsahovat:

- **Potvrzení funkčnosti a dosažitelnosti síťového zařízení** – tento typ zprávy využívá především program **ping**. Zdrojový počítač vyšle **ICMP Echo Request**, a pokud zpráva dorazí úspěšně do cíle a ten může odpovědět, odešle odpověď – zprávu **ICMP Echo Reply**.
- **Nedostupnost cíle nebo služby** – jestliže směrovač nebo počítač obdrží zprávu, kterou nemůže doručit, odešle zdrojovému zařízení chybovou zprávu o nedoručitelnosti – **Destination Unreachable**.

  Zpráva obsahuje podrobnosti o důvodu nedoručení – například nedostupná síť, nedostupné koncové zařízení, nedostupný protokol, nedostupný port.

Nedostupnost sítě hlásí směrovač tehdy, jestliže nemůže směrovat paket do cílové sítě, nedostupnost koncového zařízení v případě, že je síť cílového počítače dostupná, ale koncové zařízení pro směrovač v danou chvíli neexistuje.

Hlášení o nedostupnosti portu nebo protokolu zasílá cílový počítač v případě, že se mu nedaří předat paket vyšším vrstvám ke zpracování.

- **Překročení časového limitu** – toto hlášení směrovač zasílá, pokud dostane paket, jehož TTL sníží na nulu. Paket je zahozen a směrovač může zdrojovému počítači zaslat chybovou zprávu o překročení časového limitu – **Time Exceeded**.
- **Přesměrování** – toto hlášení může zaslat směrovač počítačům, pokud zná do cílové sítě lepší cestu. Počítač vyšle zprávu obvykle na svou výchozí bránu. Pokud směrovač, který je výchozí bránou, zjistí, že do cíle vede lepší cesta, odešle počítači zprávu o přesměrování. Počítač pak při posílání paketů zvolí tuto lepší cestu.
- **Zpráva o zahlcení** – tento typ zprávy odešle směrovač nebo cílový počítač zdrojovému počítači v případě, že je zahlcen přijímáním a zpracováním zpráv. Zdrojový počítač pak může upravit rychlost odesílání paketů.

Hlavička paketu ICMP

Typ – 8 bitů	Kód – 8 bitů	Kontrolní součet – 16 bitů

Políčko **Typ** může obsahovat následující hodnoty:

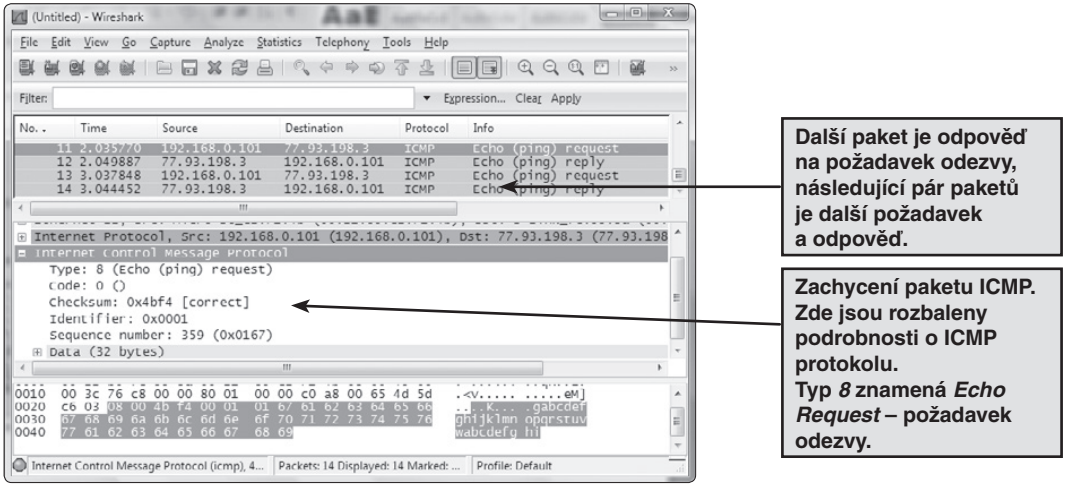
- **0** – **Echo Reply** – odeslání odezvy
- **3** – **Destination Unreachable** – nedostupnost cíle
- **4** – Zahlcení
- **5** – Přesměrování
- **8** – **Echo Request** – požadavek odezvy
- **11** – **Time Exceeded** – překročení časového limitu
- a další …

Políčko **Kód** doplňuje informaci zaslanou v políčku **Typ**.

Je-li například typ = **3** – nedostupnost cíle, pak políčko **Kód** doplňuje některá z následujících informací:

- **0** – síť nedostupná
- **1** – koncové zařízení nedostupné
- **2** – protokol nedostupný

- **3** – port nedostupný
- **4** – fragmentace potřebná, ale nepovolená
- **5** – zdrojová cesta nedostupná
- **6** – cílová síť neznámá
- **7** – cílový počítač neznámý
- **9** – komunikace s cílovou sítí je administrativně zakázána
- **10** – komunikace s cílovým zařízením je administrativně zakázána
- a další …



Na obrázku je zachycen přenos paketů příkazu **ping**.

Byl proveden **ping www.computermedia.cz**.

Došlo ke čtyřem žádostem o odpověď a čtyřem odpovědím (**Echo Request**, **Echo Reply**).

15. Spojová vrstva

Úloha spojové vrstvy

Hlavní úlohou spojové vrstvy je připravit pakety ze síťové vrstvy pro transport na přenosové médium a kontrolovat přístup na přenosové médium.

Spojová vrstva připravuje data získaná z vyšších vrstev pro vyslání na přenosové médium vytvořením datových rámců a přijímá data ze sítě.

Vytváří spojení mezi procesy programového rázu s fyzickým zařízením, na které jsou data vysílána nebo ze kterého jsou přijímána. Typickým zařízením spojové vrstvy je **síťová karta**.

V souvislosti se spojovou vrstvou se lze setkat s následujícími výrazy:

- **uzel** – síťové zařízení připojené na přenosové médium
- **médium** – materiál, který přenáší vysílaná data; patří sem metalické a optické kabely a atmosféra (prostor) pro bezdrátový přenos
- **síť** – fyzické spojení dvou a více uzlů

Během své cesty od zdrojového počítače k cílovému procházejí data různými typy sítí. Aby bylo možné data vysílat na tyto sítě, musí spojová vrstva vždy přizpůsobit datový rámec síti, na kterou bude vysílán. Jinak bude vypadat struktura datového rámce, když bude vysílán po ethernetové lince na lokální síti, jinak, když bude vysílán bezdrátově, a ještě jinak, když bude přenos prováděn do vzdálené sítě například přes satelit.

Spojová vrstva uzpůsobuje datový rámec síti, na niž bude vysílán, a nijak neovlivňuje obsah, který se v rámci skrývá. Spojová vrstva získá paket z vyšší vrstvy a podle typu sítě, na který bude data vysílat, zabalí paket do příslušného rámce.

Tím zajišťuje, že vyšší vrstvy nemusí nijak zajímat, jakými sítěmi jejich datové jednotky poběží, to vše za ně zajistí spojová vrstva.

Kdyby se vyšší vrstvy měly vypořádávat i s tím, jakou sítí jejich datové jednotky půjdou, musely by mít ve svých datových jednotkách mnohem více doprovodných informací, které by se měnily při každé změně sítě. Zde je vidět výhoda vrstevných modelů.

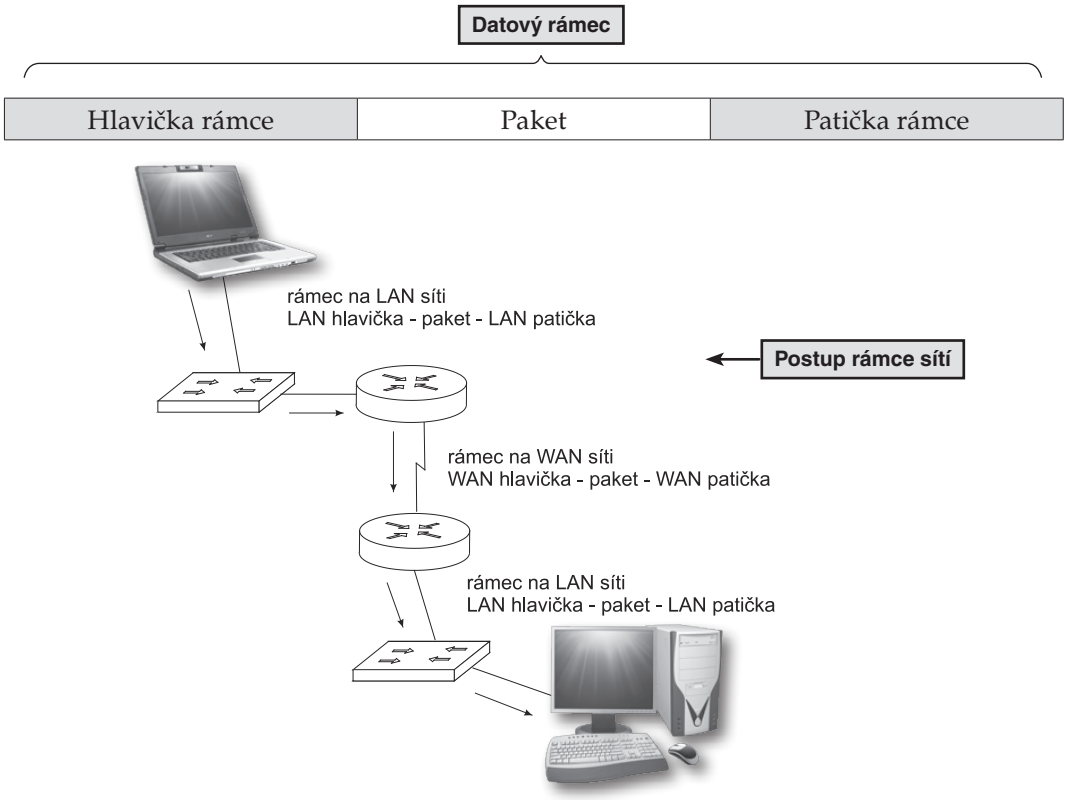
Spojová vrstva kontroluje přístup na médium pomocí různých metod, které zajišťují přístup síťových zařízení na síť a určují, jakým způsobem budou data na přenosové médium vysílána.

Na začátku vysílá zdrojový uzel prostřednictvím své síťové karty svá data na síť. V průběhu cesty mohou data procházet pomocí směrovače mezi sítěmi, například z lokální sítě do WAN sítě. Data přicházející z LAN sítě jsou zapouzdřena do rámce typického pro tuto síť. Směrovač rozbalí rámec, aby zjistil síťovou adresu, pak podle její hodnoty provede

směrovací rozhodnutí, zabalí data do jiného rámce, který bude následně vysílán ze sériového rozhraní na síť WAN, a zajistí toto vyslání.

Zjednodušeně lze říci, že spojová vrstva přidává k paketu určitá data před paket – hlavičku rámce, a jistá data za paket – patičku rámce.

V přídavných informacích datového rámce se mohou vyskytovat informace, jaká zařízení spolu komunikují, kdy může komunikace začít, zda nastaly během přenosu chyby, jaké uzly budou spolu komunikovat příště apod.



Rámec spojové vrstvy

Protože data jsou vysílána na médium ve formě jedniček a nul, je potřeba, aby cílové zařízení mohlo rozlišit, kde jednotlivé datové rámce začínají a kde končí. K této identifikaci se používají speciální vzorky jedniček a nul, podle jejichž výskytu lze poznat začátek a konec rámce.

Struktura rámce se může měnit podle sítě, na kterou je rámec vysílán. K typickým políčkům, která se nacházejí v rámci, patří pole s identifikací začátku a konce rámce,

políčko s adresami, pole určující typ datové jednotky obsažené v rámci, kontrolní políčko obsahující kontrolní údaje a uvnitř rámce zapouzdřený datový paket.

V závislosti na protokolu se mohou měnit políčka v hlavičce a patičce rámce. V hlavičce se může vyskytovat více políček pro zajištění bezproblémovosti vysílání, kvality přenosu, přiměřené rychlosti přenosu a další. V současnosti neexistuje jednotná podoba rámce, která by splňovala požadavky pro přenos všemi přenosovými médii.

V prostředí, kde je potřeba větší kontroly nebo kde hrozí nebezpečí poškození a ztráty dat, například u bezdrátového přenosu, musí rámec obsahovat více kontrolních údajů. Naopak v prostředí, kde přenos probíhá bez problémů, můžete použít rámec s menším počtem kontrolních údajů. Pak přenos probíhá rychleji a efektivněji.

Hlavička rámce

V hlavičce v políčku adres se vyskytuje fyzická adresa (**MAC adresa**) cílového zařízení v lokální síti. Může zde být i fyzická adresa zdrojového počítače.

MAC adresu přiřazuje síťové kartě (síťovému adaptéru) výrobce, neplatí zde hierarchie uplatňovaná při adresování počítačů pomocí IP adres. Počítač s jeho MAC adresou lze bez problémů přenést do jiné sítě. V jedné lokální síti musí mít každé zařízení unikátní MAC adresu.

Při posílání dat má MAC adresa význam pouze v dané lokální síti. Jestliže je cílový počítač mimo danou lokální síť, musí rámec projít přes hraniční zařízení – směrovač. Ten rámec rozbalí, vyhodnotí IP adresy a znovu vytvoří nový rámec s novými MAC adresami. Ty budou představovat odchozí rozhraní směrovače a následující rozhraní, na něž je rámec posílán, tj. MAC adresu dalšího protějšího rozhraní lokálního segmentu, kterým bude rámec putovat. Mimoto se v rámci mohou změnit i další políčka, v závislosti na typu sítě, do které bude rámec dále vysílán, a na jejích požadavcích.

Ve chvíli, kdy počítač přijme rámec, zjistí z cílové MAC adresy, zda je adresován jemu. Pokud ano, rámec rozbalí a protokoly vyšších vrstev provádějí další analýzy a zpracování. Pokud rámec není adresován tomuto počítači, zahodí jej. Počítač může přijmout rámec, který není adresován přímo jemu, například na sdíleném médiu, kde se rámce odesílají všem počítačům v rámci broadcast domény.

V lokální síti se často vyskytuje vysílání typu broadcast, které je určeno všem počítačům v dané lokální síti. Jako cílovou MAC adresu má rámec uvedenu adresu **FF:FF:FF:FF:FF:FF**.

Patička rámce

V patičce rámce jsou přidány informace, které mají podat zprávu o tom, že rámec dorazil do cíle v pořádku, nepoškozen a nezměněn. Tyto informace neslouží k opravě případných chyb, pouze k jejich detekci. V síti běžně dochází k chybám, přeslechům, rušením a podobným skutečnostem, které mohou rámec poškodit.

Políčko pro kontrolu chyb se označuje jako **FCS** – ***Frame Check Sequence***.

Zdrojový počítač provede na posílaných datech kontrolní logický součet označovaný jako **CRC** (***Cyclic Redundancy Check***) a ten vloží do políčka FCS v patičce rámce.

Jakmile data dorazí do cílového počítače, provede se stejný výpočet založený na přijatých datech uložených v rámci a výsledná hodnota CRC se porovná s hodnotou v políčku FCS.

Jestliže hodnoty nesouhlasí, rámec se zahodí.

Protokoly spojové vrstvy

Mezi protokoly spojové vrstvy patří **Ethernet**, **PPP** (***Point-to-Point Protocol***), **Frame Relay**, **HDLC** (***High-Level Data Link Control***), **ATM** (***Asynchronous Transfer Mode***).

Jiné protokoly (například **Ethernet, 802.11** pro bezdrátové vysílání) se používají na LAN sítích, kde je nutná vysoká přenosová rychlost, a jiné na WAN sítích, kde se data přenášejí podstatně menší rychlostí (například protokoly HDLC, PPP, Frame Relay).

Služby a protokoly spojové vrstvy jsou popsány organizacemi **ISO** (***International Organization for Standardization***), **IEEE** (***Institute of Electrical and Electronics Engineers***), **ANSI** (***American National Standards Institute***), **ITU** (***International Telecommunication Union***) a jinými komunikačními společnostmi.

První z nich nastavují a popisují veřejné otevřené protokoly a standardy. Komunikační společnosti si mohou vytvářet vlastní protokoly a standardy, aby lépe využily nových možností aktuálních technologií.

Protokoly spojové vrstvy nefungují pouze softwarově jako protokoly vyšších vrstev, ale jsou součástí hardwaru síťových adaptérů připojujících zařízení k síti, síťových karet i bezdrátových adaptérů.

Společnost ISO stojí za definicí standardu **HDLC** (***High Level Data Link Control***), **IEEE** definuje standardy **802.2** (**LLC**), **802.3** (**Ethernet**), **802.5** (***Token Ring***), **802.11** (**bezdrátové LAN**), společnost **ITU** stanovuje standardy **Q.922** (***Frame Relay***), **Q.921** (***ISDN standard***), **HDLC**

(*High Level Data Link Control*) a společnosti **ANSI** například standard **ADCCP** (*Advanced Data Communications Control Protocol*).

Weby těchto společností zabývající se nejen protokoly a standardy spojové vrstvy najdete na adresách www.iso.org, www.ieee.org, www.ansi.org, www.itu.int.

Podvrstvy LLC, MAC

Spojová vrstva se často rozděluje na dvě podvrstvy – **LLC** (**Logical Link Control**) a **MAC** (**Media Access Control**) pro snadnější vysvětlení funkčnosti.

Vrchní podvrstva LLC zajišťuje identifikaci protokolu ve vnořeném paketu, kontrolu chybovosti a řízení toku. Je hardwarově nezávislá.

Nižší podvrstva MAC zajišťuje adresování rámce pomocí fyzických adres (MAC adresy) a označení začátku a konce rámce. Řídí přístup k médiu a je hardwarově závislá.

Díky tomu, že existuje více různých protokolů ve spojové vrstvě, existují i různé metody kontroly přístupu k médiu. U různých médií platí různá pravidla přístupu na médium, kterými se musí vysílající uzly řídit. Někdy se vysílání na médium jeví jako vysoce řízený proces, kdy jednotlivé uzly „dostávají slovo", jindy vysílají nahodile podle potřeby.

Výběr metody pro přístup na médium závisí na tom, jakým způsobem sdílí uzly přenosové médium a v jaké topologii jsou uzly umístěny.

Řízení přístupu na sdílené médium

Základní dvě metody pro přístup na sdílené přenosové médium jsou deterministická metoda a nedeterministická metoda. Sdílené médium býval dříve koaxiální kabel, později nahrazený UTP kabelem připojujícím počítače k rozbočovači (hubu), v bezdrátovém přenosu je sdíleným médiem atmosféra (prostor).

Deterministická metoda spočívá v tom, že každý uzel má určený jistý čas pro vysílání. Přístup je kontrolován a řízen. Jestliže uzel nepotřebuje vysílat, předá tuto možnost dalšímu zařízení v pořadí. Během vysílání jednoho uzlu nesmí ostatní uzly vysílat, musí počkat, dokud data nedostane cílové zařízení. Tato metoda může být nevýhodná v tom, že přenosová linka nemusí být optimálně využita, mohou zde nastávat chvíle, kdy počítač sice může vysílat, ale nevysílá a linku nevyužívá.

Deterministickou metodu využívá technologie **Token Ring**.

Nedeterministická metoda umožňuje uzlům soupeřit o vysílání na sdílené médium. Každé zařízení, které má potřebu vysílat, se o to může pokusit. Nemusí čekat, až na ně přijde řada.

Aby nedocházelo k zahlcení linky a přenosovým problémům, používá tato metoda mechanismus **CSMA** – ***Carrier Sense Multiple Access***, kterým dokáže detekovat provoz na síti a případné kolize.

Uzly mající zájem vysílat na sdílené médium nejprve poslouchají, zda na médiu nedochází k vysílání. Pokud na médiu detekují vysílání, čekají. Po nějaké chvíli znovu provedou test, zda je na médiu klid. Jestliže na médiu žádný provoz nedetekují, zkusí vysílat.

Nedeterministickou metodu používá technologie **Ethernet** a bezdrátové vysílání.

Přesto se může stát, že dva různé uzly začnou vysílat ve stejnou chvíli. Na síti tak dojde ke kolizi a zničení obou signálů.

