IP Basics

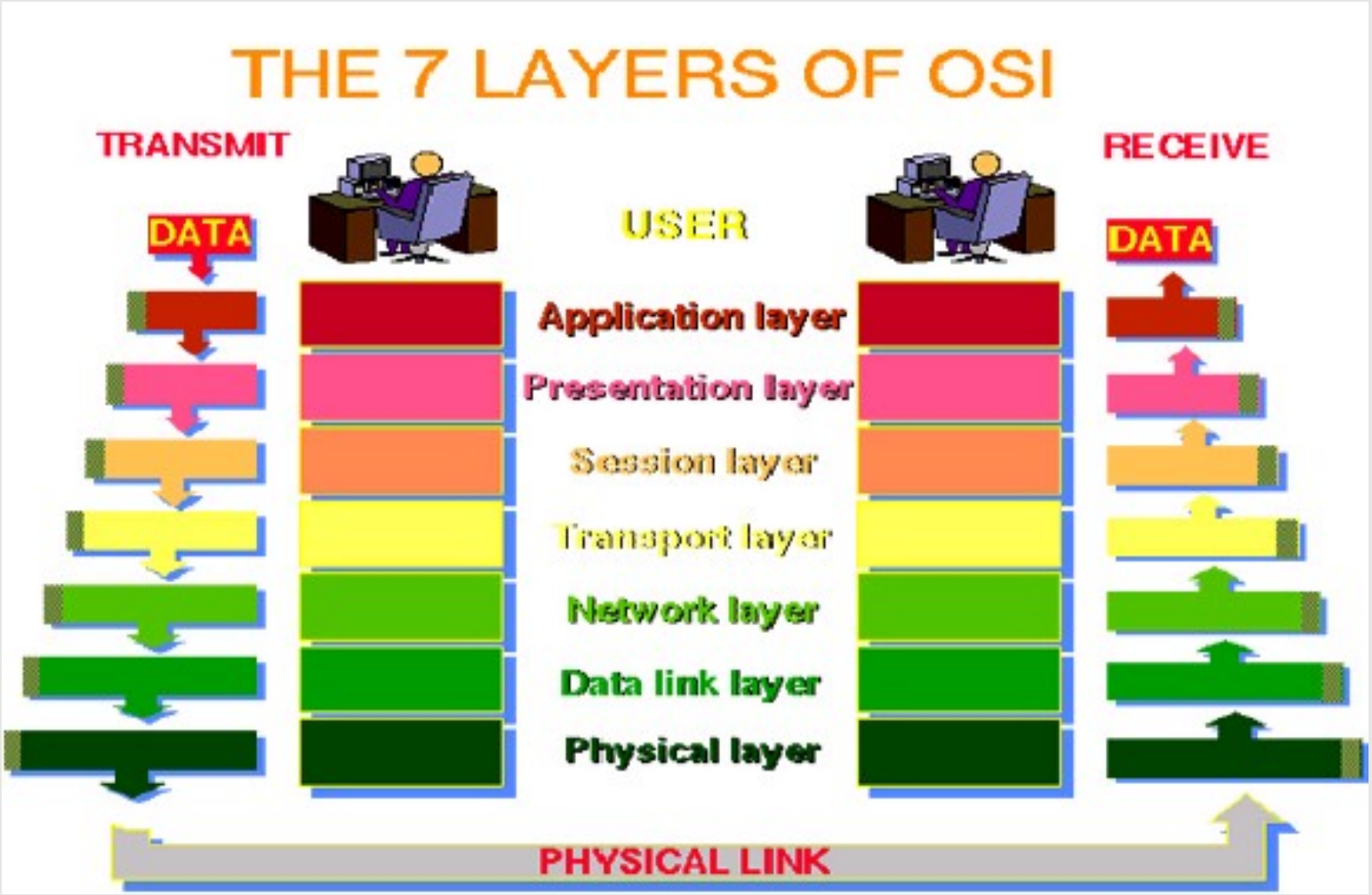
- How the computer transport system works

AFNOG IX
Rabat, Morocco
May 2008

Layers

- Complex problems can be solved using the common divide and conquer principle. In this case the internals of the Internet are divided into separate layers.
  - Makes it easier to understand
  - Developments in one layer need not require changes in another layer
  - Easy formation (and quick testing of conformation to) standards
- Two main models of layers are used:
  - OSI (Open Systems Interconnection)
  - TCP/IP

