# ## 09 - Adresace v internetu IPv4, cesta datagramu sítí

#### **MAC** adresa

- · Media Access Control
- jedinečný identifikátor síťového zařízení
  - používá ho více vrstev z OSI/ISO
- · je přiřazena síťové kartě při výrobě
- u starších karet v EEPROM paměti (nelze měnit), u moderních již lze změnit
- má 48 bitů, 2 verze zápisu:
  - tři skupiny čtyř hexadecimálních čísel (standard)
  - př.: 0123.4567.89ab
  - šest skupin dvou hexadecimálních čísel (častější)
  - **př.:** 01:23:45:67:89:ab

#### IP adresa

- · číslo, které jednoznačně identifikuje síťové rozhraní v síti
- · používá protokol IP
- adresa je buď přiřazena staticky, nebo získána z poolu DHCP
- IP adresy se dělí do tříd podle prvního bajtu:

| Třída | 1. bajt   | Maska           | Metoda + použití               |  |
|-------|-----------|-----------------|--------------------------------|--|
| А     | 0 - 127   | 255.0.0.0       | unicast, velké sítě            |  |
| В     | 128 - 191 | 255.255.0.0     | unicast, střední sítě          |  |
| С     | 192 - 223 | 255.255.255.0   | unicast, malé sítě             |  |
| D     | 224 - 239 | 255.255.255.255 | multicast                      |  |
| Е     | 240 - 255 |                 | rezerva, experimentální adresy |  |

### Rezervované adresy

- 224.0.0.0 239.255.255 multicasting
- 240.0.0.0 247.255.255.255 experimentální účely
- 127.0.0.0 a 127.0.0.1 loopback, testování software (síťový software, lokální server)
- 10.x.x.x lokální komunikace po soukromé síti
- 255.255.255 broadcast adresa

## Privátní IP adresy

- · neveřejné adresy
- třídy adres A, B, C
- pouze v lokální síti, nejsou dostupné z Internetu

### Veřejné IP adresy

- jedinečná adresa, která označuje počítač na Internetu
- za veřejnou adresou se může skrývat celá lokální síť
  - toto umožňuje NAT
- pro připojení na Internet potřebujeme veřejnou adresu
- tyto adresy přiřazuje IANA (Internet Assigned Numbers Authority)

### **ARP** protokol

- · Address Resolution Protocol
- pomocný protokol sítí TCP/IP
- zabezpečuje přiřazení IP adres k příslušným MAC adresám (L2)
  - uchovává si tabulku
- v případě, že nezná zařízení s příslušnou adresou, vyšle broadcast, počká na odpověď a zapíše si ho do tabulky
  - broadcast se nazývá ARP request
  - o odpoví jen nositel příslušené IP adresy
  - o jako odpověď posílá svou MAC adresu
  - výsledná MAC adresa se ukládá v ARP tabulce

### Subnetting (podsíťování)

- rozdělení jedné síťové adresy na více menších
- používá se např. ve firmách, kde je potřeba logicky rozdělit adresy
- pouze na lokální úrovni
- dělí se na:
  - VLSM (Variable Length Subnet Mask)
  - většinou pro adresy třídy C
  - podsítě mají různou velikost podle potřeby
  - počet podsítí lze vyjádřit jako 2<sup>x</sup>
    - x = počet jedniček v masce
  - velikost sítě se nastavuje na nejbližší velikost, do které se počet hostů vejde
  - Konstatní velikost podsítě
  - o podsítě mají stejnou velikost
  - maska se sníží o určitou velikost
  - ta se poté rozdělí rovnoměrně na stejné díly

### **Porty**

· port naslouchá

#### 1 - 1023

- · Well known (dobře známé) porty
- vyhrazené, pro typicky používané aplikace

| Číslo portu | Služba        |
|-------------|---------------|
| 20          | FTP (data)    |
| 21          | FTP (příkazy) |
| 22          | SSH           |
| 23          | Telnet        |
| 53          | DNS           |
| 67          | DHCP (server) |
| 68          | DHCP (klient) |
| 80          | HTTP          |
| 143         | IMAP          |
| 443         | HTTPS         |
| 666         | hra Doom      |

### 1024 - 49151

- Registrované porty
- jejich použití se musí registrovat u ICANN

#### 49152 - 65535

- Soukromé a dynamické porty
- pro soukromé využití
- nejsou pevně přiděleny žádné službě

### Značení portů

- fyzické porty switche se značí:
  - typem
  - FastEthernet, GigabitEthernet
  - číslem portu
  - [slot]/[port] nebo [stack]/[slot]/[port]
- · příklady značení:
  - FastEthernet0/1
  - 100Mbit/s, slot č. 0, port č. 1
  - GigabitEternet1/0/1
  - 1Gbit/s, stack č. 1, slot č. 0, port č. 1

# Komunikace a přenos dat

- serverová aplikace poslouchá na lokálním portu, čeká na požadavek o připojení
- klientská aplikace si vyžádá připojení
- porty se vzájemně propojí
- na obou stranách vzniká Socket
  - koncový bod připojení
- do Socketu se zapouzdří informace o spojení
- server port zůstává otevřený a přijímá další žádosti