CSMA/CD, CSMA/CA

CSMA proces je doplněn metodou **CD** nebo **CA**.

CSMA/CD – Collision Detection (detekce kolizí)

Tuto metodu využívá technologie **Ethernet**. Zařízení využívající metodu **CSMA/CD** čekají, až je na sdíleném přenosovém médiu klid, a pak začnou vysílat. To může vést ke kolizím signálů, které byly vyslány na sdílené médium ve stejnou chvíli. Po takové kolizi musí být data odeslána znovu.

CSMA/CA – Collision Avoidance (předcházení kolizí)

Tuto metodu využívá bezdrátové vysílání. Jestliže zařízení chce vysílat, poslouchá, zda je na sdíleném médiu klid (u bezdrátového vysílání je médiem atmosféra). Pokud ano, vyšle informaci pro ostatní uzly, že bude vysílat. Tím brání tomu, aby nedocházelo ke kolizím signálů přenášejících data.

Může dojít jen ke kolizi informací o záměru vysílat.

Pokud již probíhá vysílání jiným zařízením, řídí se zájemce o vysílání tzv. backoff algoritmem, s jehož pomocí vybírá náhodnou dobu, po kterou čeká, než se znovu pokusí o vysílání. Fakt, že se jedná o náhodně dlouhou dobu, snižuje nebezpečí kolize dalšího vysílání.

Kontrola přístupu na nesdílené médium

Protokoly pro kontrolu přístupu na nesdílené médium nemusí tolik sledovat, jaký je stav přenosové linky, než se začne vysílat. Příkladem vysílání na nesdílené médium je vysílání u topologie typu **point-to-point** (bod–bod). V tomto případě nedochází ke sdílení přenosového média, a tím se tedy zabrání vzniku kolizí.

V případě zapojení typu **point-to-point** se mohou protokoly spojové vrstvy rozhodnout, zda se bude vysílat pomocí **full-duplexu**, nebo **half-duplexu**.

Full-duplex, half-duplex

Full-duplexní provoz znamená, že v danou chvíli mohou vysílat i přijímat oba uzly na lince typu **point-to-point**.

Full-duplex lze použít u spojení počítače a přepínače, spojení dvou přepínačů nebo spojení dvou počítačů kříženým kabelem. U spojení pomocí kabelu UTP na 100 Mbps Ethernetu dochází k vysílání jedním párem vodičů a současně k přijímání signálu druhým párem vodičů.

Half-duplexní provoz umožňuje v danou chvíli vysílat jen jednomu zařízení, druhé může jen přijímat data. Až skončí vysílání prvního uzlu, může začít vysílat druhý uzel.

Pomocí protokolů spojové vrstvy se obě zařízení před započetím vysílání dohodnou, jaký typ vysílání budou používat. Pokud se dohodnou na full-duplexu, vysílají tímto způsobem. Jestliže jeden z nich nedokáže tento typ používat, dohodnou se na half-duplexu.

Ethernet

Ethernet je technologie používaná na LAN sítích. Existuje mnoho druhů **Ethernetu**, které se liší přenosovou rychlostí – **od 10 Mbps až po 10 Gbps**.

Rámec Ethernetu je ve všech případech podobný, odlišnosti jsou v systému umísťování dat na přenosové médium.

Ethernet je definován pomocí standardů **802.2** a **802.3**.

Data se posílají prostřednictvím sdíleného média pomocí metody **CSMA/CD**, proto rámce musí obsahovat zdrojovou a cílovou MAC adresu.

MAC adresa je 48bitové číslo znázorňované pomocí hexadecimální soustavy.

Na sítích typu TCP/IP se pro přenos používá rámec typu **Ethernet II**.

Rámec Ethernet II

Preambule	Cílová adresa	Zdrojová adresa	Typ	Data	FCS
8 bytů	6 bytů	6 bytů	2 byty	46–1500 bytů	4 byty

Preambule – slouží k synchronizaci mezi zdrojovým a cílovým zařízením. Začíná 7 byty obsahujícími střídající se jedničky a nuly (7x 10101010). Nakonec je odvysílán 1 byte obsahující 10101011, který označuje začátek rámce – tzv. **SFD** (***Start of Frame Delimiter***).

Cílová adresa – 48bitová MAC adresa cílového zařízení. Může obsahovat adresu jednotlivého zařízení – typ **unicast**, skupiny zařízení – typ **multicast**, nebo všech zařízení – typ **broadcast**.

Zdrojová adresa – 48bitová MAC adresa zdrojového zařízení.

Typ – pole označující typ protokolu vyšší vrstvy, který přijme data poté, co bude ethernetový rámec rozbalen a zpracován.

Data – obsahují obvykle paket **IPv4**. Minimální délka dat je 46 bytů, kratší jsou považovány za zbytky po kolizích v síti. Maximální délka je 1 500 bytů.

FCS (***Frame Check Sequence***) – kontrolní hodnota, která slouží k posouzení, zda byl rámec doručen bez poškození a změn.

PPP

PPP – ***Point-to-Point Protocol***

Tento protokol je používán na WAN sítích pro sériové spojení mezi dvěma uzly. Spojení může být realizováno různými způsoby – metalicky, opticky, bezdrátově.

Na začátku vysílání mezi dvěma uzly pomocí PPP je vytvořeno logické spojení, které může zahrnovat autentizaci a dohodnutí parametrů, jako typ komprese, šifrování a využití více přenosových linek.

Rámec protokolu PPP

Začátek	Adresa	Kontrola	Protokol	Data	FCS
1 byte	1 byte	1 byte	2 byty	0–více bytů	2 nebo 4 byty

Začátek – indikuje začátek rámce, skládá se ze vzorku 01111110.

Adresa – obsahuje adresu broadcastu v sítích **PPP. PPP** nemá potřebu přidělovat koncovým zařízením konkrétní adresy.

Kontrola – obsahuje binární vzorek 00000011, který vyjadřuje, že data mají být posílána v rámci bez rozdělení do sekvencí.

Protokol – číslo určující, jaký typ paketu je vnořen uvnitř rámce.

Data – data obsažená uvnitř rámce ve formátu protokolu uvedeného v poli protokol.

FCS (***Frame Check Sequence***) – obvykle dvoubytové políčko udávající kontrolní součet, který slouží k posouzení, zda byl rámec doručen bez poškození a změn.

802.11 Protokol pro bezdrátové vysílání

Tento protokol používá adresování pomocí 48bitových fyzických adres typických pro LAN sítě.

U bezdrátového přenosu může nastávat více problémů s doručením dat, častější rušení signálu a poškození dat, proto rámec protokolu **802.11** musí obsahovat více kontrolních políček, která zajistí správné doručení.

Standard **802.11** je často nazýván **Wi-Fi**.

Během vysílání dat se používá mechanismus **CSMA/CA**, který byl již popsán dříve.

Po odvysílání rámce čeká zdrojové zařízení na potvrzení o přijetí od cílového zařízení. Pokud takové potvrzení nedostane, vyšle data znovu.

802.11 podporuje autentizaci – ověření komunikujících stran, asociaci (připojení) s přípojným bodem a zabezpečení přenosu pomocí šifrování.

Rámec protokolu 802.11

Verze protokolu	Typ	Podtyp	K DS	Od DS	Více fragmentů	Opětovné vysílání dat	Úsporný režim	Další data	WEP šifrování	Pořadí	Doba vysílání / ID stanice	Cílová adresa	Zdrojová adresa	Adresa přijímače	Číslo fragmentu	Číslo sekvence	Adresa vysílače	Tělo rámce	FCS
2 bity	2 bity	4 bity	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	1 bit	2 byty	8 bytů	8 bytů	8 bytů	4 bity	12 bitů	8 bytů	0 až 2312 bytů	4 byty

Verze protokolu je verze použitého rámce **802.11**.

Typ a **podtyp** identifikují typ rámce – kontrolní, datový, řídící.

K DS – nastaveno na **1**, pokud rámec cestuje do distribučního systému.

Od DS – nastaveno na **1**, pokud rámec cestuje z distribučního systému.

Více fragmentů – nastaveno na **1**, pokud má rámec další fragmenty.

Opětovné vysílání dat – nastaveno na **1**, pokud se jedná o nové vysílání již dříve odvysílaných dat.

Úsporný režim – nastaveno na **1**, pokud se uzel chystá přejít do úsporného režimu.

Další data – nastaveno na **1**, pokud je potřeba uzlu v úsporném režimu sdělit, že jsou pro něj připravena další data.

WEP šifrování (*Wired Equivalent Privacy*) – nastaveno na **1**, pokud rámec obsahuje šifrovaná data.

Pořadí – nastaveno na **1**, pokud se jedná o datový rámec, který nepotřebuje přeuspořádat

Doba vysílání / ID stanice – v závislosti na typu rámce obsahuje informaci o době potřebné k odvysílání rámce nebo přiřazenou identitu stanice, která rámec odvysílala.

Cílová adresa je fyzická MAC adresa cílového zařízení v síti.

Zdrojová adresa je fyzická MAC adresa zdrojového zařízení, které vyslalo rámec.

Adresa přijímače je fyzická MAC adresa zařízení, které přijme rámec jako další v pořadí.

Číslo fragmentu je číslo určující pořadí fragmentů daného rámce.

Číslo sekvence určuje číslo sekvence přiřazené rámci.

Adresa vysílače je fyzická MAC adresa zařízení, které právě vyslalo rámec.

Tělo rámce obsahuje přepravované informace, většinou IP paket.

FCS (*Frame Check Sequence*) – číslo sloužící ke zjištění, zda data dorazila bez porušení a změny, výsledek operace **CRC** (*Cyclic Redundancy Check*).

Struktura bezdrátové sítě

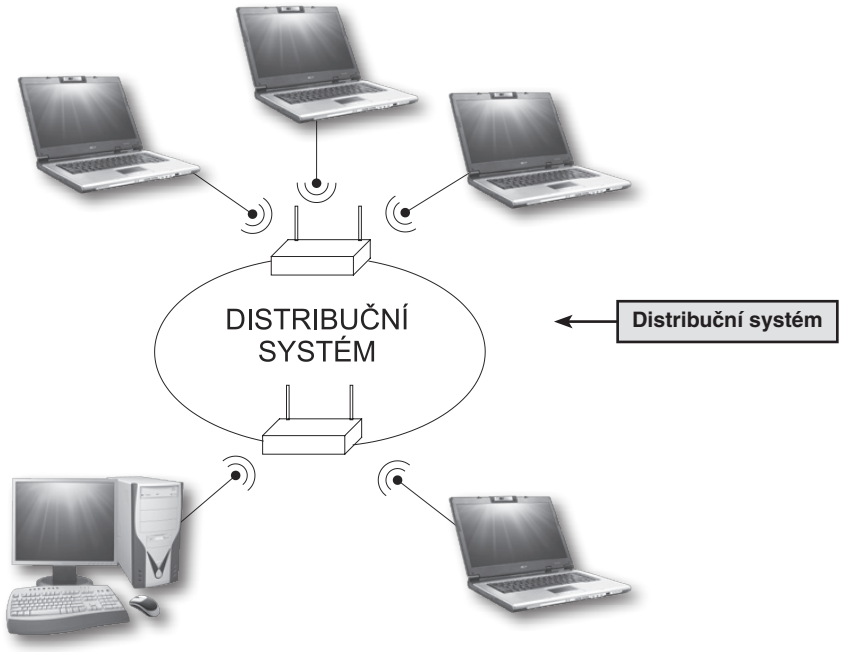
V síti používající bezdrátový přenos se vyskytují počítače, jimž přístup do sítě umožňuje bezdrátová síťová karta nebo bezdrátový síťový adaptér (**USB**, **PCMCIA**). Počítače mohou spolu komunikovat přímo nebo pomocí přípojného bodu – tzv. **access pointu**.

Síť skládající se z několika počítačů komunikujících bezdrátově bez asistence access pointu se někdy nazývá bezdrátová **síť typu ad-hoc**.

Základní bezdrátová síť se skládá z jednoho access pointu, ke kterému se přihlašují jednotlivé bezdrátové počítače, jimž zprostředkovává komunikaci.

Access point zajišťuje komunikaci nejen mezi počítači s bezdrátovým vysíláním, ale i bezdrátové sítě s metalickou sítí, například když počítač z bezdrátové sítě potřebuje komunikovat s nějakým jiným počítačem na metalické síti.

Distribuční systém je spojení několika základních bezdrátových sítí. Spojení je většinou uskutečněno pomocí metalických rozvodů, ale může být zajištěno také bezdrátově.



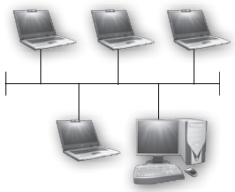
16. Topologie

Topologií v síti se rozumí vzájemné uspořádání síťových zařízení. Je možné na topologii nahlížet z pohledu fyzického uspořádání nebo logického uspořádání, kde se jedná o způsob vysílání na síť.

Fyzická topologie

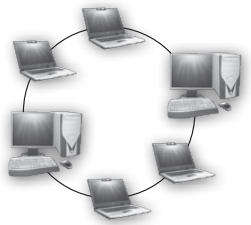
Jde o způsob zapojení síťových zařízení mezi sebou.

Sběrnicová topologie



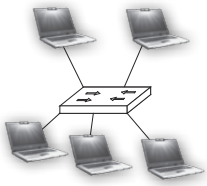
Toto uspořádání bývalo typické pro zapojení počítačů do sítě pomocí koaxiálního kabelu. Všechny počítače sdílejí jedno společné přenosové médium a jsou součástí jedné kolizní domény. Konec kabelu je zakončen ukončovacím článkem – terminátorem.

Kruhová topologie



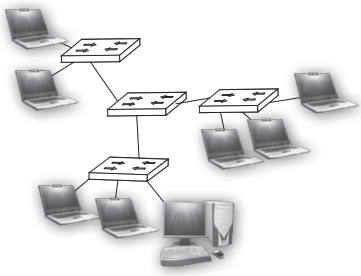
Počítače jsou zapojeny do kruhu jeden k druhému. Data procházejí všemi počítači mezi zdrojovým a cílovým a posílají se jedním směrem. Pokud dojde k problému s jedním uzlem, nastává problém s přenosem.

Topologie typu hvězda



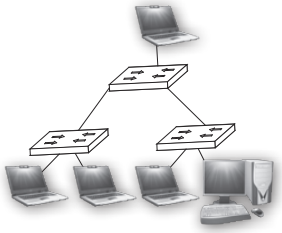
Jednotlivé počítače jsou zapojeny pomocí kabelů k centrálnímu bodu, jímž může být rozbočovač (hub) nebo přepínač (switch). Výhodou je, že pokud zkolabuje jeden počítač nebo nastane problém s jednou přenosovou linkou, neohrozí to funkčnost ostatní sítě.

Topologie rozšířená hvězda



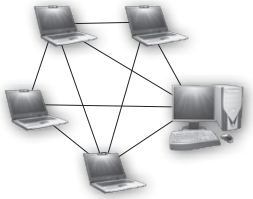
Tento způsob zapojení lze provést pomocí rozbočovačů nebo přepínačů. Několik segmentů typu hvězda je spojeno dohromady pomocí rozbočovače nebo přepínače. V případě rozbočovačů je potřeba dát pozor na příliš velkou rozlehlost takové sítě, kdy by mohlo docházet k pozdním kolizím, které by byl problém detekovat. U přepínačů tento problém mizí, každá dvě zařízení komunikují mezi sebou, aniž by jejich provoz omezoval ostatní počítače.

Hierarchická topologie



Hierarchická topologie je podobná topologii rozšířené hvězdy, jen s tím rozdílem, že na vrcholu stromu je umístěn počítač, který kontroluje provoz v síti. Spojení počítačů je provedeno stejně jako u topologie rozšířené hvězdy pomocí rozbočovačů nebo lépe přepínačů.

Topologie mesh (síť)



Při tomto druhu zapojení je každý počítač spojen s každým přímou linkou. Pro vytvoření úplné topologie **mesh** by bylo potřeba velké množství spojovacích linek, které by neúměrně rostlo s dalšími zařízeními přidanými do sítě. Proto se nepoužívá úplná, ale pouze částečná topologie mesh. Některé linky se vynechávají.

Logická topologie

V případě sítě se sdíleným médiem se jedná o mnohonásobný přístup počítačů na toto médium.

Pokud jde o výhradní spojení mezi dvěma počítači, je to spojení **point-to-point**.

Mnohonásobný přístup na sdílené médium

Z pohledu logické topologie existují v síti se sdíleným médiem dvě základní topologie – **broadcast** a **token passing**. Liší se přístupem k vysílání dat na sdílené médium.

Broadcast (vysílání)

Princip této topologie spočívá v tom, že všechna zařízení v síti jsou si rovna, a pokud chtějí vysílat, mohou se o to pokusit.

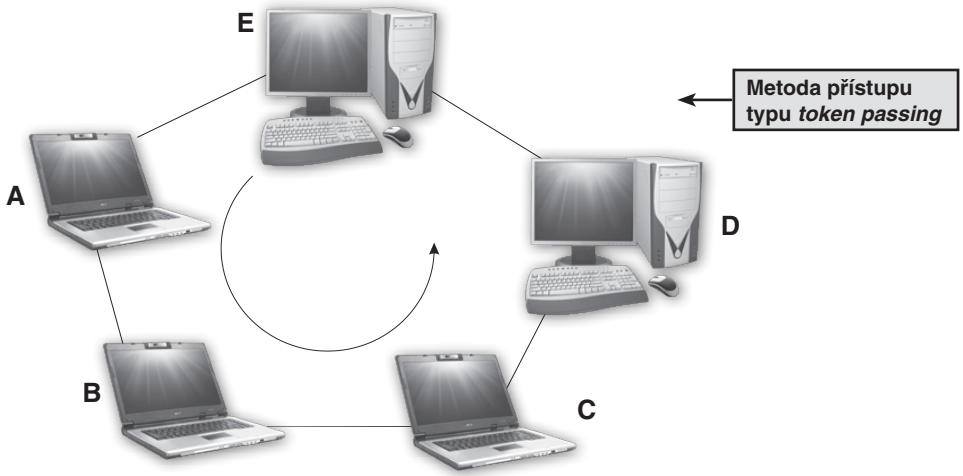
Všechna zařízení v síti, která mají zájem vysílat, naslouchají, zda je na přenosovém médiu klid. Jestliže ano, mohou vysílat. Kdo začne vysílat jako první, obvykle uspěje.

V síti může docházet ke kolizím, po jejichž vzniku nastává chvíle klidu, a poté se počítače znovu pokusí data odvysílat. Neexistuje žádná přednost ve vysílání.

Typicky používá tuto logickou topologii technologie **Ethernet**.

Kruhová topologie, token passing

Každý uzel mezi zdrojovým a cílovým počítačem dostane data, která odeslal zdrojový počítač, a pokud nejsou určena jemu, pošle je dál. Tak to probíhá až do chvíle, kdy data najdou svůj cíl.



Síťová zařízení si předávají elektronický poukaz pro vysílání. Každé má pro své vysílání vymezen čas. Pokud zařízení nemá nic k vysílání, předá poukaz dále.

Tato technika kontroly přístupu na síť se nazývá **token passing** („předávání peška").

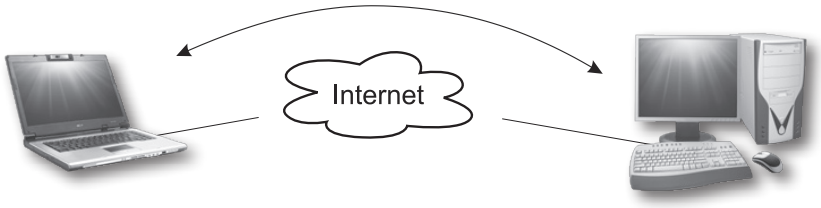
Tuto logickou topologii používají technologie **Token Ring** a **FDDI** (***Fiber Distributed Data Interface***).

Fyzicky může být tato topologie jiná, počítače nemusí být ve skutečnosti rozmístěny a spojeny do kruhu. Jedná se o způsob předávání vysílaných dat.