OSI Model



AfNOG

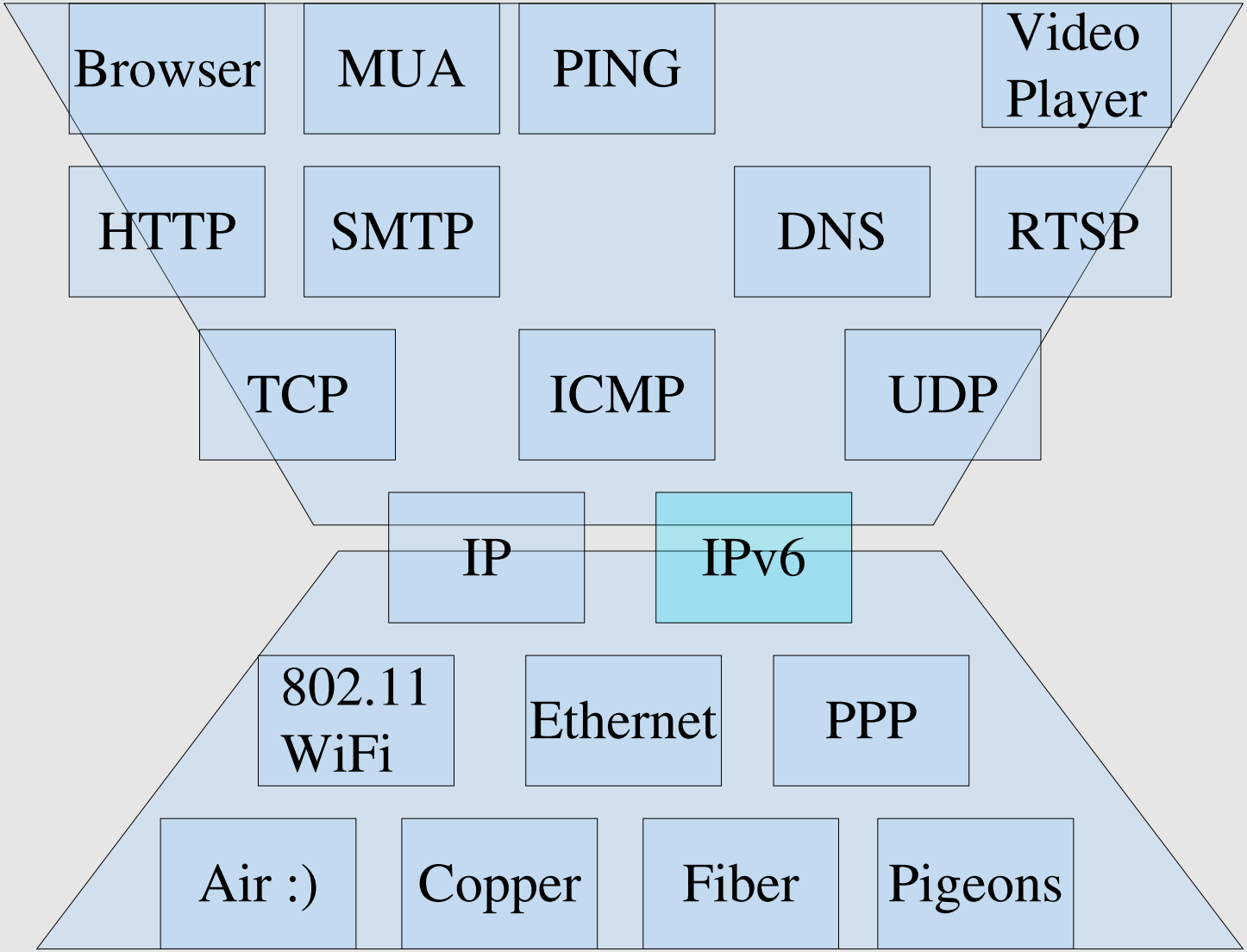
OSI

- Conceptual model composed of seven layers, developed by the International Organization for Standardization (ISO) in 1984.
  - Layer 7 – Application (servers and clients etc web browsers, httpd)
  - Layer 6 – Presentation (file formats e.g pdf, ASCII, jpeg etc)
  - Layer 5 – Session (conversation initialisation, termination,)
  - Layer 4 – Transport (inter host comm – error correction, QOS)
  - Layer 3 – Network (routing – path determination, IP[x] addresses etc)
  - Layer 2 – Data link (switching – media acces, MAC addresses etc)
  - Layer 1 – Physical (signalling – representation of binary digits)
- Acronym: All People Seem To Need Data Processing

