**Binary / decimal / hex systems**

Binary – powers of 2 (each subsequent left slot represents an increase in power of 2 (so 32, 16, 8, 4, 2, 1)

Decimal – powers of 10 (an increase in power of 10 (so 10 000, 1 000, 100, 10, 1)

Hexadecimal – powers of 16 (an increase in power of 16 (so 4096, 256, 16, 1)

Equivalence table:

Binary - 0 1 10 11 100 101 110 111 1000 1001 1010 1011 1100 1101 1110 1111 10000

Decimal - 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

Hex - 0 1 2 3 4 5 6 7 8 9 A B C D E F 10

Formula – xyzn = x\*base^3 + y\*base^2 + z\*base^1 + n\*1

Decimal example – 3487 = 3\*10^3 + 4\*10^2 + 8\*10^1 + 7 = 3487

Binary example – 1101 = 1\*2^3 + 1\*2^2 + 0\*2^1 + 1 = 13

Hex example – FACE = F\*16^3 + A\*16^2 + C\*16^1 + E = 64206

**IP Addressing**

What is an IP address?

* Assigned to a network card, it is the address of that card on a network. Divided in two parts – first three sets signify the network (that is, a group of devices - say, 203.107.0), while the last set signifies the host (an individual device on that specific network)
* Same way as our address has an individual identifier for our house (street number, apartment) and an identifier for the area (city, postal code)
* An IP address is always 32 bits long – 4 sets of 8 bits (called octets). In decimal it might be 1 to 3 numbers long, but it is always 8 binary bits (hence 255 being the biggest).

**Classless addressing** – we determine which part of the IP address refers to the network and which part refers to the host by using a **subnet mask**. A subnet mask has full 1’s in the network portion and 0’s in the host portion (so a standard mask leaving the last octet for host and the first three for network looks like 255.255.255.0 in decimal). It does not need to be in octets!! We might as well have the first 10 bits as the network portion, and the remaining 22 as host (pls don’t).

**Classful addressing** – fuck me. Total address space (0.0.0.0 to 255.255.255.255) is divided in a few parts (classes). Depending on which part an address is in, its network portion is determined by its first one, two, or three octets. (A – 1, B – 2, C – 3, D – 4 (that is all) – used for multiclass broadcast, E – experimental, not used).

**IP Address Types** – **MIND THE FUCIKNG SUBNET MASK!! IT MIGHT BE DIFFERENT FROM 255.255.255.0**

* Network address (network prefix)
  + Signifies the group of devices
  + If an address has all binary 0’s in the host portion – then it is a network address
  + NEVER ASSIGN TO A DEVICE ON THE NETWORK
* Broadcast address
  + Wtf? A way to broadcast to all devices on the network at once?!?
  + If an address has all binary 1’s in the host portion – then it is a broadcast address
  + NEVER ASSIGN TO A DEVICE ON THE NETWORK
* Host address
  + A particular machine (computer, printer, etc) on the network
  + If an address has anything whatsoever EXCEPT all 0’s or all 1’s in the host portion
  + Free to assign devices here

**Private vs public IP addresses**

* Private IP addresses can only be used within an internal network (work, school, home, etc)
  + Reserved ranges
    - Full 10. (i.e. 10.0.0.0 – 10.255.255.255)
    - 172.16.0.0 – 172.31.255.255
    - 192.168.0.0 – 192.168.255.255
    - Also – 127.0.0.1 – loopback address (localhost)

**Subnetting**

Instead of writing down the mask every time, we can use the CIDR (Classless Inter-Domain Routing) notation. So instead of writing 203.0.113.10 with a subnet mask of 255.255.255.0, we can write 203.0.113.10**/24,** indicating that the subnet mask is 24-bit long.

**MIND THE MASK!**

203.0.113.10**/27** and 203.0.113.36**/27** and in **different networks**, even if it might not seem like that at first sight. Convert to binary to see why.

**A simple subnet calculator**

Tell us how many hosts can we have with X bits in the host portion of the subnet mask, and how many networks can we have given X bits in the network portion of the mask.

|  |  |  |
| --- | --- | --- |
| Networks (2^network bits) | Bits | Hosts (2^hosts bits - 2) |
| 1 | 0 | 0 |
| 2 | 1 | 0 |
| 4 | 2 | 2 |
| 8 | 3 | 6 |
| 16 | 4 | 14 |
| 32 | 5 | 30 |
| 64 | 6 | 62 |
| 128 | 7 | 126 |
| 256 | 8 | 254 |
| 512 | 9 | 510 |
| 1024 | 10 | 1022 |
| 2048 | 11 | 2046 |

172.16.128.0/17 – 30 networks – 5 bits

10101100 00010000 10000000 00000000

10101100 00010000 10000000 00000000– 172.16.128.0/22 – Network 0

10101100 00010000 10000100 00000000 – 172.16.132.0/22 – Network 1

10101100 00010000 10101000 00000000 – 172.16.168.0/22 – Network 10

10101100 00010000 11011100 00000000 – 172.16.220.0/22 – Network 23

10101100 00010000 10111100 00000000 – 172.16.188.0/22 – Network 15

10101100 00010000 10011100 00000000 – 172.16.156.0/22 – Network 7

10101100 00010000 11000000 00000000 – 172.16.192.0/22 – Network 16

10101100 00010000 11100100 00000000 – 172.16.228.0/22 – Network 25

172.17.20.0/22 – 17 networks – 5 bits

10101100 00010001 00010100 00000000

10101100 00010001 00010100 00000000 – 172.17.20.0/27 – Network 0

10101100 00010001 00010100 00100000 – 172.17.20.32/27 – Network 1

10101100 00010001 00010100 01000000 – 172.17.20.64/27 – Network 2

10101100 00010001 00010100 10100000 – 172.17.20.160/27 – Network 5

10101100 00010001 00010100 11100000 – 172.17.20.224/27 – Network 7

10101100 00010001 00010101 11100000 – 172.17.21.224/27 – Network 15

10101100 00010001 00010110 10000000 – 172.17.22.128/27 – Network 20

172.31.96.0/19 – 300 networks – 9 bits

10101100 00011111 01100000 00000000

10101100 00011111 01100000 00000000 – 172.31.96.0/28 – Network 0

10101100 00011111 01100000 00010000 – 172.31.96.16/28 – Network 1

10101100 00011111 01100011 01110000 – 172.31.99.112/28 – Network 55

10101100 00011111 01100100 10000000 – 172.31.100.128/28 – Network 72

10101100 00011111 01101000 00000000 – 172.31.104.0/28 – Network 128

10101100 00011111 01101001 00010000 – 172.31.105.16/28 – Network 145