Point-to-point (bod–bod)

Pokud počítače svým vysíláním nezasahují do provozu ostatních počítačů v síti, jedná se o logické spojení typu **point-to-point** (bod-bod).

Počítače, které komunikují výhradně spolu (a nedochází k šíření dat i k jiným počítačům v síti), jsou propojeny v logické topologii **point-to-point**. Neznamená to, že by musely být propojeny fyzicky jednou linkou. Může mezi nimi být řada jiných síťových zařízení, ale protože komunikují výhradně mezi sebou, jedná se o topologii **point-to-point**. Počítače mají mezi sebou vytvořen virtuální okruh.



17. Fyzická vrstva

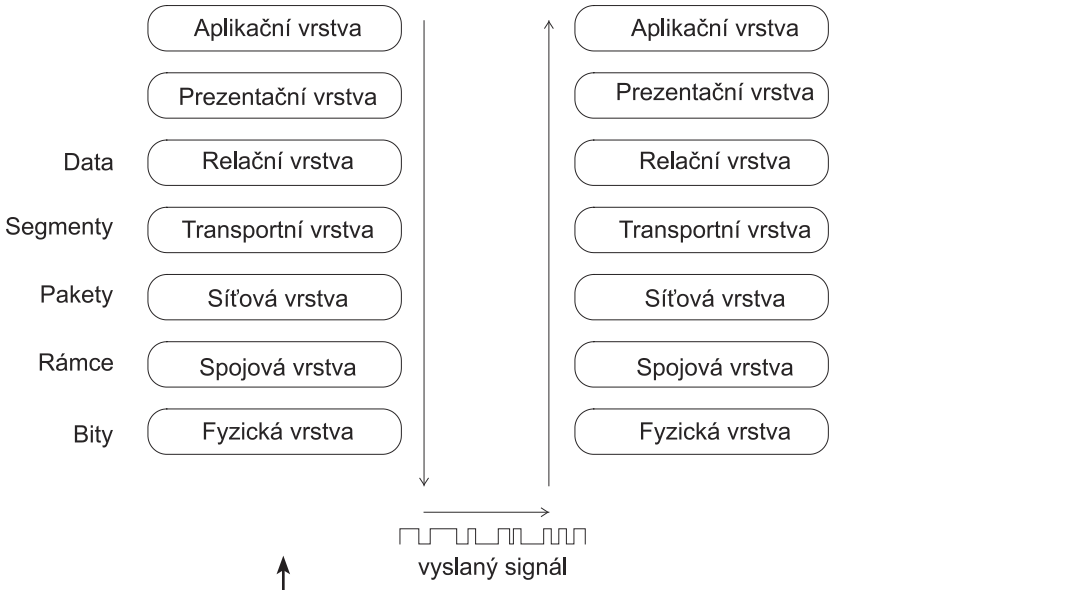
Úloha fyzické vrstvy

Úlohou fyzické vrstvy je zajistit vysílání dat na fyzické médium. Fyzická vrstva přebere datový rámec z vyšší, spojové vrstvy, překóduje binární vyjádření rámce do signálu a ten následně odvysílá na médium. Na protější straně převede signál získaný z přenosového média na digitální kód – jedničky a nuly – a předá jej ke zpracování spojové vrstvě.

Fyzická vrstva vysílá data na přenosové médium příslušným způsobem – jinak na metalické rozvody, jinak na optické a jinak v případě bezdrátového přenosu.

Tato vrstva zajistí převedení binárního kódu, který reprezentuje datový rámec, do příslušného optického, elektrického nebo rádiového signálu a ten odvysílá na příslušné přenosové médium. Přijímač tento signál z přenosového média převede zpět do digitálního kódu, který dokáže dále zpracovávat spojová vrstva a vyšší vrstvy.

Fyzická vrstva pracuje na úrovni přenosových médií, konektorů a neinteligentních síťových prvků, například rozbočovačů, které žádným způsobem neřeší obsah přenášeného signálu.



Příprava dat pro přenos probíhá postupně, data se v jednotlivých vrstvách zabalují do dalších přidaných informací a následně se provede vysílání dat formou příslušného signálu na přenosové médium.
Na druhé straně je signál převeden zpět do digitální podoby a zpracován postupně všemi vrstvami od nejnižší po nejvyšší.

Podle vybraného přenosového média je zvolen způsob vysílání digitálního signálu na síť. Pro přijímač je vyslaný rámec sada impulzů – elektrických, optických nebo rádiových. Aby přijímač mohl odlišit, kde jeden rámec končí a druhý začíná, je k vyslanému rámci přidáván ještě určitý vzorek signálu, kterým se označuje hranice rámce.

Signál vysílaný formou optického signálu se skládá z impulzů světla. Zjednodušeně lze říci, že svítí-li světlo, znamená to jedničku, a naopak, nesvítí-li – znamená to nulu.

U přenosu elektrickým signálem po metalických rozvodech se jedničky a nuly kódují pomocí změny napětí na kabelu.

U bezdrátového přenosu se signál kóduje pomocí různých typů modulací – frekvenční, amplitudové nebo fázové (**FM**, **AM**, **PM**).

Fyzická vrstva zahrnuje fyzické součásti sítě, zakódování dat do signálu vhodného pro přenos médiem a vlastní přenos signálu.

Standardy

Stejně jako u vyšších vrstev, i v této nejnižší fyzické vrstvě existují určitá pravidla, která popisují, jak by měl hardware fungovat.

Tyto standardy pro použité technologie vytváří například organizace **ISO** (***International Organization for Standardization***), **ANSI** (***American National Standards Institute***), **IEEE** (***Institute of Electrical and Electronics Engineers***), **ITU** (***International Telecommunication Union***), **EIA/TIA** (***Electronics Industry Alliance/Telecommunications Industry Association***), **FCC** (***Federal Communication Commission***).

Dodržování pravidel zajišťuje vzájemnou kompatibilitu nejrůznějších zařízení od různých výrobců. Můžete pak vyměnit jedno zařízení za jiné, aniž byste museli měnit konektory nebo kabeláž.

Dodržování standardů má vliv na hardwarové provedení výrobků.

Standardizuje se, jak mají vypadat konektory, jaké elektrické a fyzické vlastnosti mají mít přenosová média, jak bude na médiu vypadat signál reprezentující přenášená data a jak budou vypadat kontrolní signály.

Díky standardizaci signálů mohou různá zařízení vzájemně komunikovat. Standardizace koncovek a kabelů zase zaručuje, že kabel lze použít s danou koncovkou i na jiném zařízení, které má pro tuto koncovku port, ačkoliv jsou vyrobeny různými výrobci.

Kódování

Kódování je převedení digitálních dat do předem definovaného kódu. Tato skupina bitů vytvářející určitý předvídatelný vzorek je rozpoznatelná oběma stranami, vysílačem i přijímačem.

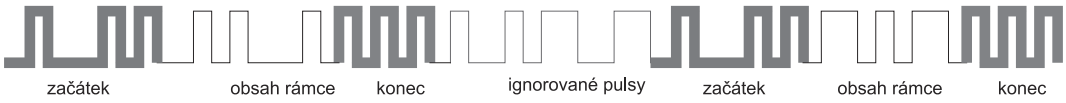
Takový vzorek bitů reprezentující data lze odlišit od kontrolních nebo nahodilých vzorků bitů. Fyzická vrstva může vytvářet kontrolní vzorky bitů, které označují začátek a konec rámců.

Kódování pomocí vytváření skupin bitů

Předtím než je skupina bitů rámce odvysílána na přenosové médium, je opatřena na svém začátku a konci určitou skupinou bitů, které symbolizují začátek a konec rámce. Tyto skupiny jsou přiřazeny buď ve spojové, nebo ve fyzické vrstvě.

Přijímací zařízení je pak schopno odlišit takovou skupinu bitů od nahodilých impulzů, které mohou vznikat různým typem rušení nebo vlivem kontrolních pulzů.

Seskupování do skupin bitů umožňuje vyšší přenosové rychlosti, protože lze snáze detekovat přenosové chyby. Pokud přenášený rámec nezačíná a nekončí předepsaným vzorkem bitů, přijímací zařízení jej bude ignorovat a nebude jej předávat ke zpracování spojové vrstvě.



Pro lepší schopnost indikování chyby přenosu se vždy určitý počet bitů nahradí jinou skupinou bitů nazývanou **symbol**.

Například symbol **10101** může nahrazovat sadu bitů **1110**.

Tento typ kódování pomáhá v tom, že se na přenosové médium nevysílá příliš dlouhá spojitá řada jedniček nebo nul. Taková řada je zakódována příslušným způsobem do symbolu, ve kterém se dlouhé spojité řady úmyslně nevyskytují. Tím je zajištěno, že obě komunikující zařízení udrží vzájemnou synchronizaci a nebude docházet k chybné interpretaci přenášených dat.

V symbolech se vyskytují nějaké bity navíc oproti původnímu signálu, což zatěžuje linku o něco více, než kdyby se vysílala jen původní data. Je to ale ochrana proti chybám.

Dále díky optimálnímu kódování do symbolů klesá spotřeba energie potřebné pro vysílání. V symbolech jsou počty jedniček a nul vyvážené. Kdyby se například na optickém typu vysílání vysílací lasery a přijímací fotodiody přetěžovaly vysíláním převážné většiny jedniček, docházelo by nejen k vyšší spotřebě, ale i k vyšší chybovosti.

Symboly slouží k přenášení vlastních dat umístěných v rámci, mohou označovat hranice rámců nebo signalizovat nečinné přenosové médium. Jestliže přijímač přijme symboly, které nejsou ani datové, ani kontrolní, považuje je za chybné a nezabývá se jimi. Mohou to být symboly obsahující velké množství bitů stejného typu.

Kódování pomocí 4B/5B

Pomocí tohoto typu kódování se nahradí 4 bity dat pětibitovými symboly. Zajistí se tak, že alespoň jeden bit v symbolu bude jiný než ostatní čtyři bity, a tím se zabezpečí synchronizace vysílače a přijímače.

Šestnáct symbolů slouží ke kódování šestnácti možností čtyřbitových skupin bitů, zbývajících šestnáct symbolů je určeno ke kontrolním účelům, jako je označení volné linky, označení začátku a konce vysílání, označení vysílací chyby. Některé symboly jsou nevyužité a jsou považovány za chybné.

4B – data	5B – symbol
0000	11110
0001	01001
0010	10100
0011	10101
0100	01010
0101	01011
0110	01110
0111	01111
1000	10010
1001	10011
1010	10110
1011	10111
1100	11010
1101	11011
1110	11100
1111	11101

Účel	5B – symbol
Volná linka	11111
Začátek vysílání	11000
Začátek vysílání	10001
Konec vysílání	01101
Konec vysílání – reset	00111
Chyba přenosu	00100
Chybný symbol	00000
Chybný symbol	00001
Chybný symbol	00010
Chybný symbol	00011
Chybný symbol	01100
Chybný symbol	00101
Chybný symbol	00110
Chybný symbol	01000
Chybný symbol	10000
Chybný symbol	11001

Toto kódování je využitelné na **100Mbps Ethernetu**. Na vyšších přenosových rychlostech se nahrazuje jinými, podobnými typy kódování.

Signalizace

Signalizace je metoda převedení digitálního vzorku dat do signálu, který je dále šířen pomocí přenosových médií různých typů – optických nebo metalických kabelů nebo bezdrátovým přenosem. Jakým způsobem jsou signalizovány bity obsahující jedničku nebo nulu, definují standardy fyzické vrstvy. Každý bit z datového rámce je vyslán na

médium jako jednotlivý signál. Vyslání celého rámce pak spočívá v odvysílání všech bitů rámce jako sady signalizačních impulzů.

Doba, po kterou se vysílá jeden bit, se nazývá **bit time**. Aby mohla protější zařízení spolu úspěšně komunikovat, musí si sladit rychlost vysílání a přijímání bitů, musí se synchronizovat.

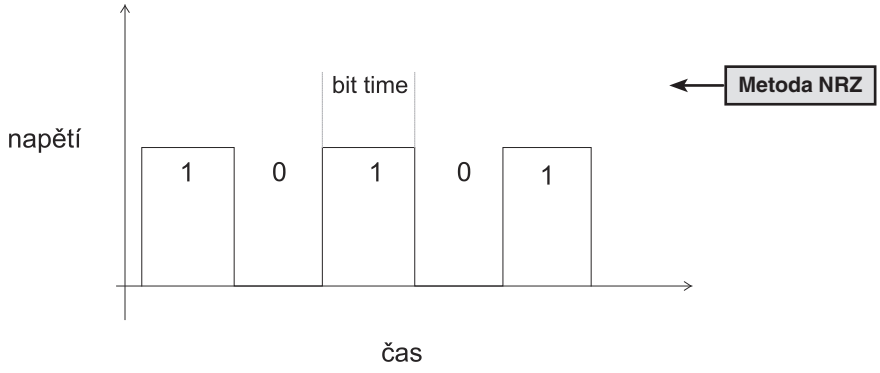
Přijímací zařízení přijme na úrovni fyzické vrstvy sadu bitů, vyhledá vzorky bitů typické pro označení začátku a konce rámce a pak předá celý rámec spojové vrstvě.

Způsob reprezentace jedniček a nul při vysílání závisí na zvolené signalizační metodě.

Než začnou protější síťová zařízení mezi sebou přenášet data, musí se nejprve dohodnout, jakým způsobem budou přenášet jedničku a jakým nulu.

Metoda NRZ

Nula může být signalizována určitým nízkým napětím a jednička zvýšením tohoto napětí – takto pracuje signalizační metoda **NRZ** (***Non-Return to Zero***). Je použitelná pro přenosy s menší přenosovou rychlostí, přenos je náchylnější k rušení elektromagnetickým polem. Při delších sadách samých jedniček nebo nul může dojít k nesprávné identifikaci počtu těchto bitů, protože při jejich odvysílání nedochází ke změnám v napětí.

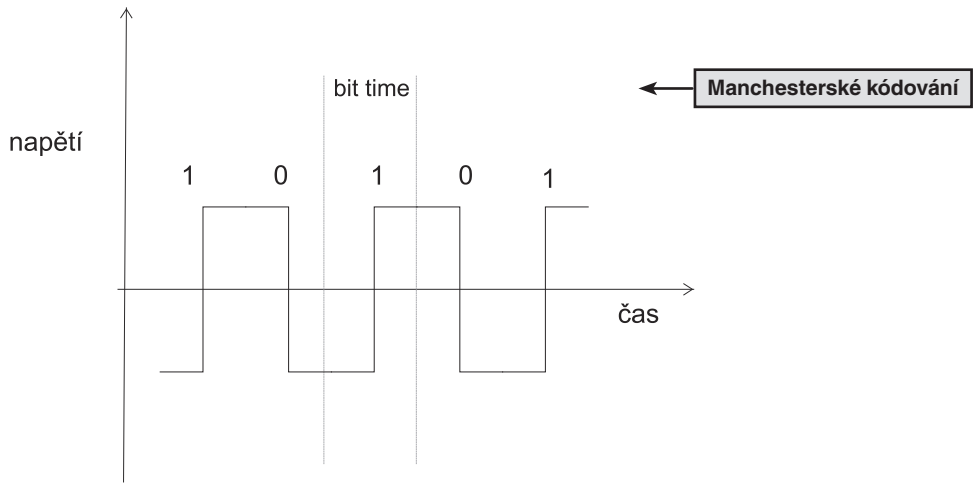


Signalizace pomocí Manchesterského kódování

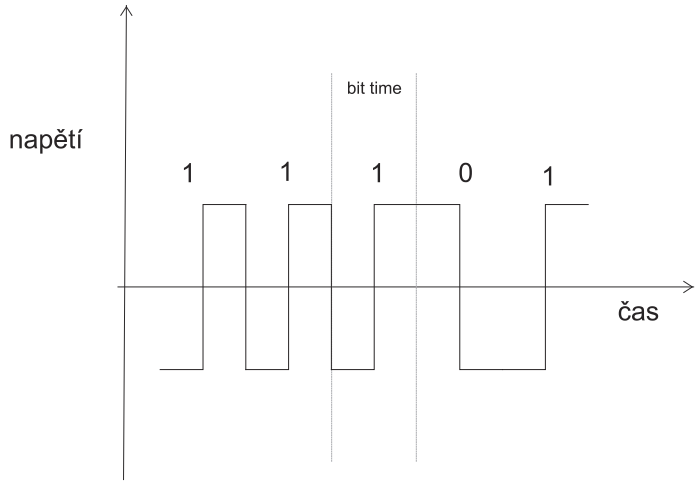
Jiný typ signalizace signalizuje jedničku zvýšením napětí a nulu snížením napětí během doby vyhrazené pro odvysílání jednoho bitu – takto pracuje **Manchesterské kódování**.

Změna probíhá v polovině doby vyhrazené pro odvysílání jednoho bitu.

Díky tomu, že jednička i nula jsou symbolizovány změnou napětí, může průběžně probíhat úprava synchronizace mezi zdrojovým a cílovým zařízením.



Jestliže je potřeba odvysílat dvě jedničky, pak po první jedničce symbolizované zvýšením napětí musí na hranici bitů hodnota napětí okamžitě klesnout na dolní mez, aby následující jednička mohla být opět reprezentována zvýšením napětí.



Manchesterská metoda signalizace není efektivní pro použití u vyšších přenosových rychlostí, ale na rychlosti 10 Mbps se využívala.

Přenosová kapacita

U měření přenosové rychlosti se můžete především v anglických materiálech setkat s pojmy jako bandwidth, throughput a goodput. Často se tyto anglické názvy používají i v české terminologii.

Bandwidth – šířka pásma

Bandwidth (neboli **šířka pásma**) označuje maximální možnou kapacitu přenosové linky. Vyjadřuje, jaké množství dat lze za jednotku času teoreticky přepravit.

Šířka pásma se obvykle měří v kilobitech za sekundu – **kb/s** (**kbps** – ***kilobit per second***) nebo megabitech za sekundu – **Mb/s** (**Mbps** – ***megabit per second***).

Základní (nejmenší) jednotka šířky pásma je **b/s** (**bps** – ***bit per second***).

Vztahy mezi jednotkami jsou následující:

- **1 kb/s = 1 000 b/s**
- **1 Mb/s = 1 000 kb/s = 1 000 000 b/s**
- **1 Gb/s = 1 000 Mb/s = 1 000 000 kb/s = 1 000 000 000 b/s**
- **1 Tb/s = 1 000 Gb/s = 1 000 000 Mb/s = 1 000 000 000 kb/s = 1 000 000 000 000 b/s**

M – mega, **G** – giga, **T** – tera

Šířka pásma přenosového média je ovlivněna jednak vlastnostmi tohoto média a jednak signalizační metodou zvolenou pro přenos.

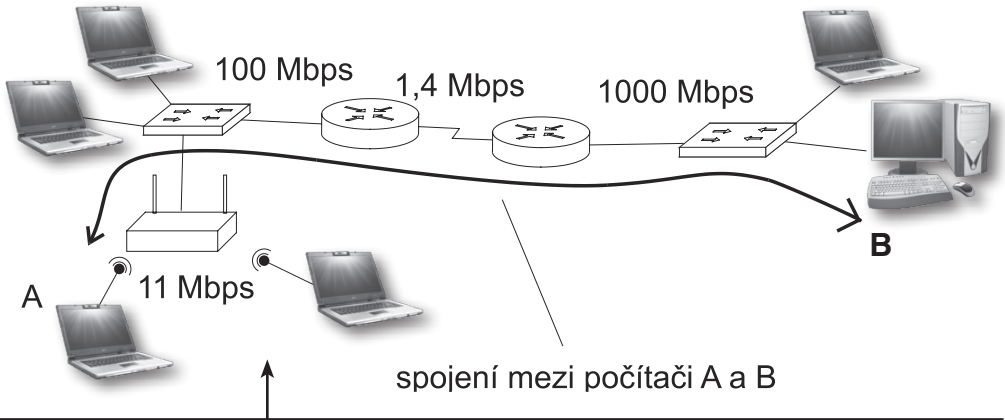
Throughput – výkon, propustnost

Kvůli mnoha faktorům, které negativně ovlivňují přenos dat síťovým médiem, nedosahuje skutečná rychlost přenosu dat teoretické hranice, kterou vyjadřuje **bandwidth** – šířka pásma.

