

Obsah

8 Internet	1
8.1 Připojení k internetu	1
8.2 Možnosti internetu	1
8.2.1 WWW	1
8.2.2 Email	1
8.2.3 VoIP	1
8.2.4 Přenos souborů	1
8.3 IP adresa	1
8.4 Mac adresa	2
8.5 Podsítě	2
8.6 ARP	2
8.7 Směrování (routing)	3
8.7.1 Směrovací tabulky (routing table)	3
8.7.2 Směrovací algoritmy	3

8 Internet

- systém propojených počítačových sítí
- komunikace pomocí TCP/IP

8.1 Připojení k internetu

- zprostředkovatel – ISP (internet service provider)
- telefonní linka
 - analogový signál
 - konvertování přes modem
 - zastaralé
- kabel – Ethernet, optické kabely
- bezdrátově – satelity (starlink), mobilní síť (4G – 60 Mb/s; 5G – 600 Mb/s), WiFi
- elektrické rozvody

8.2 Možnosti internetu

8.2.1 WWW

- *world wide web*
- informační systém, informace přenášeny přes HTTP
- dostupné přes webový prohlížeč
- webové stránky uloženy na webových serverech
- přístup pomocí Uniform Resource Locators (URLs)
- HTML, JS, CSS, PHP...

8.2.2 Email

- přenos emailových zpráv
- příjemce, odesílatel, předmět, samotná zpráva, přílohy
- přenos pomocí POP3 (příjem) a SMTP (odesílání) nebo IMAP (přístup k serveru)

8.2.3 VoIP

- Voice over IP
- hlasová komunikace pomocí IP
- možno zobecnit na telefonní utilities přes internet (hlas, zprávy, dnes i video...)

8.2.4 Přenos souborů

- přenos souborů přes internet
- FTP, SMB, NFS

8.3 IP adresa

- adresa označující umístění na síti
- statická nebo přidělaná DHCP serverem
- správa protokolem IP
- IPv4
 - jednodušší verze
 - 32 bitové adresy – 0.0.0.0 – 255.255.255.255
 - komplexní, možné errorry
 - dnes již došli na veřejném internetu IP adresy
- IPv6
 - rozšíření IP protokolu
 - 128 bitů na adresy – 2001:db8:1234::f350:2256:f3dd
 - efektivnější a bezpečnější než IPv4
 - dnes se již plně nepřešlo, hlavně na lokálních sítích
- některé IP vytříbeny pro lokální sítě
 - class A – 10.0.0.0 – 10.255.255.255
 - class B – 172.16.0.0 – 172.31.255.255
 - class C – 192.168.0.0 – 192.168.255.255

8.4 Mac adresa

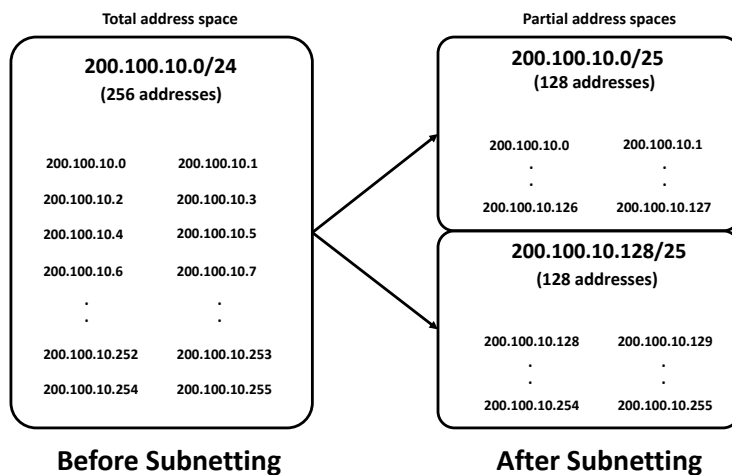
- unikátní číslo NICu (network interface controller)
- 6 dvojčíferných hexadecimálních čísel
- indentifikace zařízení na lokální síti
- dříve přiděleno výrobcem; dnes výrobci přidělena první polovina, druhou určuje sám
- dnes lze softwarem měnit
- protokol IEEE 802

8.5 Podsítě

- označení pro samostatnou část sítě uvnitř větší sítě
- určení IP adresy pomocí masky sítě
 - určuje, kolik bitů je konstantních a které jsou možné využít pro adresování
 - značená počtem bitů za adresou (např. /24), nebo jako binární číslo (255.255.255.0)
- využito pro separování zařízení na samostatné sítě
- poslední IP rezervována pro subnet broadcast pro všem uživatelům
- příklad
 - Prefix sítě 192.168.88.0/24 – host 192.168.88.1–192.168.88.254
 - * broadcast na 192.168.88.255
 - Prefix sítě 10.5.24.128/25 – host 10.5.24.129–10.5.24.254

	Binární zápis	Tečka-decimální zápis
IP adresa	11000000.00000000.00000010.10000010	192.0.2.130
Maska podsítě	11111111.11111111.11111111.11000000	255.255.255.192
Prefix sítě	11000000.00000000.00000010.10000000	192.0.2.128
Číslo hosta	00000000.00000000.00000000.00000010	0.0.0.2