Aktuální výkon sítě závisí na množství přepravovaných dat a síťových zařízení na síti a na typu provozu.

Například na lokální síti používající technologii **Ethernet** soupeří jednotlivá síťová zařízení o přístup na médium, tudíž jejich aktuální výkon je snižován tím víc, čím víc zařízení na síti pracuje.

Mezi zdrojovým a cílovým síťovým zařízením může být množství síťových segmentů, z nichž každý může nabízet jinou šířku pásma. Pak je evidentní, že rychlost přepravy dat od zdroje k cíli bude omezena nejpomalejším segmentem mezi nimi. Linka s výrazně nižším výkonem se označuje za úzké hrdlo.



Spojení mezi počítači A a B nebude schopno dosáhnout vyšší přenosové rychlosti než 1,4 Mbps. Úzké hrdlo tvoří sériová linka spojující směrovače.

Goodput – skutečný výkon

Pro uživatele je nejdůležitější skutečný výkon přenosových linek. Zajímá ho, jak dlouhá doba bude potřebná k přenosu určitého souboru.

Skutečný výkon sítě označovaný též jako goodput je množství dat přenesených za jednotku času, kde se jedná o původní data bez doprovodných přidaných informací, které je na cestě sítí doprovází. Doprovodná data jsou různé hlavičky a patičky přidané v procesu zapouzdřování v průběhu procházení od nejvyšší vrstvy k nejnižší. Slouží k adresování, navazování spojení a zajištění kontroly a bezchybnosti přenosu.

Jestliže linka spojující dva komunikující počítače má šířku pásma 10 Mbps (**bandwidth**), výkon linky (**throughput**) je nižší (například 6 Mbps) z důvodu přítomnosti dalších počítačů a rušení a skutečný výkon je kvůli přepravování dalších doprovodných informací ještě nižší, například 4 Mbps.

Bandwidth (šířka pásma) ≥ throughput (výkon, propustnost) ≥ goodput (skutečný výkon)

Média

Standardy fyzické vrstvy udávají mimo jiné také vlastnosti přenosových médií.

U metalických médií určují, jaký typ přenosového média lze použít pro jakou šířku pásma, jakými koncovkami musí být kabel ukončen, na jakou vzdálenost lze určité přenosové médium použít bez nějakého aktivního mezičlánku a jakým způsobem lze provádět testování kabelu.

UTP kabel

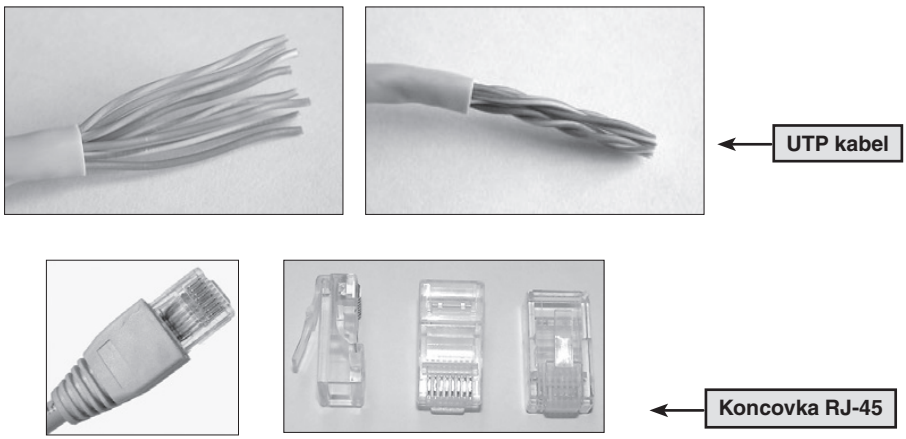
Určitý typ kabelu lze použít často u více technologií, například kabel UTP lze použít u **Ethernetu 10 Mbps**, **100 Mbps** i **1 000 Mbps**. Použití kabelu UTP závisí také na kvalitě tohoto kabelu. Kabel UTP byl probrán podrobně v druhé kapitole o přenosových médiích.

I různé typy konektorů lze použít u více přenosových technologií. Například koncovka **RJ-45** běžně používaná na LAN sítích v technologii **Ethernet** se může používat i na spojeních v rámci WAN sítí pomocí jiných typů přenosových médií.

Na metalických rozvodech dochází během přenosu signálu k nejrůznějším druhům rušení, a to z vnějšího prostředí vlivem elektromagnetického záření různého původu, síťovým provozem v jiných kabelech, vlivem elektromotorů, elektromagnetického pole v zářivkách, rádiovým vysíláním a podobně.

Na metalických rozvodech se proti tomuto rušení můžete chránit použitím kabelů obsahujících stínění. V případě kabelu UTP snižuje vliv elektromagnetického záření stočení jednotlivých párů vodičů v kabelu.

Stočení vodičů po párech v kabelu UTP chrání proti vnějšímu rušení – jelikož jsou vodiče fyzicky velmi blízko, vzniká na nich velmi podobný rušivý signál. Přijímač jej díky této podobnosti dokáže odhalit a odstranit.



Podobné elektromagnetické pole vzniká i uvnitř kabelu. Elektrický signál prochází kabelem jedním vodičem na jednu stranu a druhým na opačnou. Průchodem signálu vodičem vzniká v okolí elektromagnetické pole, které ve vedlejším vodiči indukuje elektrický signál, tzv. **přeslech**. Díky tomu, že provozem na vedlejším vodiči vzniká opačné elektromagnetické pole, dochází k jejich vzájemnému vyrušení a minimalizaci přeslechů.

K zamezení vzniku přeslechů mezi jednotlivými páry vodičů v kabelu slouží jejich různé stočení, každý pár má jiný počet stočení na metr.

Podle kvality kabelů je UTP rozdělen do kategorií. *Podrobně bylo popsáno v druhé kapitole o přenosových médiích.*

Kabely vyšší kategorie jsou schopny přenášet data vyšší rychlostí. V současné době se nepoužívají kabely kategorie nižší než **5e** a pro nové instalace se doporučuje používat kabely UTP kategorie **6**. Je vhodné investovat do kvalitnější kabeláže s ohledem na možný budoucí vývoj. Jestliže bude v budoucnu záměr používat nové technologie umožňující rychlejší přenos dat, musely by se nevhodné kabeláže celé vyměnit za kvalitnější, což by ve výsledku zvýšilo výdaje na síťovou infrastrukturu.

UTP kabel je zakončen koncovkou **RJ-45**. Pořadí vodičů v koncovce je důležité. Jak již bylo popsáno v druhé kapitole, jiné pořadí má koncovka typu **A** a jiné koncovka typu **B** (**T568A** a **T568B**).

Kombinací koncovky typu **A** a **B** vzniká křížený kabel, kombinací dvou koncovek stejného typu vzniká přímý kabel. Jejich použití je závislé na typu zařízení, která kabel spojuje. Zařízení stejného typu jsou spojena kříženým kabelem (spojení **počítač – počítač**, **směrovač – směrovač přes ethernetové porty**, **směrovač přes ethernetový port – počítač**, **přepínač – přepínač**), zatímco zařízení různého typu kabelem přímým (**počítač – přepínač**, **směrovač – přepínač**).

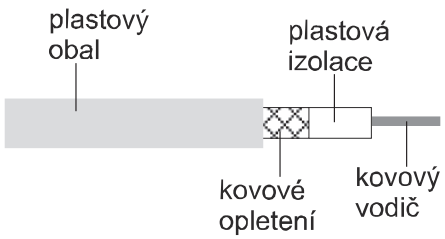
V případě použití nesprávného typu kabelu nehrozí vážné nebezpečí poškození, ale bude to mít vliv na funkčnost. Přenos nebude správný.

Některá zařízení umí rozpoznat typ připojeného kabelu, a pokud potřebují opačný typ, dokážou vnitřně přepnout provoz tak, aby síťová komunikace mohla probíhat.

Koaxiální kabel

Podrobnosti o koaxiálním kabelu již byly uvedeny v druhé kapitole o přenosových médiích. Jen připomeňme, že odrušení vnějšího elektromagnetického pole zajišťuje vnější kovové opletení, které musí být správně ukončeno, aby nefungovalo jako anténa a nežádoucí vlivy nezesilovalo.

V minulosti byl koaxiální kabel používán u síťových rozvodů v LAN sítích pro přenosovou rychlost 10 Mbps. Nahradilo ho používání kabelu UTP, který umožňuje snazší instalaci a z pohledu nezávislosti počítačů i topologii typu hvězda, kdy jsou jednotlivé počítače připojeny UTP kabely k přepínači nebo rozbočovači.



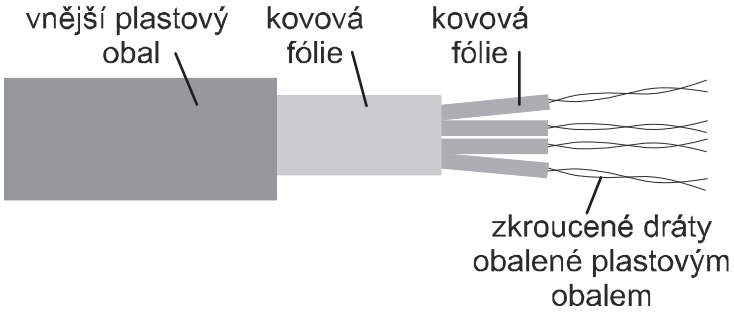
V současnosti je využíván pro přenos vysokofrekvenčního rádiového signálu, obzvláště pro přenos televizního signálu formou kabelové televize nebo signálu přijatého anténou do televizního přijímače. Také se používá pro spojení antény s aktivním Wi-Fi prvkem. Používané koaxiální kabely mají impedanci **50** nebo **75 Ω**, která musí souhlasit s požadavky připojeného zařízení.

STP kabel

Jak již bylo popsáno v druhé kapitole o přenosových médiích, dalším kovovým kabelem, který obsahuje stínění, je **kabel STP**.

Původně se využíval hlavně v LAN sítích s technologií **Token Ring**. V současné době nachází uplatnění v LAN sítích, kde se používá **vysokorychlostní Ethernet 10 Gbps**.

Z hlediska možnosti úrazu elektrickým proudem na metalických rozvodech nebo poničení jiných síťových zařízení je nutno dbát na správné uzemnění elektrických zařízení, správné zapojení kabelů, oddělení kabelů vedoucích elektrický proud a data a pravidelně kontrolovat rozvody kabelů.



Optická přenosová média

Pomocí optických médií lze dosahovat výrazně vyšší přenosové rychlosti než u metalických rozvodů. Jak již bylo popsáno dříve (viz 2. kapitola), data se vysílají na optické médium formou světelných pulzů, které se na druhé straně pomocí fotodiod převádějí zpět na elektrický signál.

Výhodou oproti metalickým médiím (kromě šířky pásma) je imunita proti elektromagnetickému rušení a skutečnost, že nevytvářejí žádné rušivé elektromagnetické pole. Nehrozí tu ani úraz nebo poškození jiných zařízení elektrickým proudem při špatném uzemnění. Optická vlákna dokážou díky svému nízkému útlumu vést signál na výrazně delší vzdálenosti, až na vzdálenost mnoha kilometrů. Pak je potřeba přidat **regenerátor signálu** – **repeater** (**opakovač**).

Nevýhodou ve srovnání s metalickými médii je vyšší cena. Na druhé straně ale optická vlákna umožňují vyšší přenosovou rychlost. Jistou nevýhodou může být i potřeba kvalifikovaného správce, který dokáže s těmito médii pracovat. Aby nedocházelo k poškozením kabelu, například nadměrným ohybům, což by vedlo ke ztrátám signálu, je nutné s optickými kabely zacházet s přiměřenou opatrností. S aktivním kabelem zacházejte velmi opatrně, protože díky síle laseru, který vytváří světelné pulzy, by mohlo snadno dojít k poškození zraku, pokud byste se podívali do koncovky aktivního kabelu.

Jak již bylo vysvětleno dříve, existují **dvě základní varianty optických vláken** – **jednovidové** a **mnohovidové**. **Jednovidové vlákno** používá pro přenos dat laser, a je proto dražší než mnohovidové, které může používat místo laseru infračervenou diodu. Jednovidové vlákno dokáže vést signál na výrazně větší vzdálenosti než mnohovidové, protože jednak používá laser a také zde nedochází ke světelnému rozkladu jako u mnohovidového vlákna. Tato vzdálenost je přibližně několik desítek kilometrů, zatímco u mnohovidového vlákna se vzdálenost pohybuje řádově od stovek metrů do přibližně dvou kilometrů. **Mnohovidové vlákno** je použitelné spíše v lokálních sítích na kratší vzdálenosti (stovky metrů), zatímco jednovidové lze použít pro spojení vzdálených sítí.

Bezdrátový přenos

Bezdrátový přenos používá pro přenos dat rádiové a mikrovlnné frekvence.

Dobře se uplatňuje v otevřeném prostoru. Některé materiály, budovy nebo členitost terénu mohou způsobovat rušení a zeslabování signálu. K rušení často dochází i vlivem jiných bezdrátových zařízení, vlivem elektromagnetického pole v okolí spotřebičů nebo vysíláním podobných frekvencí elektrospotřebiči.

Nevýhodou u bezdrátového přenosu je možnost odposlechu přenášené komunikace. Proto se do přenosu přidává prvek zabezpečení a šifrování dat.

Výhodou je mobilita počítačů a absence kabelů.

Na bezdrátovém přenosu se obvykle **nedosahuje příliš vysokých přenosových rychlostí.** Ve srovnání s metalickými nebo dokonce optickými rozvody, kde se data běžně přenášejí rychlostmi 1 Gbps nebo 10 Gbps, je přenos u bezdrátů obvykle řádově **do desítek Mbps**, **výjimečně stovek Mbps**.

Základní čtyři typy bezdrátových přenosů jsou popsány následujícími standardy.

- **802.11** – **bezdrátová síť** (*Wireless LAN, WLAN*) – často označovaná **Wi-Fi**. Používá se na lokálních sítích. Využívá metodu přístupu na sdílené médium označovanou jako **CSMA/CA**.
- **802.15** – **WPAN** (*Wireless Personal Area Network*) – často označovaná jako **Bluetooth**. Používá se ke spojení dvou zařízení na vzdálenost několika metrů (1–100 m).

- **802.16 – WiMAX** (***Worldwide Interoperability for Microwave Access***) – širokopásmový bezdrátový přístup k internetu. Používá se pro venkovní bezdrátové sítě jako doplněk k síti Wi-Fi, která je chápána jako standard pro vnitřní použití. Dosah je řádově okolo 50 km.
- **GSM** (***Global System for Mobile Communication***) – standard pro mobilní telefony, zajišťuje digitální přenos hovoru a sms. Umožňuje přenos dat v síti mobilních telefonů pomocí protokolu **GPRS**. Dosah je přibližně od několika stovek metrů až po desítky kilometrů, podle specifikace GSM až do 35 km. Dosah je závislý na výkonu antény, výšce jejího umístění a terénu, ve kterém se signál šíří.

Standardy WLAN

Během let se vyvinulo několik standardů pro sítě **WLAN**. Mají různé specifikace, které je potřeba vzít v úvahu kvůli vzájemné kompatibilitě a schopnosti zařízení spolupracovat. V následujícím seznamu jsou uvedeny nejznámější a rozšířené standardy.

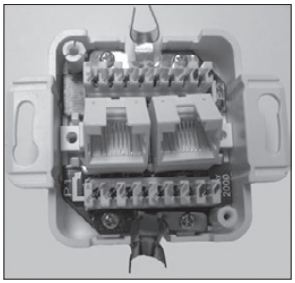
- **802.11a** – zařízení pracují na frekvenci 5 GHz a rychlosti přenosu až 54 Mbps. Kvůli vyšší frekvenci má menší pokrytí a hůře prochází skrz budovy. Zařízení pracující na tomto standardu nejsou schopna spolupracovat se zařízeními pracujícími na standardu 802.11b a 802.11g.
- **802.11b** – zařízení pracují na frekvenci 2,4 GHz a mají rychlost přenosu do 11 Mbps. Signál s touto frekvencí má vyšší dosah a lépe proniká budovami než u standardu 802.11a. Je představitelem technologie označované jako Wi-Fi.
- **802.11g** – zařízení pracují na frekvenci 2,4 GHz a mají rychlost přenosu až 54 Mbps. Zpětně kompatibilní s 802.11b.
- **802.11n** – nový standard, jehož záměrem je, aby zařízení pracovala na frekvenci 2,4 nebo 5 GHz a data byla přenášena rychlostí od 100 do 210 Mbps na vzdálenost přibližně 70 metrů.

Konektory

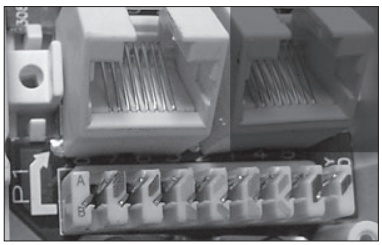
Standardy fyzické vrstvy definují, jak mají vypadat zakončení kabelů. Například **kabel UTP** využívaný v technologii **Ethernet** bývá často ukončen koncovkou **RJ-45**. Kabel lze také zapojit do zásuvky nebo tzv. **patch panelu**. Způsob instalace koncovky byl popsán dříve. Způsob zapojení zásuvky nebo připojení kabelu do patch panelu se také řídí standardy a pravidly. Zapojení je často usnadněno zobrazením barev, které odpovídají barvám vodičů v **UTP kabelu**.

Ukázka zásuvky pro konektory RJ-45

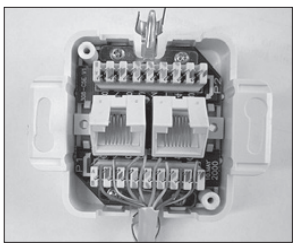
Dvě zásuvky propojené kabelem UTP je vhodné nainstalovat tak, aby obě byly typu **A** nebo typu **B**, tedy aby kabel, který je spojuje, byl přímý.



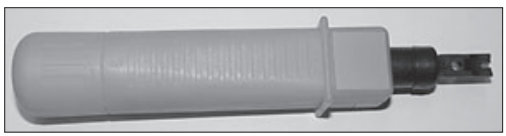
Zásuvka se dvěma zdířkami pro konektory RJ-45 V detailu je vidět barevné označení samořezných kontaktů, do kterých se zastrčí příslušně barevně označené vodiče kabelu UTP a pomocí zářezového narážecího nástroje se do kontaktů nainstalují.



V detailu je vidět označení dvou barevných řad jako A a B, které odpovídá standardům A a B u konektorů RJ-45.



Dráty kabelu UTP nainstalované v samořezných zdířkách zásuvky



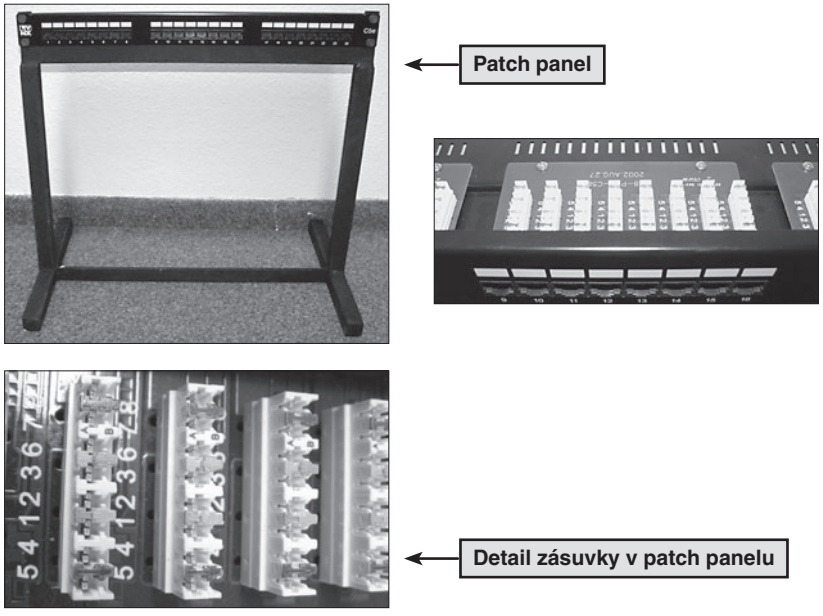
Zářezový narážecí nástroj

Ukázka patch panelu pro konektory RJ-45

Patch panel pro konektory **RJ-45** je panel obsahující několik portů pro zapojení koncovek **RJ-45**. Úsporným systémem je zde integrováno mnoho zásuvek do jednoho panelu, který lze připevnit například do rozvodné skříně spolu s přepínači.

Pak lze pomocí krátkých **UTP kabelů** zajistit přehledné propojení mezi zásuvkami v **patch panelu** a zásuvkami v přepínačích.

Nainstalování **UTP kabelu** do zásuvky v **patch panelu** probíhá obdobně jako u zásuvky na zeď. Pomocí narážecího nástroje se vodiče **UTP kabelu** zasunou do příslušně zbarvených samořezných zdířek v zásuvce **patch panelu**. Zdířky jsou navíc očíslovány, čísla odpovídají pořadí vodičů v koncovce **RJ-45** typu **A** nebo **B**.



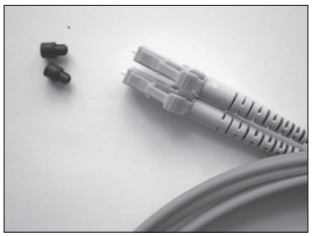
Optické konektory

K zakončení optických kabelů se používají speciální konektory. Jejich instalace není snadná, musí se dodržet několik požadavků, aby konektor správně fungoval. Napojení optických vláken musí být naprosto přesné, jednotlivé konce se musí vzájemně dotýkat, vlákno uložené do konektoru musí být správně vloženo až ke konci ve správném směru, konec nesmí být poškozen nebo znečištěn.

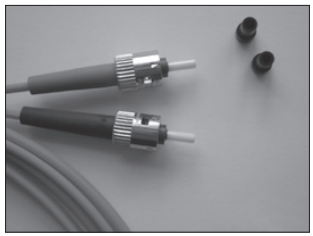
Každý optický kabel by měl projít kompletním ověřením a měřením. Kabely včetně instalovaných konektorů se běžně prodávají a jsou již otestované.

Časté typy konektorů

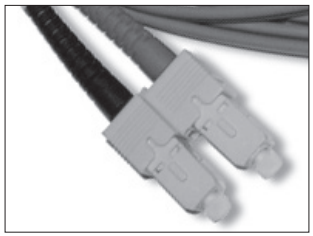
- **LC** (***Lucent Connector***) – drobný konektor, který lze použít k zakončení obou typů vláken.
- **ST** (***Straight-Tip***) – používá se obvykle k zakončení jednovidových optických vláken, má bajonetové uchycení.
- **SC** (***Subscriber Connector***) – obvykle je používán k zakončení mnohovidových vláken.



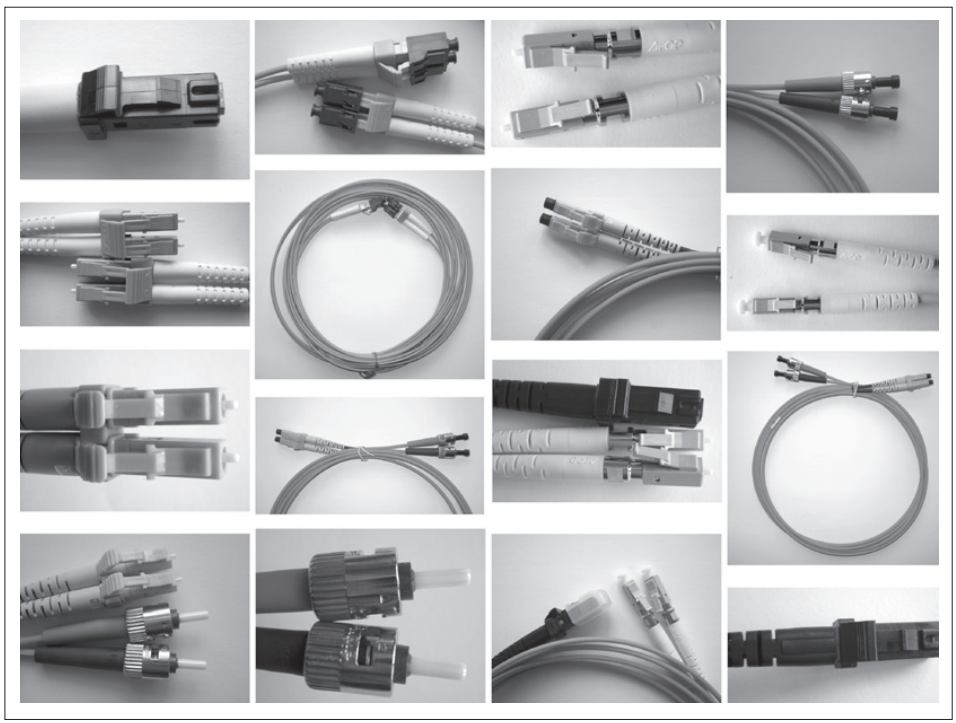
LC konektor



ST konektor



SC konektor



18. Ethernet

Vlastnosti

V současné době je nejrozšířenější technologií v sítích LAN právě **Ethernet**.

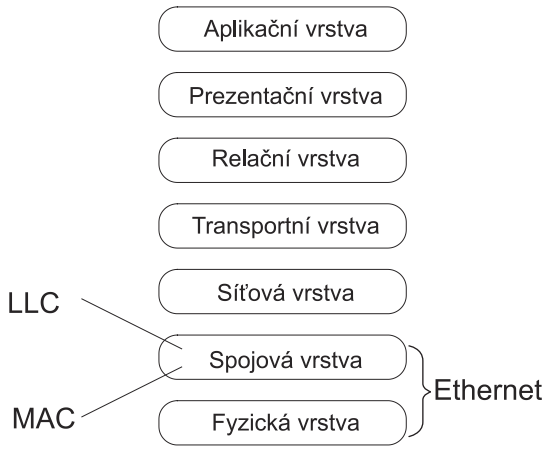
Jeho výhodami jsou snadné zavedení i údržba, schopnost přizpůsobovat se novým technologiím, spolehlivost a vcelku nízká cena při nových instalacích a obnově.

V dnešní době je stejná technologie schopna přenášet data naprosto odlišnými přenosovými rychlostmi, od megabitů za sekundu až po desítky gigabitů za sekundu. Schopnost přizpůsobit se i výrazně vyšším přenosovým rychlostem dělá z **Ethernetu** nejen technologii, kterou lze používat v lokálních sítích LAN, ale která může rozsahem pokrýt spojení v sítích MAN nebo i WAN.

Během svého vývoje prošel velkou proměnou od technologie umožňující komunikovat zařízením na sdíleném médiu nízkými přenosovými rychlostmi až po dnešní **vysokorychlostní full-duplexní Ethernet**.

První ethernetový standard byl zveřejněn v roce 1980 společnostmi ***Digital Equipment Corporation***, ***Intel*** a ***Xerox*** (**DIX**). Stal se otevřeným standardem, aby z něj mohl mít každý člověk užitek.

V roce 1985 vydala společnost ***IEEE*** **standard pro LAN sítě**. Standard pro **Ethernet** měl číslo **802.3** a byl vytvořen tak, aby byl v souladu s normami **ISO** a **OSI modelem**.



Z pohledu modelu **OSI** se **Ethernet** vyskytuje v dolních dvou vrstvách – **fyzické** a **spojové**. Ve spojové vrstvě zasahuje dolní **podvrstvu MAC**.

Ve fyzické vrstvě se **Ethernet** zabývá fyzickou stránkou přenosu, vysíláním signálu na přenosové médium, topologiemi a fyzickými součástmi tvořícími síť.

Podvrstva MAC ve spojové vrstvě se zabývá identifikací zařízení pomocí adresování fyzickými adresami, předává rámce ve formě bitů fyzické vrstvě k odvysílání, zajišťuje spojení mezi fyzickým zařízením a vyššími vrstvami OSI modelu.

Nad podvrstvou MAC je **vyšší podvrstva spojové vrstvy – LLC** (***Logical Link Control***). Ta zajišťuje komunikaci mezi horními vrstvami a dolní vrstvou, mezi softwarem a hardwarem.

Horní podvrstva LLC zajišťuje identifikaci protokolu ve vnořeném paketu, kontrolu chybovosti a řízení toku. Je hardwarově nezávislá.

LLC pracuje jako ovladač síťové karty, umožňuje síťové kartě komunikovat s vyššími vrstvami OSI modelu.

MAC je nižší podvrstva spojové vrstvy, její funkce je implementována v hardwaru, zejména v síťové kartě.

Zajišťuje adresování rámce pomocí fyzických adres (MAC adresy) a označení začátku a konce rámce. Řídí přístup k médiu a je hardwarově závislá.

V procesu zapouzdřování dat se ve spojové vrstvě v podvrstvě MAC označí rámec skupinou bitů, které označují jeho hranice, vloží se informace o cílové a zdrojové MAC adrese a přidá se kontrolní políčko umožňující zjistit, zda rámec přišel v pořádku do cíle.

Podvrstva MAC zajišťuje umístění dat na přenosové médium a v cílovém zařízení zajistí převzetí dat z média k dalšímu zpracování.

V ethernetových standardech je specifikováno, jakým způsobem mají být data ve fyzické vrstvě zakódována a jakým způsobem odvysílána.

Ethernet dokáže fungovat jak na metalických, tak na optických rozvodech. I přes různorodost přenosových médií a konektorů má základní struktura ethernetového rámce stále stejnou podobu.

Pro spojení vzdálených počítačů v LAN síti se používají mezičlánky – přepínače a rozbočovače. Rozbočovače jsou dnes nahrazovány přepínači, které umožňují lepší správu kapacity přenosového média a jsou i cenově srovnatelné.

Zapojení

Z hlediska logické topologie jsou počítače využívající technologii **Ethernet** na sdíleném médiu. Jedná se tedy o **sběrnicovou topologii**.

Znamená to, že všechny počítače na určitém segmentu sítě přijímají všechna data, která jsou v segmentu vysílána. Následně s daty naloží podle toho, zda jsou určena pro něj. Jestliže ano, pak data postupují vrstvami modelu směrem nahoru. Pokud ne, data jsou ignorována.

Pro přístup na médium používá **Ethernet** metodu **CSMA/CD**.

Historie

Vznik **Ethernetu** se datuje od 70. let 20. století, kdy na Havajských ostrovech vznikla rádiová síť spojující ostrovy a umožňující přenos informací. Síť nesla název **Alohanet**.

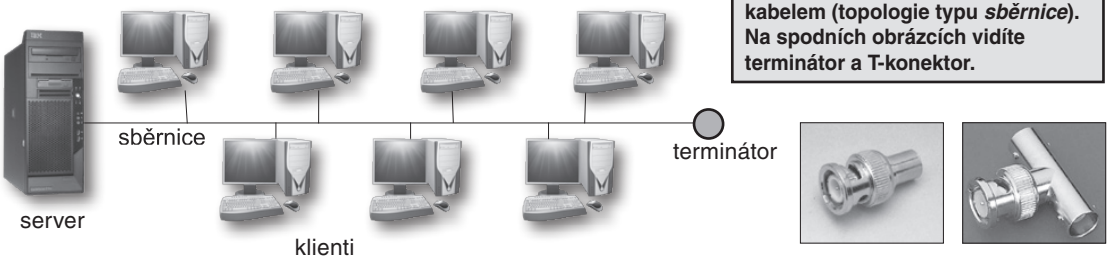
Všechny stanice dodržovaly stejný protokol, kdy se nepotvrzené vysílání po chvíli znovu opakovalo.

Podobná technika se později rozšířila i na metalických rozvodech a vyvinul se **Ethernet**.

Ethernet zajišťoval spojení počítačů na sdíleném médiu, počítače byly v logické **topologii sběrnice**.

Princip **CSMA/CD** vyřešil problém vysílání více počítačů na sdílené médium v jednu chvíli, problém vzniku kolize a zpracování takové kolize.

První verze **Ethernetu** připojovaly počítač pomocí **koaxiálního kabelu**. Všechny počítače byly připojeny na jeden kabel, a pokud se kterýkoliv z počítačů nechtěně odpojil, nastávaly problémy. Pak přestal přenos fungovat z důvodu špatného zakončení kabelu. Na konci sběrnice musel být **ukončovací článek** – **terminátor**, počítače byly k hlavnímu kabelu připojeny pomocí **rozbočovacích článků** – **T-konektorů**.



Propojení počítačů v síti koaxiálním kabelem (topologie typu *sběrnice*). Na spodních obrázcích vidíte terminátor a T-konektor.

Pomocí **tlustého koaxiálního kabelu** se mohla vytvořit síť o délce segmentu až 500 m, pomocí **tenkého** až 185 m.

První případ se označuje jako **10Base5**, druhý jako **10Base2**.

Práce s tenčím koaxiálním kabelem byla příjemnější, neboť byl ohebnější a lehčí.

Na obou typech byla maximální šířka pásma **10 Mbps**.

Jelikož počítače sdílely jedno přenosové médium, nastávaly problémy se vzájemnými kolizemi. S vyšším počtem počítačů na sdíleném segmentu dochází ke kolizím častěji a nakonec se síť stává přetíženou.

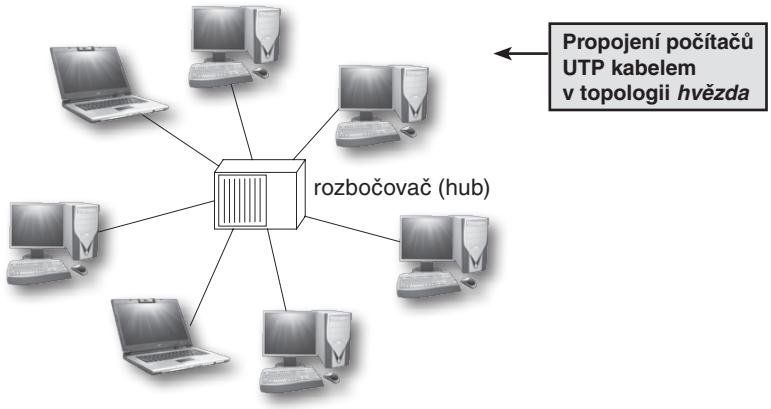
Koaxiální kabel byl později nahrazen **kabelem UTP**, který je lehčí, snáz se s ním manipuluje, je i levnější.

Navíc nový typ zapojení umožnil **topologii typu hvězda**, v níž odpojení jednoho počítače nezpůsobí kolaps celé sítě.

Všechny počítače se kabelem UTP zapojily do rozbočovače. Stále se jedná o sdílené médium, neboť rozbočovač kopíruje data přijatá na jednom portu na všechny ostatní.

Použití rozbočovače nevyřešilo problém s množstvím kolizí na síti.

Při použití rozbočovače vzniká fyzicky topologie typu hvězda, ale logicky je to stále topologie sběrnicová, protože počítače vysílají data na sdílené médium.



hvězda

Sítě, které měly jako centrální prvek **rozbočovač** (**hub**), k němuž byly **kabelem UTP** připojeny počítače, využívaly šířku pásma 10 Mbps a označovaly se jako **10Base-T**.

Na sdíleném médiu mohl v danou chvíli vysílat pouze jeden počítač, aby nedocházelo ke kolizím. Tento typ vysílání je typický pro **half-duplex**.

S rostoucím počtem počítačů v síti rostl významně počet kolizí, což výrazně snižovalo výkon počítačů a celé sítě.

Tento problém byl vyřešen nahrazením rozbočovačů **přepínači** (**switch**). Přepínač posílá data jen tím portem, na kterém je připojeno cílové zařízení, jemuž jsou data určena (s výjimkou broadcastu, který je adresován všem zařízením v kolizní doméně).

V době, kdy se začaly používat přepínače místo rozbočovačů, se také **výrazně zvýšila přenosová rychlost**. Šířka pásma vzrostla na 100 Mbps. V současnosti se stále využívají sítě s touto přenosovou rychlostí (**100 Mbps**).

Technologie, kdy se používají kabely UTP a šířka pásma je 100 Mbps, se označuje **100Base-TX**. Používají se přepínače jako centrální body, ke kterým se kabelem UTP připojují počítače. Přepínače zajišťují, že data putují pouze k cílovému zařízení, nezatěžují tímto provozem ostatní síťová zařízení a minimalizují výskyt kolizí na síti.

Vývoj přepínačů spolu s vývojem vysílání typu **full-duplex**, kdy v jednu chvíli probíhá vysílání i příjem signálu, vedl k rozvoji vysokorychlostního **Ethernetu 1000 Mbps** a více.

U gigabitového **Ethernetu** a vyšších se kromě UTP kabelu (**1 Gbps**) používají optické kabely.

Použití optických kabelů umožňuje přenos dat na větší vzdálenosti, a díky tomu dochází k potlačení hranic mezi LAN a WAN sítěmi.

Struktura ethernetového rámce

Dva základní typy ethernetového rámce jsou popsány jako rámec **Ethernet II** a rámec **IEEE 802.3**.

Jejich struktura je velmi podobná. **IEEE 802.3** je rámec, který byl vícekrát měněn, aby odpovídal nově vznikajícím technologiím.

Rámec **IEEE 802.3** obsahuje **SFD** (označení začátku rámce) a místo pole **Typ** má pole **Délka**.

Oba typy rámců definují **délku rámce minimálně 64 B** a **maximálně 1 518 B**. Do této délky se nezapočítává preambule a **SFD**. Je to délka rámce počítaná od cílové adresy po **FCS**.

Standard **802.3ac** z roku 1998 prodloužil maximální délku rámce na **1 522 B**, aby byla umožněna nová technologie **VLAN** (**virtuální sítě**).

Rámce, které mají délku mimo vymezené hranice, jsou zahozeny.

Ethernet II					
Preambule	Cílová adresa	Zdrojová adresa	Typ	Data	FCS
8 B	6 B	6 B	2B	46–1500 B	4 B

IEEE 802.3						
Preambule	SFD	Cílová adresa	Zdrojová adresa	Délka	Data	FCS
7 B	1 B	6 B	6 B	2 B	46–1500 B	4 B

Preambule slouží k **synchronizaci zařízení**.

SFD (***Start of Frame Delimiter***) – označuje začátek rámce.

Preambule a **SFD** slouží k **synchronizaci zdrojového a cílového síťového zařízení**. Připravuje přijímací zařízení na přijetí rámce.

FCS (***Frame Check Sequence***) je **kontrolní součet**. Po přijetí dat provede cílové zařízení vlastní výpočet, a pokud nesouhlasí s hodnotou **FCS**, pak je rámec zahozen. Rozpor ve výsledku ukazuje na nějaký problém přenosu, poškození nebo nedovolenou změnu rámce.

Délka udává **délku rámce**.

Typ označuje **typ vnořeného vyššího protokolu** v paketu.

Zdrojová a cílová adresa jsou 6B fyzické MAC adresy síťových zařízení. Zařízení, které rámec přijme, porovná cílovou adresu se svou vlastní MAC adresou, a pokud nesouhlasí, rámec zahodí. V opačném případě se rámcem začne zabývat a předá jej vyšší síťové vrstvě.

V poli **Délka** u rámce **IEEE 802.3** byla uvedena přesná délka rámce. Tato hodnota je spolu s políčkem **FCS** využita k tomu, aby bylo možné určit, zda byl rámec přijat v pořádku.

U rámce **Ethernet II** je v poli **Typ** uveden typ vnořeného protokolu.

Protože oba tyto významy se často využívaly, došlo po roce 1997 k jejich kombinaci v rámci standardu **802.3x**.

Pokud je v políčku **Délka** uvedeno číslo větší nebo rovno **1 536** (hexadecimálně **0x0600**), pak toto číslo určuje vnořený protokol a zapouzdřená data jsou dekódována v souladu s tímto protokolem.

Pokud je číslo menší nebo rovno **1 500** (hexadecimálně **0x05DC**), pak toto číslo určuje délku rámce.

V políčku **Data** se vyskytuje datová jednotka třetí vrstvy, obvykle je to **IP paket v4**. Délka dat musí být **alespoň 46 B**, kratší rámec by byl považován za poškozený v důsledku kolize. Kratší rámec se proto doplňuje výplní na potřebných 46 B dat. 46 B dat spolu s doprovodnými informacemi rámce dávají dohromady potřebných 64 B, což je minimální délka rámce.