Tab. 8.1: Příklad zařízení s IP adresou 192.0.2.130/26



Obr. 8.1: Příklad rozdělení sítě do dvou podsítí

8.6 ARP

- *address resolution protocol*
- definován v RFC 826 v roce 1982; v rámci TCP/IP
- protokol k získání linkové adresy (např. MAC adresa), používán IPv4
 - IPv6 – NDP (neighbor discovery protocol)
- funguje v rámci jedné podsítě, využito při posílání dat neopouštějící podsít
- práce na linkové vrstvě (vrstva síťového rozhraní)
- posílání dat
 - host 1 se podívá na DNS na IP hostu 2
 - host 1 se podívá do své routovací tabulky na IP hostu 2
 - IP hostu 2 je nalezeno v cachnuté tabulce
 - * host 1 pošle data na MAC adresu hostu 2
 - IP hostu 2 není v cachnuté tabulce
 - * host 1 vyšle broadcast na `ff:ff:ff:ff:ff:ff` (přijímáno všemi hosty na síti) s požadavkem o MAC adresu IP hostu 2
 - * host 2 odpoví svojí MAC adresou a přitom může přidat hosta 1 do své routovací tabulky

8.7 Směrování (routing)

- určení cesty datagramů (resp. paketů) po síti ke svému cíli
- snaha optimalizace rychlosti a efektivnosti
- zajišťováno síťovou vrstvou ISO/OSI
- cesty mnohdy velmi složitá – řešení vždy pouze jedno kroku / komu dalšímu předat paket
- cesta nalezena na základě směrovacích tabulek

8.7.1 Směrovací tabulky (routing table)

- udržují záznamy o cestách na různá místa sítě
- možné specifikovat ručně, ovšem většiny plněny routing protokolem
- uložena, používána a i vytvářena v aktivních uzlech sítě
- záznamy seřazeny sestupně podle síťové masky (nejkonkrétnější IP adresa / nejdelší maska nahore)
 - poslední záznam – nejbližší router k internetu (default gateway)
- informace záznamu (řádku tabulky)
 - cíl
 - maska podsítě
 - brána
 - síťové rozhraní
- vznik směrovací tabulky
 - statické směrování – tabulka vytvořena administrátorem, neměnná
 - dynamické směrování – reaguje na změny v síti, mění tabulku, využití směrovacích algoritmů

8.7.2 Směrovací algoritmy

- algoritmy zodpovědné za hledání cesty na síti

Distance vektor

- využití Bellman-Fordova algoritmu
- každé spojení nodes má svoji váhu
- nodes posílají pakety po trase s nejmenší sumou vah
- vytváření spojení
 - při zapnutí node ví pouze své přímé sousedy a cenu, za jakou se k nim dostane
 - každá node si se svým sousedem vyměňuje informace o dalších nodes, o kterých ví (výměna routing tables)
 - jestliže přijatá cesta je efektivnější než již známá → zapsání do vlastní tabulky
 - při přidání či ubrání node → update informací
- časem konvergence nejkratších cest v celé síti

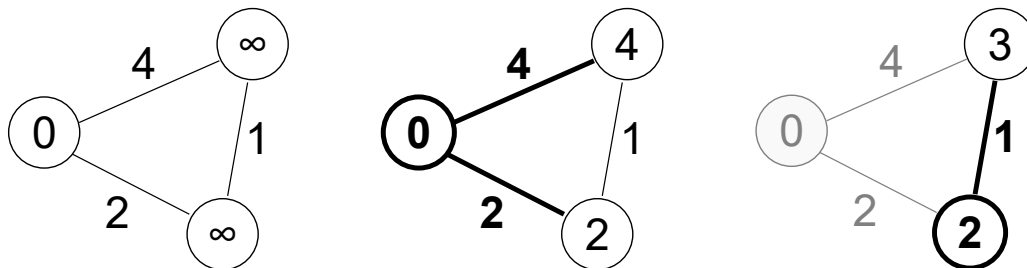
Link state

- node zaplaví síť requestem o sousedech každé jiné node
- z přijatých informací sestaví weighted graph
- na základě grafu zjistí pomocí shortest path algoritmů nejkratší cestu ke každé node
- z výsledku vytvoří tree graph s rootem v momentálním zařízení – cesta k jakémukoliv zařízení je poté pouze cesta tímto stromem

Dijkstrovův algoritmus

- algoritmus používán k hledání cesty v grafu s váhami spojení
- nalezne nejlepší cestu mezi zdrojovou node a všemi ostatními
- postup při hledání nejkratší cesty
 1. Mark all nodes unvisited. Create a set of all the unvisited nodes called the unvisited set.
 2. Assign to every node a tentative distance value: set it to zero for our initial node and to infinity for all other nodes. The tentative distance of a node v is the length of the shortest path discovered so far between the node v and the starting node. Since initially no path is known to any other vertex than the source itself (which is a path of length zero), all other tentative distances are initially set to infinity. Set the initial node as current.[15]
 3. For the current node, consider all of its unvisited neighbors and calculate their tentative distances through the current node. Compare the newly calculated tentative distance to the current assigned value and assign the smaller one. For example, if the current node A is marked with a distance of 6, and the edge connecting it with a neighbor B has length

- 2, then the distance to B through A will be $6 + 2 = 8$. If B was previously marked with a distance greater than 8 then change it to 8. Otherwise, the current value will be kept.
4. When we are done considering all of the unvisited neighbors of the current node, mark the current node as visited and remove it from the unvisited set. A visited node will never be checked again.
 5. If the destination node has been marked visited (when planning a route between two specific nodes) or if the smallest tentative distance among the nodes in the unvisited set is infinity (when planning a complete traversal; occurs when there is no connection between the initial node and remaining unvisited nodes), then stop. The algorithm has finished.
 6. Otherwise, select the unvisited node that is marked with the smallest tentative distance, set it as the new current node, and go back to step 3.



Obr. 8.2: Průběh Dijkstrovova algoritmu na malém grafu