MAC adresa

Je to jedinečný identifikátor, který používají různé protokoly spojové vrstvy.

MAC adresa je **fyzická adresa, kterou v sobě nese každá síťová karta nebo síťový adaptér**. Ethernetová MAC adresa je 48bitová, zapisovaná nejčastěji jako šest skupin dvouciferných hexadecimálních čísel oddělených dvojtečkou nebo pomlčkou nebo jako tři skupiny čtyřciferných hexadecimálních čísel oddělených tečkami.

00-21-85-E2-A3-F5

00:21:85:E2:A3:F5

0021.85E2.A3F5

MAC adresa byla vyvinuta kvůli jednoznačné identifikaci síťových zařízení na lokální síti používající pro přenos technologii **Ethernet**. Tato adresa je obsažena v ethernetovém rámci. Síťová zařízení dokáží podle MAC adresy rozhodnout, zda je rámec určen pro ně, či nikoliv.

MAC adresa má stabilní strukturu nezávislou na druhu **Ethernetu**, který ji využívá.

První polovina MAC adresy označuje výrobce, který ručí za to, že druhá část MAC adresy bude unikátní.

První část MAC adresy přiděluje výrobci **centrální správce IEEE**. Některým velkým výrobcům je přiděleno více identifikátorů **OUI** (***Organizationally Unique Identifier***).

Tato adresa je **obvykle vložena v ROM paměti síťové karty**, a pak **je neměnná**. V případě, že se adresa MAC změní, musí se dbát na to, aby nedošlo v lokální síti k duplikaci, protože pak by nastával problém v síťovém přenosu.

Výpis MAC adresy počítače lze provést příkazem **ipconfig /all**.

MAC adresa slouží k přenosu rámce na lokálním síťovém segmentu. Aby mohl být rámec přepraven do jiného segmentu, je potřeba provést směrování pomocí směrovače. Ten z vnořeného paketu získá informaci o cílové IP adrese a provede směrování na další segment sítě. Přechodem do jiného segmentu se mění zdrojová a cílová MAC adresa tak, aby vyjadřovala adresu těch zařízení v aktuálním lokálním segmentu, která si rámec předávají. IP adresa zůstává neměnná.

Ethernetový unicast, multicast a broadcast

Fyzická MAC adresa určitého konkrétního síťového zařízení je adresa typu **unicast**. Slouží k adresování rámce pro konkrétní zařízení. Příkladem může být žádost jednoho počítače o zobrazení webové stránky z jiného počítače. V jejich vzájemné komunikaci se budou objevovat jejich vlastní fyzické MAC adresy, komunikace bude probíhat mezi těmito dvěma počítači, bude to komunikace typu unicast.

Ethernetová broadcast adresa sestává v šestnáctkovém vyjádření ze samých **F**.

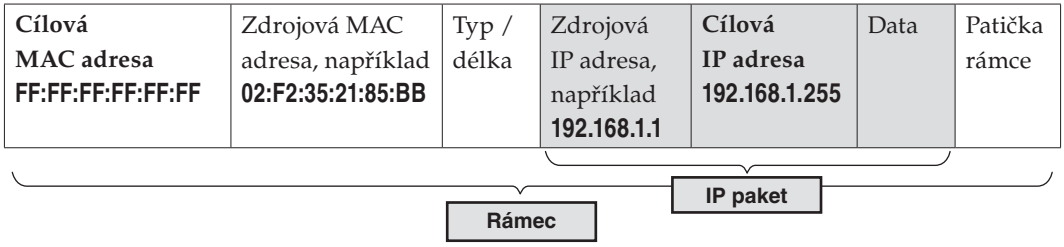
FF:FF:FF:FF:FF:FF

Jestliže síťové zařízení dostane rámec, který jako cílovou MAC adresu obsahuje adresu broadcastu, zpracuje ho.

Jak již bylo dříve řečeno, jestliže je potřeba vyslat paket všem počítačům na dané podsíti, obsahuje IP adresa cílového zařízení samé jedničky v části identifikující tato zařízení (řečeno binárně). Jedná se o broadcast na třetí vrstvě.

Například **v síti 192.168.1.0/24** je **adresa broadcastu** IP adresa **192.168.1.255**.

Protože je paket zapouzdřen do rámce, do kterého se přidávají další informace, mezi jiným i MAC adresa cílového zařízení, pak v případě broadcastu musí být v políčku cílové MAC adresy uvedena odpovídající adresa broadcastu na druhé vrstvě - **FF:FF:FF:FF:FF:FF**.



Multicast je vysílání směřované k určité skupině cílových zařízení. Jak už víte, existuje multicast na třetí vrstvě a jsou pro něj vyhrazeny IP adresy **224.0.0.0** až **239.255.255.255**.

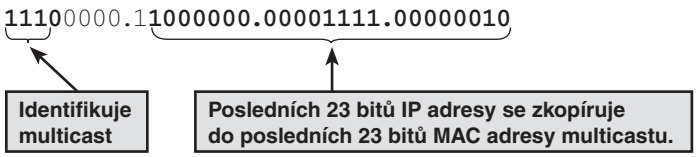
Těmto IP adresám musí nějakým způsobem odpovídat i multicastová adresa na druhé vrstvě.

Multicast na druhé vrstvě vždy začíná čísly **01-00-5E** – vyjádřeno hexadecimálně. Zbývajících 24 bitů v MAC adrese multicastu získáte z IP adresy multicastu tak, že posledních 23 bitů z IP adresy přenesete do posledních 23 bitů MAC adresy a 24. bit od konce v MAC adrese vyplníte nulou. Pak tento binární zápis převedete do šestnáctkové soustavy a MAC adresa multicastu je připravena.

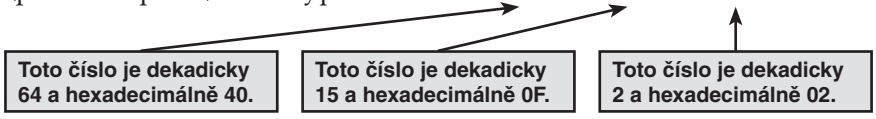
Například k multicastové adrese **224.192.15.2** je multicast na druhé vrstvě číslo **01-00-5E-40-0F-02**.

Postup

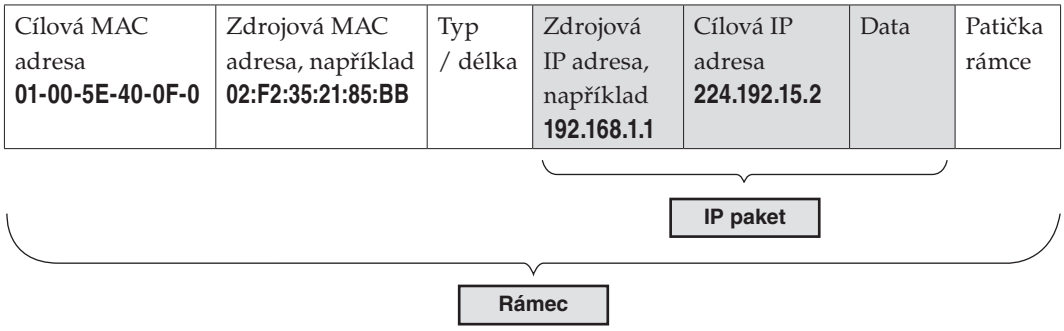
IP adresa vyjádřená binárně vypadá takto:



MAC adresa multicastu bude mít v posledních 23 bitech číslice **1000000.00001111.00000010**. Chybějící 24. bit (počítáno zprava) bude vyplněn nulou: **0**1000000.00001111.00000010



Multicast na druhé vrstvě k IP adrese **224.192.15.2** je tedy MAC adresa **01-00-5E-40-0F-02**.



Hexadecimální soustava

Čísla vyjádřená v hexadecimální soustavě se používají k zápisu MAC adres a IP adresy verze **6**.

Hexadecimální (šestnáctková) soustava používá pro zápis čísel šestnáct číslic: **0–9**, **A–F**.

Příkladem čísla zapsaného v šestnáctkové soustavě může být **FF**, **AB04**5, **21**, **C568A** a podobně.

Je dobré znát převody mezi šestnáctkovou (hexadecimální) a desítkovou (dekadickou) soustavou. K tomu musíte znát základní interpretaci jednotlivých číslic šestnáctkové soustavy.

Deka	Hexa	Deka	Hexa	Deka	Hexa	Deka	Hexa
0	0	4	4	8	8	12	C
1	1	5	5	9	9	13	D
2	2	6	6	10	A	14	E
3	3	7	7	11	B	15	F

Například číslo **FF** v šestnáctkové soustavě znamená $\mathbf{F \cdot 16^1 + F \cdot 16^0 = 15 \cdot 16 + 15 \cdot 1 = 255}$.

Princip zápisu je stejný u všech číselných soustav, jen umocňovaný základ se liší podle typu soustavy. U šestnáctkové soustavy je základem číslo 16, u desítkové je to číslo 10 atd.

Pro odlišení čísla zapsaného v hexadecimální soustavě se před číslo píše symbol **0x** (například **0xF0B**).

Převod čísla z hexadecimální soustavy do desítkové

Převeďte číslo **C568A** do desítkové soustavy.

C568A = C·16^4 + 5·16^3 + 6·16^2 + 8·16^1 + A·16^0 = 12·65536 + 5·4096 + 6·256 + 8·16 + 10·1 =

= 786 432 + 20 480 + 1 536 + 128 + 10 = 808 586

Číslo **C568A** v šestnáctkové soustavě je totéž jako číslo **808 586** v soustavě desítkové.

Převod čísla z desítkové soustavy do šestnáctkové

Princip převodu spočívá v tom, že se od čísla postupně odečítá maximální mocnina šestnáctky v takovém násobku, aby zbytek byl nula nebo číslo menší než odečítaná mocnina šestnáctky.

Převeďte číslo **61 480** z desítkové soustavy do šestnáctkové.

Mocniny šestnáctky jsou následující (podstatné jsou jen ty mocniny šestnáctky, které nepřevýší zadané číslo):

- **16^0 = 1**
- **16^1 = 16**
- **16^2 = 256**
- **16^3 = 4 096**
- **16^4 = 65 536** – to už je příliš velké číslo

Od zadaného čísla odečtěte číslo **4096**, a to celkem **15krát**.

61 480 – 15·4 096 = 40

Do zbytku **40** se nižší mocnina šestnáctky – číslo **256** – nevejde, resp. vejde se **0krát**.

Do zbytku **40** se vejde číslo **16**, a to celkem **2krát**.

40 – 2·16 = 8

Do zbytku **8** se vejde číslo **1** (jakožto nejnižší mocnina šestnáctky) celkem **8krát**.

Pak už žádný zbytek nezůstane.

Celkem tedy můžete číslo **61 480** rozložit následovně:

61 480 = 15·4 096 + 0·256 + 2·16 + 8·1

Pro výsledný převod do šestnáctkové soustavy je výhodné mít číslo rozložené tak, aby součet zleva začínal nejvyšší mocninou šestnáctky a pokračoval směrem doprava nižšími mocninami – tak jak je to vidět na výše provedeném rozkladu.

Pro názornost můžete přepsat čísla **4 096**, **256**… do mocnin šestnáctky, která je tu základem soustavy.

$61\,480 = 15 \cdot 16^3 + 0 \cdot 16^2 + 2 \cdot 16^1 + 8 \cdot 16^0 = F \cdot 16^3 + 0 \cdot 16^2 + 2 \cdot 16^1 + 8 \cdot 16^0$

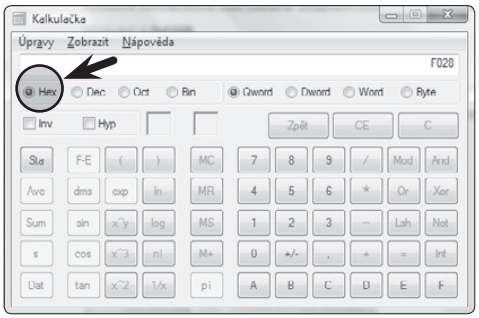
Výsledné šestnáctkové číslo získáte soupisem číslic vyskytujících se u mocnin šestnáctky – **F028**.

Výsledek je **0xF028**.

Využití kalkulačky pro převody mezi číselnými soustavami

Pokud je to možné, pak je rychlejší a obvykle méně chybové použít pro převody kalkulačku. Nicméně znalost principu převodu mezi soustavami a schopnost jej provést patří k základnímu vzdělání středoškoláka.

Zadané číslo v desítkové soustavě převedete do šestnáctkové (resp. dvojkové, osmičkové) pouhým přepnutím příslušného přepínače.



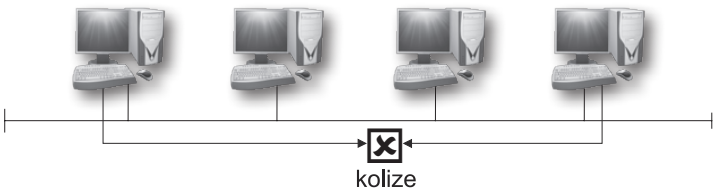
Přístup na médium u Ethernetu

Síťová zařízení používají **metodu CSMA/CD** (***Carrier Sense Multiple Access/Collision Detection***) ke zjištění, zda na sdíleném médiu zrovna neprobíhá vysílání.

Pomocí této metody síťová zařízení zjistí, zda během vysílání nedošlo ke kolizi, a pokud ano, pak data vyšlou znovu.

Ve chvíli, kdy počítače chtějí vysílat, poslouchají, zda na médiu zrovna nevysílá někdo jiný, a pokud ano, počkají náhodnou chvíli a pak médium otestují znovu. Jestliže ani pak žádný signál na médiu nedetekují, začnou vysílat.

Pokud začnou dva počítače vysílat ve stejnou chvíli nebo krátce po sobě a jsou od sebe hodně vzdáleny, pak kvůli zpoždění na síti o sobě zpočátku nevědí,



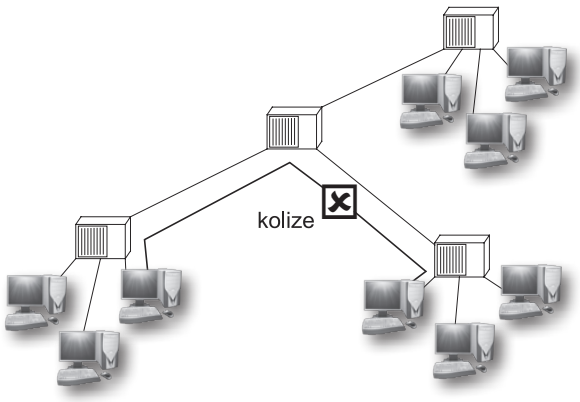
a tak dojde ke kolizi. Signály se smísí a dojde k jejich zničení. V tomto stavu pak dále putují sítí.

Počítače, které zrovna poslouchají, zda na síti neprobíhá provoz, tuto kolizi zaregistrují. Je typická zvýšením amplitudy signálu. Původní počítače, které kolizi způsobily, vyšlou navíc další signál, tzv. **jam signál**, kterým informují všechny ostatní počítače, že došlo ke kolizi. Jam signál je dlouhý 32 bitů a umocňuje kolizní signál za účelem informovat všechny ostatní počítače na síti. Obvykle obsahuje sadu opakujících se jedniček a nul. Kolizní fragmenty jsou obvykle kratší než 64 bytů, což je menší velikost, než je minimální povolená velikost rámce.

Všechny počítače, které mají v úmyslu vysílat, spustí tzv. **backoff algoritmus**. Pomocí něj si vyberou náhodnou dobu z předem definovaného intervalu, po kterou budou čekat, než znovu začnou poslouchat sdílené médium, a v případě nulového provozu začnou vysílat. Tím, že si vybírají náhodně, se zvyšuje šance, že až příště začnou vysílat, neshodnou se a nedojde k další kolizi. Pokud přesto při následujícím vysílání ke kolizi dojde, spustí se opět tento **backoff algoritmus**, ale nyní si počítače budou již vybírat náhodnou prodlevu před vysíláním z dvojnásobného intervalu než původně. A tak to jde dál – dojde-li k další kolizi, interval se opět dvojnásobí. Počítač se pokusí odeslat rámec maximálně šestnáctkrát. Jestliže se mu to nepovede, oznámí to síťové vrstvě a další pokusy neprovádí. Výskyt problémů tohoto typu signalizuje přetíženost sítě.

Rozbočovač a kolizní doména

V případě, že se ke spojení počítačů použije **rozbočovač** (**hub**) nebo dokonce několik rozbočovačů a na každém je připojeno několik počítačů, sdílejí všechny počítače jedno médium, jsou v jedné kolizní doméně. Použitím rozbočovačů se zvětšují vzdálenosti jednotlivých počítačů a snadno se může stát, že více počítačů začne vysílat v přibližně stejnou chvíli, kdy na síti detekují klid. Protože na síti dochází kvůli vzdálenosti a síťovým mezičlánkům ke zpožděním, dojde ke kolizi.

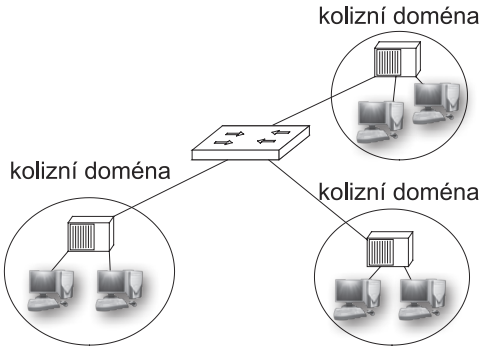


V síti, kde je hlavním centrálním mezičlánkem rozbočovač, který nijak nefiltruje provoz, může snadno dojít k zahlcení sítě. Stačí, aby jeden počítač měl nějaké problémy a začal do sítě šířit nekontrolovaně data. Všechny ostatní počítače v síti budou tato data přijímat, budou se jimi muset zabývat a nebudou moci samy vysílat.

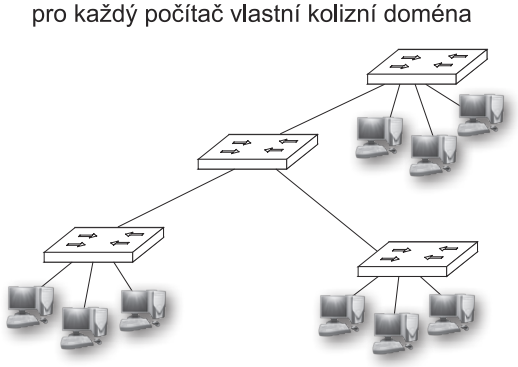
Problémy s rozšířením kolizní domény pomohly vyřešit přepínače (switche). Díky nim mohou probíhat souběžná vysílání mezi různými páry počítačů a ke kolizím nedochází téměř vůbec.

Přepínač jako centrální prvek

Přepínač rozděluje síť na jednotlivé kolizní domény. Dříve sloužil jako centrální prvek, ke kterému byly připojeny rozbočovače a na ty byly připojeny jednotlivé počítače. Na každém rozbočovači vznikla kolizní doména, která už ale svým provozem neovlivňovala jiné kolizní domény připojené na jiných portech přepínače.



Později, se snížením ceny přepínačů, byly rozbočovače z velké části vyměněny za přepínače. Na každém portu tak vzniká jedna kolizní doména. Počítače se pak již většinou vyhnou kolizím na síti.



Připojením počítačů k přepínači získávají počítače přístup k celé šířce pásma, kterou mohou používat pro přenos mezi sebou a přepínačem.

Téměř nedochází ke kolizím ve vysílání, což zvyšuje výkon sítě a datovou propustnost.

Přepínač podporuje full-duplexní provoz, proto lze data současně přijímat i vysílat. Tím se zdvojuje šířka pásma. Je-li například šířka pásma 100 Mbps, je tato šířka pásma přístupná jak pro přijímání, tak pro vysílání signálu.

Počítač vyšle rámec přepínači, ten se rozhodne, na který port jej přepne, a pak data tímto portem pošle k cílovému počítači. Tak mezi zdrojovým a cílovým počítačem vzniká momentální virtuální okruh, v rámci kterého mohou využívat plnou šířku pásma.

Pokud není protistrana připravena, přepínač pozdrží data ve své paměti a pošle je ve vhodnou chvíli. Tato metoda se nazývá anglicky ***store and forward***, v překladu „pozdržet a poslat". Při této metodě posílání dat přepínač přijme celý rámec a zkontroluje jeho kontrolní součet FCS. Pokud je v pořádku, přepne ho na patřičný port, jinak ho zahodí.

Přepínač si během provozu zjistí, na kterém portu má připojeny které počítače. Tyto údaje si vede v tabulce MAC adres. Zkoumá rámce, zjišťuje, jaká je zdrojová MAC adresa, a tuto adresu si spojí s příchozím portem. Dvojici MAC adresa + port si zapíše do své tabulky. Někdy se tato tabulka nazývá **přepínací tabulka**.

Na začátku svého provozu je přepínací tabulka prázdná, a pak se přepínač chová jako rozbočovač – posílá data všemi porty s výjimkou příchozího. Stejně se chová i během provozu, kdy má tabulku již vybudovanou, ale dostane rámec pro počítač, jehož MAC adresu v tabulce nenajde. Pošle data všemi porty mimo příchozí port, neboť v příchozím segmentu již všechny počítače rámec dostaly.

Během provozu si přepínací tabulku vytváří, prochází procesem učení, a pak začne plnit svou roli přepínače.

Jakmile je přepínací tabulka vytvořena, může začít cílené přepínání. Přepínač přepne rámec pouze na port, který vede k cílovému počítači.

Údaje v přepínací tabulce jsou časově označeny. Po uplynutí určité doby si přepínač vymění údaj v tabulce za aktuální.

Někdy dochází k filtrování provozu. Mohou být nastavena určitá bezpečnostní opatření, která brání přepínání rámců z určitých adres na určité segmenty. Pak jsou takové rámce zahozeny stejně jako rámec, který je poškozený. To přepínač zjistí z políčka FCS z rámce.

Časování na Ethernetu

Zpoždění

Kvůli vzdálenostem počítačů dochází během šíření signálu po médiu ke **zpoždění**. Přispívají k němu také síťové mezičlánky, které data přeposílají.

Zpoždění na síti má za následek kolize signálu, protože naslouchající počítače žádný provoz na síti nedetekují, přestože tam je, jen o něm kvůli zpoždění nevědí. Začnou proto vysílat, a tak dojde ke kolizi.

Synchronizace

U **Ethernetu** s přenosovou rychlostí 10 Mbps (a méně) se prvních 64 bitů preambule a SFD použije k **synchronizaci přijímacího obvodu**. Po přijetí těchto prvních 8 bytů jsou tato synchronizační data zahozena. Ethernet 10 Mbps je označován za asynchronní, protože potřebuje synchronizační bity, aby se přijímací okruh synchronizoval s vysílačem.

Ethernet verze 100 Mbps a vyšší je synchronní, a tak synchronizační bity na začátku rámce nepotřebuje, přesto je těchto prvních 8 bytů kvůli zpětné kompatibilitě vysíláno.

Bit time

Za **bit time** se označuje doba potřebná k odvysílání jednoho bitu. Liší se podle přenosové rychlosti.

U **Ethernetu** s přenosovou rychlostí 10 Mbps je bit time **100 ns** (**ns** – nanosekunda). U **Ethernetu** s přenosovou rychlostí 100 Mbps je to **10 ns** atd.

Signál se kabelem UTP šíří **rychlostí** přibližně **20 cm za 1 ns**. Kabelem o délce 100 m proběhne signál za přibližně 500 ns.

U **Ethernetu** s přenosovou rychlostí 10 Mbps se na kabel o délce 100 m vejde najednou 5 odvysílaných bitů, u **Ethernetu** s přenosovou rychlostí 100 Mbps přibližně 50 bitů. U **Ethernetu** s rychlostí 1 000 Mbps je to asi 500 bitů.

Minimální délka ethernetového rámce je 64 bytů, tj. 512 bitů.

Slot time

Slot time je doba potřebná k tomu, aby obě vysílací zařízení mohla detekovat vzniklou kolizi na nejvzdálenější části sítě, tedy doba potřebná k cestování signálu na nejvzdálenější místo sítě a zpět. Slot time je vyjádřen jako násobek parametru bit time.

Je to také doba, po kterou vysílací stanice čeká, než přistoupí k opětovnému odvysílání rámce, který byl zničen v důsledku kolize.

Slot time je důležitý parametr na síti používající **Ethernet** do přenosové rychlosti 1 000 Mbps v režimu half-duplex.

Přenosová rychlost	Bit time	Slot time	Doba odpovídající parametru slot time
10 Mbps	100 ns	512 x bit time	51,2 µs
100 Mbps	10 ns	512 x bit time	5,12 µs
1 Gbps	1 ns	4 096 x bit time	4,096 µs
10 Gbps	0,1 ns	nepoužívá se	nepoužívá se

Od parametru slot time se odvíjí minimální povolená délka rámce. Vychází to z požadavku, aby se vysílací stanice dověděla o kolizi ještě před dokončením vysílání rámce a mohla je znovu odvysílat.

Rámec kratší než minimální povolená délka je považován za kolizní fragment a je zahozen.

U rámce na **Ethernetu** s přenosovou rychlostí do 100 Mbps je tato délka 64 bytů. Aby mohl i **Ethernet** pracující na rychlosti 1 000 Mbps fungovat v half-duplexním režimu, doplňuje se minimální délka rámce výplní na délku 512 bytů. Pak je rámec dostatečně dlouhý na to, aby se vysílací stanice stihla dovědět o kolizi ještě před ukončením vysílání. Tato výplň se pak před zpracováním přijímací stanicí odstraní.

Mezera mezi rámci

Mezera mezi dvěma nekolidujícími rámci je důležitá proto, aby se přenosové médium mohlo stabilizovat a aby měl přijímač čas přijatý rámec zpracovat. U všech typů **Ethernetu** od 10 Mbps do 10 Gbps je tato mezera definována jako 96násobek hodnoty bit time.

Přenosová rychlost	Mezera mezi rámci	Doba trvání mezery
10 Mbps	96 x bit time	9,6 µs
100 Mbps	96 x bit time	0,96 µs
1 Gbps	96 x bit time	0,096 µs
10 Gbps	96 x bit time	0,0096 µs

Rámec 1	Mezera mezi rámci	Rámec 2

Mezera je měřena od posledního bitu FCS prvního rámce po první bit preambule dalšího rámce.

Druhy Ethernetu

Pomocí standardu **802.3** jsou definovány čtyři varianty **Ethernetu**, které využívají jako přenosové médium optické nebo metalické kabely.

- **10 Mbps** – označuje se jako **Ethernet 10Base-T**
- **100 Mbps** – označuje se jako **rychlý Ethernet**
- **1 Gbps** – označuje se jako **gigabitový Ethernet**
- **10 Gbps** – označuje se jako **desetigigabitový Ethernet**

Hodnoty vyjadřují maximální teoretickou šířku pásma. K realizování přenosu se používají různá přenosová média a konektory.

V následující tabulce jsou uvedeny nejrozšířenější varianty **Ethernetu** i s jejich přenosovými médii, maximálními doporučenými vzdálenostmi pro vysílání, šířkou pásma a typem vysílání.

Šířka pásma	Označení	Typ vysílání	Přenosové médium	Maximální vzdálenost
10 Mbps	10Base-5	Half-duplex	Tlustý koaxiální kabel	500 m
10 Mbps	10Base-2	Half-duplex	Tenký koaxiální kabel	185 m
10 Mbps	10Base-T	Half-duplex	UTP kabel od kategorie 3	100 m
100 Mbps	100Base-T	Half-duplex	UTP kabel od kategorie 5	100 m
200 Mbps	100Base-TX	Full-duplex	UTP kabel od kategorie 5	100 m
100 Mbps	100Base-FX	Half-duplex	Mnohovidové optické vlákno	400 m
200 Mbps	100Base-FX	Full-duplex	Mnohovidové optické vlákno	2 km
1 Gbps	1000Base-T	Full-duplex	UTP kabel kategorie 5e	100 m
1 Gbps	1000Base-TX	Full-duplex	UTP kabel kategorie 6	100 m
1 Gbps	1000Base-SX	Full-duplex	Mnohovidové optické vlákno	550 m
1 Gbps	1000Base-LX	Full-duplex	Jednovidové optické vlákno	5 km
10 Gbps	10GBase-CX4	Full-duplex	Twinaxial – kabel podobný koaxiálnímu kabelu, obsahuje místo jednoho vnitřního vodiče dva	15 m
10 Gbps	10GBase-T	Full-duplex	UTP kabel kategorie 6a nebo 7	100 m
10 Gbps	10GBase-SX4	Full-duplex	Mnohovidové optické vlákno	300 m
10 Gbps	10GBase-LX4	Full-duplex	Jednovidové optické vlákno	10 km

10 Mbps Ethernet

Původní verze **desetimegabitového Ethernetu** používaly pro přenos **tenký** nebo **tlustý koaxiální kabel** a označovaly se **10Base-2** a **10Base-5**. Dnes se tato původní varianta již nepoužívá, byla nahrazena Ethernetem se stejnou šířkou pásma, ale využívajícím pro přenos kabel UTP. Původně se používala **kategorie 3** tohoto kabelu, později byla nahrazena **kategorií 5**.

K signalizaci se používá **Manchesterská metoda**.

UTP kabel umožňoval připojení počítačů do centrálního prvku – **rozbočovače** (**hub**) – a tím se vytvořila **topologie typu hvězda**.

Délka jednoho segmentu kabelu UTP je maximálně 100 m, pak je potřeba signál zregenerovat a zesílit. K tomuto účelu se používal **rozbočovač** nebo **opakovač** (**hub** a **repeater**), později byly rozbočovače nahrazeny **přepínači** (**switch**).

K ukončení kabelu se používá osmipinová koncovka označovaná jako **RJ-45**.

Jedním párem vodičů se signál vysílá a jiným párem se signál přijímá. Vysílá se vodičem číslo **1** a **2**, přijímá se vodiči číslo **3** a **6**.

Na obrázku je zobrazena koncovka typu **A**. Vodič číslo **1** je bílo-zelený, vodič číslo **2** je zelený – těmito vodiči se vysílá.

Přijímá se vodiči č. **3** a **6**, zde jsou to vodiče bílo-oranžový a oranžový.

Ostatní vodiče jsou nevyužité.

V dnešní době se stále můžete setkat se sítěmi tohoto typu, ale jedná se již spíše o historickou záležitost.

Rychlý Ethernet (100 Mbps)

Tento typ (někdy zvaný **rychlý Ethernet**) nahradil svého desetimegabitového předchůdce. Šířka pásma je zde **100 Mbps**.

Přenosovými médii jsou nejčastěji **kabel UTP** a **optické vlákno**.

Při vysílání přes kabel UTP se využívají stejné páry vodičů jako u **Ethernetu 10 Mbps**. Kabel UTP musí být **kategorie 5** a vyšší.

Do centra se místo rozbočovače umísťují **přepínače**, které řeší problém s kolizemi na síti.

Před odvysíláním se data kódují metodou **4B/5B**.

Jestliže se jako přenosové médium použije optický kabel, pak jedním optickým vláknem probíhá vysílání a druhým se data přijímají. Pro zakončení mnohovidového optického

vlákna se obvykle používají **konektory SC**. Spojení optickým vláknem je spojení typu **point-to-point**. Spojují se dva počítače, dva přepínače, nebo počítač a přepínač.

U metalického média se používá signalizace pomocí elektrického proudu, u optického média pomocí světelných pulzů.

Gigabitový Ethernet (1 Gbps)

U tohoto typu **Ethernetu** se data zdržují na přenosové lince velmi krátkou dobu, proto se klade velký důraz na přesné časování, synchronizaci a přesnou interpretaci signálu na vzdáleném konci média. Data jsou náchylnější k rušení, proto jejich kódování probíhá komplexněji.

K přenosu se podle standardu používá **kabel UTP** a **optické vlákno**.

V případě přenosu pomocí kabelu UTP se již používají všechny čtyři páry a provoz je full-duplexní. Kabel musí být **kategorie 5** a více.

K přenosu se data kódují pomocí kódování **4D-PAM5** (***Four-Dimensional Pulse Amplitude Modulation 5***). Jeden byte se rozdělí po dvou bitech, které se ve formě symbolů pošlou paralelně po čtyřech vodičích. Na konci dojde k dekódování a zpětnému složení. K signalizaci se používá pětivrstevná amplitudová modulace.

Díky full-duplexnímu provozu se data současně vysílají i přijímají.

Během klidového stavu se na médiu vyskytuje 9 různých hladin napětí, během provozu až 17. Signál pak vypadá spíš jako analogová křivka než jako digitální signál. Proto je také signál náchylnější k rušení z důvodu problémů s kabelem a koncovkami.

V případě přenosu pomocí optického vlákna se lze zbavit problémů s rušením, a také délka segmentu, po kterém je třeba signál zregenerovat, je vyšší.

Data se kódují pomocí metody **8B/10B**. Osmibitové úseky dat se kódují do desetibitových symbolů. Účelem je, aby po sobě nešlo více než pět stejných bitů (jedniček nebo nul) a aby počet jedniček a nul byl vyvážený (na dvaceti bitech rozdíl mezi počtem jedniček a nul maximálně 2).

Můžete zvolit přenos jednovidovým nebo mnohovidovým vláknem.

V případě jednovidového vlákna dochází k přenosu pomocí laseru o vlnové délce 1310 nm.

U mnohovidového vlákna lze data přenášet infračerveným světlem o vlnové délce 850 nebo 1310 nm, světelným zdrojem bývá příslušná LED dioda.

10 Gbps Ethernet

Jedná se o přenos především **pomocí optického vlákna** s šířkou pásma **10 Gbps**.

Rámec má formát slučitelný se staršími verzemi, proto může být tato technologie použita u již existujících sítí pro zvýšení přenosové rychlosti se zachováním stávajícího zařízení.

Bit time je **0,1 ns**.

Je povoleno **pouze full-duplexní vysílání**. Není potřeba detekovat signál na síti a řešit kolize pomocí metody **CSMA/CD**.

Tuto technologii vysokorychlostního přenosu lze použít nejen na LAN sítích, ale může sloužit i pro vzdálená spojení typu MAN nebo WAN.

Do budoucna se pracuje na standardech umožňujících přenosové rychlosti ještě vyšší, a to **40–160 Gbps**.

19. ARP

ARP (***Address Resolution Protocol***) umožňuje zjistit MAC adresu síťového zařízení na stejném lokálním segmentu z jeho IP adresy.

Použije se v případě, že zdrojový počítač má v úmyslu poslat rámec sousednímu počítači a má k dispozici jeho IP adresu, ale zatím k ní nezná příslušnou MAC adresu. Pro adresování v rámci je potřeba zadat i cílovou MAC adresu.

Počítač vyšle broadcast na druhé vrstvě na adresu **FF:FF:FF:FF:FF:FF**. V rámci je uvedena hledaná IP adresa. Jako zdrojovou adresu uvede svou vlastní MAC adresu a IP adresu. Tento broadcast dostanou všechny počítače na lokálním segmentu. IP adresu a MAC adresu odesílatele použijí k vytvoření záznamu do vlastní ARP tabulky.

Počítač s hledanou IP adresou odpoví na tento broadcast odesílateli. Ten tak zjistí, jakou má počítač s danou IP adresou MAC adresu. Zapíše si ji do své ARP tabulky a použije ji k adresování rámce, který chce hledanému počítači odeslat.

V ARP tabulce se během chodu počítače postupně zaznamenávají páry IP adres a MAC adres zařízení na lokálním segmentu.

Při sestavování rámce se zkoumá vlastní ARP tabulka, zda obsahuje k příslušné IP adrese MAC adresu. Pokud ano, použije se pro sestavení rámce a data se odešlou. Jinak počítač odešle ARP dotaz, aby získal informace o MAC adrese příslušné dané IP adrese.

Údaje v ARP tabulce se udrží určitou dobu, která závisí na operačním systému. Obvykle je to několik minut. Pokud se adresa opakovaně používá, existence záznamu v tabulce se prodlouží.

Po spuštění více počítačů najednou se může chvilkově snížit výkon sítě, protože v ní probíhá najednou více broadcastů za účelem zjistit MAC adresy k IP adresám.

Proxy ARP

Pokud není cílový počítač na stejném lokálním segmentu, spáruje si zdrojový počítač cílovou IP adresu s MAC adresou výchozí brány – přilehlého rozhraní směrovače, který zprostředkovává spojení se vzdálenou sítí, v níž je cílový počítač.

Pro zjištění MAC adresy rozhraní směrovače se použije ARP dotaz. Směrovač odpoví MAC adresou svého rozhraní. Tento pár si pak počítač vede ve své ARP tabulce a použije jej pro sestavení rámce.

Komunikace ARP funguje pouze na lokálním segmentu a nedostává se za hranici, kterou vymezuje směrovač. Pokud je vhodné, aby spolu komunikovalo více segmentů, můžete na směrovači zapnout Proxy ARP, pomocí něhož směrovač zprostředkovává ARP komunikaci mezi jednotlivými segmenty sítě.

Směrovač pak vykonává roli prostředníka mezi počítači na dvou segmentech sítě.

Zdrojový počítač pak má spárovanou IP adresu počítače na vzdáleném segmentu s MAC adresou rozhraní směrovače, který mu komunikaci do této vzdálené sítě zprostředkovává.

K jedné MAC adrese tak může být přiřazeno více IP adres.

Příkaz ARP

ARP tabulku lze vypsat nebo smazat.

Výpis se provede příkazem **arp –a**.

```
C:\Users\Iva>arp -a

Rozhraní: 192.168.0.101 --- 0xa
  internetová adresa     fyzická adresa         typ
  192.168.0.1            00-22-b0-fc-99-3a      dynamická
  192.168.0.255          ff-ff-ff-ff-ff-ff      statická
  224.0.0.22             01-00-5e-00-00-16      statická
  224.0.0.252            01-00-5e-00-00-fc      statická
  239.255.255.250        01-00-5e-7f-ff-fa      statická
  255.255.255.255        ff-ff-ff-ff-ff-ff      statická
```

Výpis ARP tabulky

Přidání statického záznamu do tabulky se provede příkazem **arp -s IP MAC**. Takový záznam v tabulce je trvalý. Použití příkazu **arp –s** může být svázáno s oprávněními správce.

```
C:\Windows\system32>arp -s 81.95.96.94 00-22-b0-fc-99-3a

C:\Windows\system32>arp -a

Rozhraní: 192.168.0.101 --- 0xa
  internetová adresa     fyzická adresa         typ
  81.95.96.94            00-22-b0-fc-99-3a      statická
  192.168.0.1            00-22-b0-fc-99-3a      dynamická
  192.168.0.255          ff-ff-ff-ff-ff-ff      statická
  224.0.0.22             01-00-5e-00-00-16      statická
  224.0.0.252            01-00-5e-00-00-fc      statická
  239.255.255.250        01-00-5e-7f-ff-fa      statická
  255.255.255.255        ff-ff-ff-ff-ff-ff      statická
```

Přidání statického záznamu do ARP tabulky

Výpis ARP tabulky

Smazat záznamy v ARP tabulce lze příkazem **arp –d *** v případě smazání všech záznamů nebo příkazem **arp –d IPadresa** v případě smazání konkrétního záznamu.

```
C:\Windows\system32>arp -d *

C:\Windows\system32>arp -a
Žádné položky tabulky ARP nebyly nalezeny.
```

Smazání všech záznamů ARP tabulky

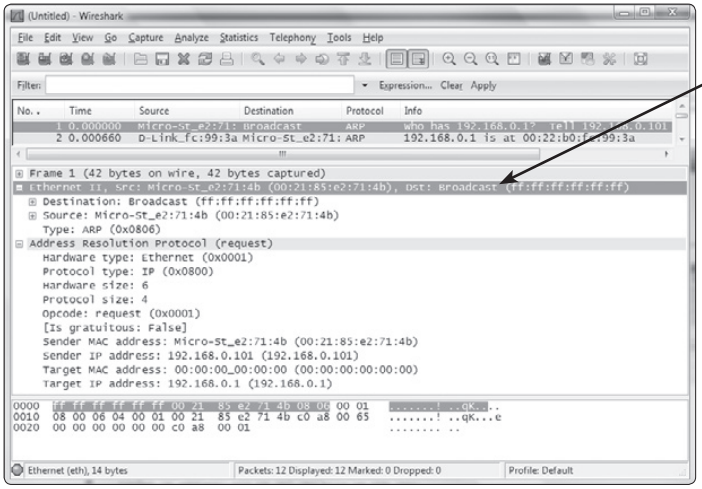
```
C:\Windows\system32>arp -a

Rozhraní: 192.168.0.101 --- 0xa
  internetová adresa     fyzická adresa         typ
  192.168.0.1            00-22-b0-fc-99-3a      dynamická

C:\Windows\system32>arp -d 192.168.0.1

C:\Windows\system32>arp -a
Žádné položky tabulky ARP nebyly nalezeny.
```

Smazání konkrétního záznamu ARP tabulky

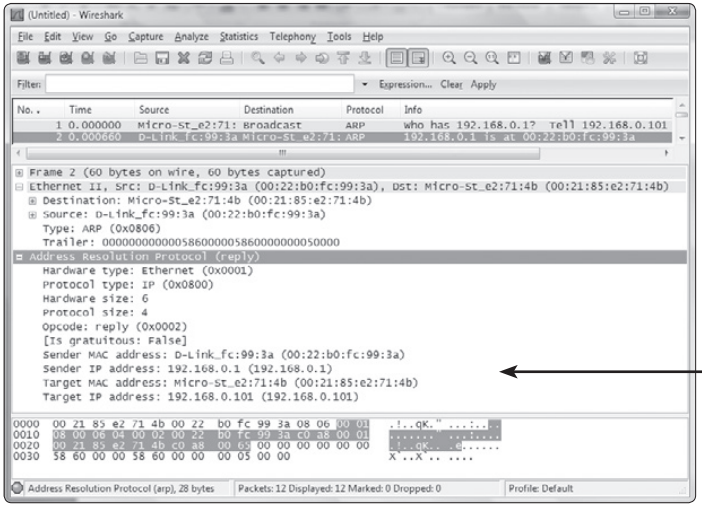
Zachycení ARP komunikace programem Wireshark

Na začátku je ARP tabulka vyprázdněna příkazem **arp –d ***.

Poté je proveden **ping** na vzdálenou adresu (IP adresu nebo její doménové jméno). Protože počítač bude potřebovat komunikovat mimo svou lokální síť, bude se dotazovat na MAC adresu své výchozí brány. Pošle ARP dotaz broadcastem na svou lokální síť a brána (směrovač) mu odpoví MAC adresou svého rozhraní.



ARP dotaz – směřuje broadcastem ke všem lokálním zařízením.

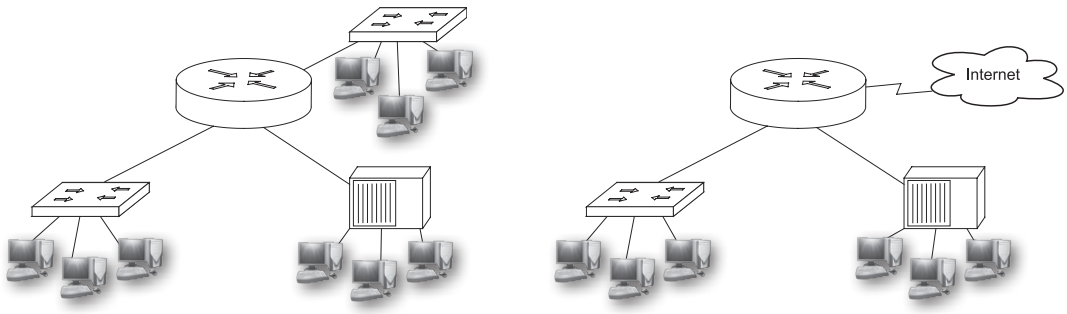


ARP odpověď – směřuje od konkrétního zařízení k dotazujícímu se počítači.
Počítač získává MAC adresu výchozí brány.

20. Zapojení LAN sítě

Zapojení

Každá lokální síť, která potřebuje zajistit přístup do jiné lokální sítě nebo do sítě WAN, má na své hranici obvykle router (směrovač), který zajistí směrování mezi sítěmi.



Směrovač má různé typy rozhraní, některá slouží ke spojení do vnitřní sítě, jiná pro sériové spojení s jiným směrovačem.

Lokální síť může být ke směrovači připojena pomocí UTP kabelu s koncovkou RJ-45. Směrovač může obsahovat porty pro připojení optického kabelu, kterým k němu může být připojena lokální síť, a porty pro připojení sériového kabelu, jenž slouží k propojení s jiným směrovačem.

V rámci lokální sítě jsou počítače obvykle připojeny k přepínačům (dříve rozbočovačům), které jsou centrálním prvkem síťové komunikace. I v současných sítích se mohou vyskytovat rozbočovače, které nerozdělují síť na jednotlivé kolizní domény, ale všechny počítače připojené na rozbočovač jsou členy jedné kolizní domény.

Počítače uvnitř lokální sítě oddělené od ostatních sítí směrovačem tvoří jednu broadcast doménu. Směrovač (pokud není nastaven jinak) nešíří broadcast vysílání a udržuje provoz uvnitř lokální sítě.

Přepínače rozdělují síť na jednotlivé kolizní domény – na každém portu je jedna kolizní doména.

Pro připojení počítačů k přepínačům nebo rozbočovačům se obvykle používá UTP kabel. Dnešní standard z pohledu přenosové rychlosti je **minimálně 100 Mbps**, **často 1 000 Mbps**.

Nejčastější standard pro síťový provoz na LAN sítích je **Ethernet** s metodou vysílání **CSMA/CD**.

Metalická nebo optická síť může být **doplněna o připojení počítačů pomocí bezdrátového připojení**. Počítače obsahující příslušnou bezdrátovou síťovou kartu se pak připojují **k přístupovému bodu** (**access pointu**), který jim zprostředkuje přístup do další sítě. Protože bezdrátový síťový provoz je náchylnější k rušení a bezpečnostním problémům, aplikují se zde zabezpečující prvky, jako je **šifrování dat** a **autorizace uživatele**. Pro přenos se používá metoda **CSMA/CA**.

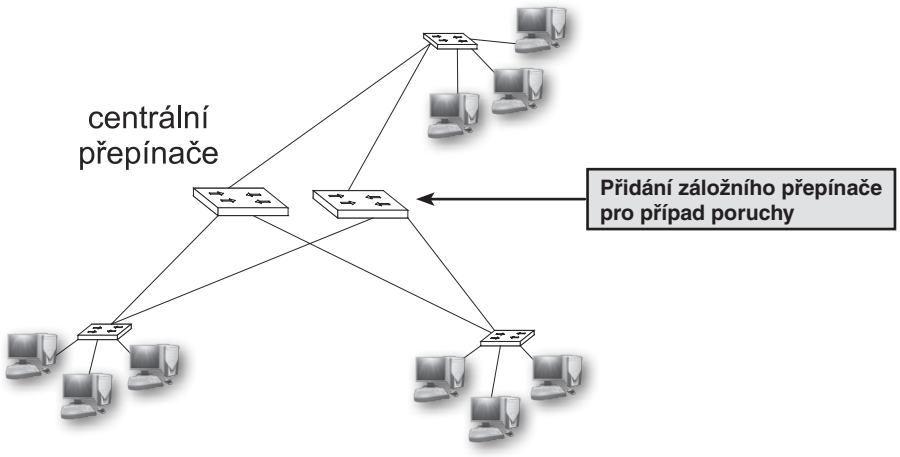
Bezdrátové spojení obvykle nedosahuje z hlediska přenosové rychlosti výkonu metalické nebo optické sítě, má ale jiné výhody, jako například mobilitu účastníků.

Při zapojení lokální sítě s využitím přepínačů je obvyklé připojit počítače **k několika menším přepínačům** a ty pak připojit na **centrální přepínač**.

Přidání nadbytečných záložních zařízení

Aby nedocházelo k problémům při výpadku centrálního přepínače, je vhodné předejít takové situaci přidáním dalšího centrálního přepínače, který bude fungovat paralelně s prvním a v případě výpadku prvního přepínače zajistí provoz.

Můžete také přidat nadbytečné linky pro propojení přepínačů, aby v případě problému s portem mohl být provoz přenášen druhým portem. Aby toto bylo možné provést, musí být přepínač schopen druhou linku v případě provozu první linkou vypnout, aby zde nedocházelo k vytvoření okruhu, problému s přepínáním provozu a zahlcení sítě. Vytvoření stromové struktury a vypnutí nadbytečných linek zajišťuje **STP** (***Spanning Tree Protocol***).



Volba přiměřeného vybavení

Při výběru přepínače zvažte, jaké počítače s jakou přenosovou rychlostí se budou k přepínači připojovat, jaký typ kabeláže bude použit a jaký je předpokládaný vývoj použitého síťového zařízení.

Přepínač může obsahovat všechny porty pro přenos metalickou kabeláží, do kterých se zapojuje koncovka RJ-45, nebo může obsahovat také optické porty. Porty pro UTP kabel mohou být různých rychlostí. Některé typy přepínačů umožňují přidávání dalších modulů s porty, jiné mají trvalou neměnnou konfiguraci.

Toto vše spolu s cenou za přepínač je nutno zvážit.

Stejně uvažujte i při výběru routeru (směrovače). Výše ceny se odvíjí od počtu a typu portů, modularity a služeb, které bude směrovač schopen poskytovat (bezpečnost, zajištění kvality přenosu, schopnost přenášet různé protokoly třetí vrstvy, překlady adres, DHCP apod.).

Počítače se k přepínačům nebo rozbočovačům mohou připojovat kabelem UTP, který by neměl přesáhnout maximální délku 100 m.

Pak dojde k regeneraci signálu na přepínači nebo rozbočovači a signál může ve své cestě pokračovat dále.

Při použití rozbočovačů se zvětšuje kolizní doména a při přílišné rozlehlosti by docházelo k problémům s přenosem a kolizemi. **Nedoporučuje se včleňovat mezi jakékoliv dva počítače více než čtyři rozbočovače.**

Při použití přepínačů odpadá problém s kolizemi, které jsou časté u rozlehlé kolizní domény v případě rozbočovačů.

Běžně se kabely UTP skrývají v lištách a jsou ukončeny zásuvkami. K nim se pak krátkými kabely připojují jednotlivé počítače. Na druhé straně se lišty s kabely sbíhají do centrálního místa, často rozvaděče, kde pak dojde k propojení krátkým kabelem mezi patch panelem a přepínačem. Stále platí, že součet délky kabelů od počítače do přepínače nemá přesáhnout 100 m. Nastával by problém s útlumem signálu a špatným rozpoznáním na druhé straně, čímž by docházelo ke zhoršení přenosu.

U spojení pomocí UTP kabelů je potřeba zvolit správný typ kabelu – přímý, nebo křížený. Některá zařízení umí provést vnitřně přepnutí podle potřeby.

Pro spojení vzdálenějších segmentů sítě můžete použít optickou kabeláž, která umožňuje připojení na větší vzdálenosti při zachování vysoké přenosové rychlosti.

Při volbě přenosových médií je dobré zvážit, na jakou vzdálenost je třeba data přenášet, jaké jsou podmínky pro přenos, zda nebude problémem zvýšené elektromagnetické a rádiové rušení, jaká je potřebná přenosová rychlost, jaká je cena za tato média, zda je k dispozici schopný administrátor znalý instalace daného typu médií a konektorů atd.

Pokud na síti existují společné zdroje, ke kterým je potřeba se často z mnoha míst dostávat, je vhodné umístit tyto zdroje na vysokorychlostní linku, aby nedocházelo k zahlcení linky. Může se jednat o server, který zprostředkovává služby nebo ukládání dat, může to být připojení mezi různými lokálními segmenty firmy. Připojení sítě k internetu obvykle nedosahuje rychlosti linek na lokální síti, přesto berte při volbě linky v úvahu, jaké vytížení bude přenos z internetu a na internet způsobovat, a podle toho s ohledem na cenu připojení zvolte šířku pásma.

Při vytváření nové sítě byste měli zvolit vhodné adresování síťových zařízení, podle potřeb rozdělit síť na podsítě a zajistit jejich provoz. Do podsítí je vhodné rozdělit počítače například podle typu provozu, který budou vytvářet, podle zdrojů, jež budou využívat, a podle zabezpečení jednotlivých segmentů.

Síťová zařízení, která potřebují IP adresu, jsou obvykle uživatelské počítače, servery, administrátorské počítače, síťové tiskárny, IP telefony, rozhraní na směrovači připojené do sítě, IP adresa přepínačů a access pointů používaná pro jejich vzdálenou správu atd.

Rozhraní směrovače, který zprostředkovává komunikaci počítačům v daném segmentu, se obvykle přiřazuje nejnižší IP adresa daného segmentu. Zařízením se pak přiřazují zbývající adresy rozsahu. Je dobré se držet určitého systému, aby později nedošlo k problémům s adresováním zařízení, jako je například zdvojení adres a podobně.

Testování spojení

Jak již bylo podrobně popsáno dříve, existuje celá řada možností jak ověřit funkčnost sítě. Patří mezi ně příkazy **ping**, **tracert**, **ipconfig**, **arp**, **netstat** a další. Připomeňme je již jen v krátkosti.

Ping

Příkazem **ping** lze otestovat správnou konfiguraci TCP/IP protokolu na počítači nebo spojení mezi dvěma různými počítači.

Příkazem **ping 127.0.0.1** se provede **ping** na vlastní interní virtuální rozhraní počítače. V případě neúspěchu je pravděpodobně problém s konfigurací TCP/IP protokolu na počítači.

```
C:\Windows\system32>ping 127.0.0.1

Příkaz PING na 127.0.0.1 - 32 bajtů dat:
Odpověď od 127.0.0.1: bajty=32 čas < 1ms TTL=128
Odpověď od 127.0.0.1: bajty=32 čas < 1ms TTL=128
Odpověď od 127.0.0.1: bajty=32 čas < 1ms TTL=128
Odpověď od 127.0.0.1: bajty=32 čas < 1ms TTL=128

Statistika ping pro 127.0.0.1:
Pakety: Odeslané = 4, Přijaté = 4, Ztracené = 0 (ztráta 0%),
Přibližná doba do přijetí odezvy v milisekundách:
    Minimum = 0ms, Maximum = 0ms, Průměr = 0ms
```

Testování interního rozhraní pomocí příkazu *ping*

Příkazem **ping IP-adresa** se provede **ping** na počítač se zadanou IP adresou.

```
C:\Windows\system32>ping 192.168.0.1

Příkaz PING na 192.168.0.1 - 32 bajtů dat:
Odpověď od 192.168.0.1: bajty=32 čas=1ms TTL=64
Odpověď od 192.168.0.1: bajty=32 čas=1ms TTL=64
Odpověď od 192.168.0.1: bajty=32 čas=1ms TTL=64
Odpověď od 192.168.0.1: bajty=32 čas=1ms TTL=64

Statistika ping pro 192.168.0.1:
Pakety: Odeslané = 4, Přijaté = 4, Ztracené = 0 (ztráta 0%),
Přibližná doba do přijetí odezvy v milisekundách:
    Minimum = 1ms, Maximum = 1ms, Průměr = 1ms
```

Testování spojení s jiným počítačem

Tracert

Příkazem **tracert** lze získat výpis uzlů od zdrojového počítače k cílovému.

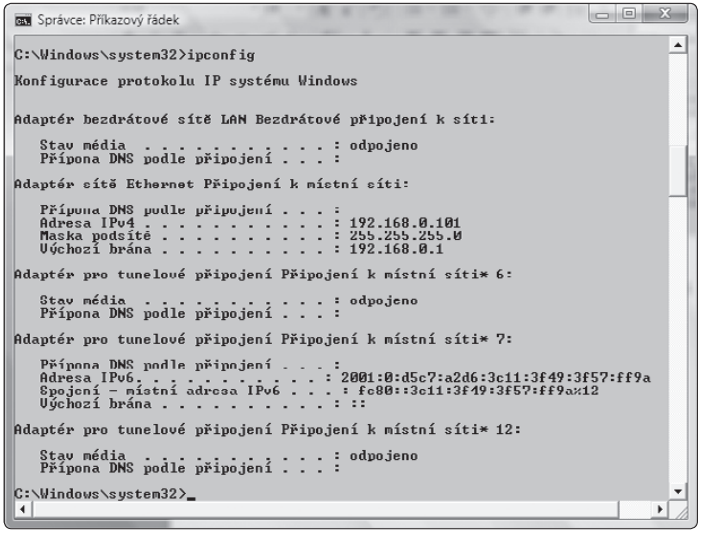
Tracert IP-adresa

```
C:\Windows\system32>tracert 81.95.96.94

Výpis trasy k uvirt7.active24.cz [81.95.96.94]
s nejvýše 30 směrováními:

  1     1 ms     1 ms    < 1 ms  info.mladotova.cz [212.27.205.129]
  2     1 ms     1 ms     1 ms  v10.bytovedruzstvoinstart.cpe.vol.cz [212.27
.57]
  3     3 ms     3 ms     3 ms  v50.b1.opa.prg.vol.cz [195.250.133.1]
  4     3 ms     5 ms     3 ms  v302.c1.opa.prg.vol.cz [195.250.141.153]
  5     4 ms     4 ms     3 ms  v124.bb3.prg2.vol.cz [212.20.124.42]
  6     4 ms     4 ms     3 ms  ge3-1.tr1.prg2.vol.cz [212.20.124.62]
  7     6 ms     4 ms     4 ms  nix4-ge.active24.cz [194.50.100.236]
  8     5 ms     4 ms     4 ms  uvirt7.active24.cz [81.95.96.94]

Trasování bylo dokončeno.
```

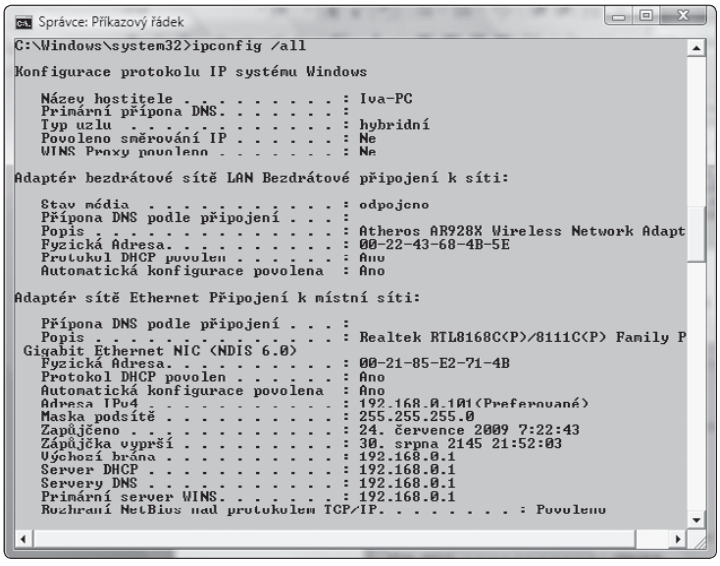
Testování spojení s jiným počítačem pomocí příkazu *tracert*

Ipconfig

Nastavení IP adresy, masky, výchozí brány, MAC adresy atd. lze získat příkazem **ipconfig**. Pro komplexnější výpis lze doplnit parametrem **/all (ipconfig /all).**



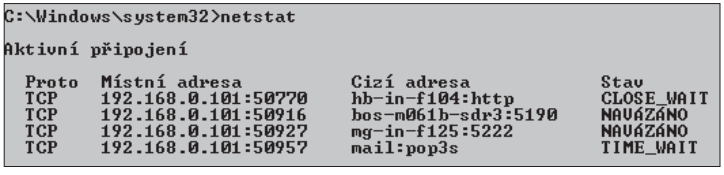
Výpis síťových nastavení příkazem *ipconfig*



Příkaz *Ipconfig /all*. Jsou vidět podrobnosti konfigurace.

Netstat

Tento příkaz slouží k výpisu navázaných spojení.



Výpis spojení příkazem *netstat*