TCP/IP

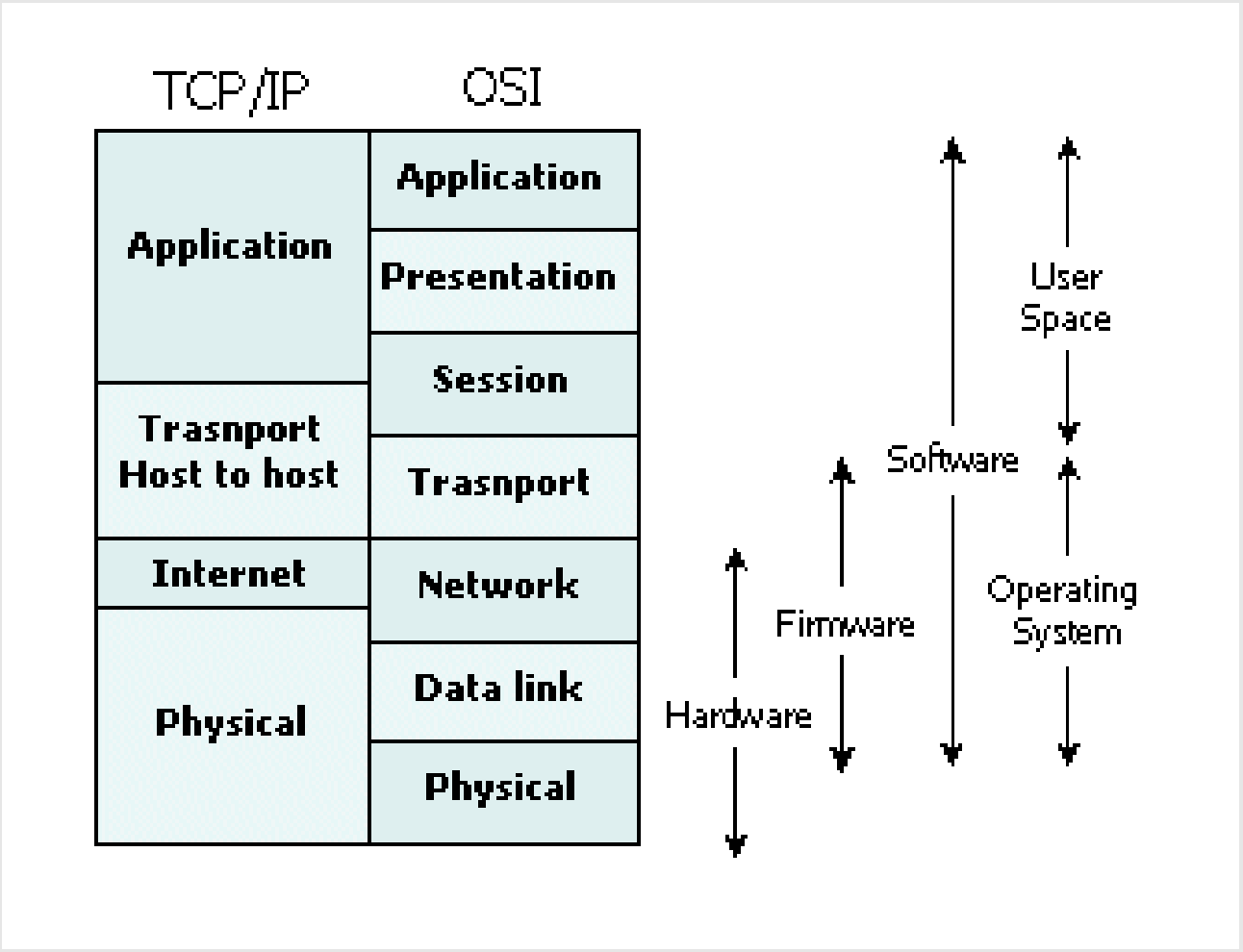
- Generally, TCP/IP (Transmission Control Protocol/Internet Protocol) is described using three to five functional layers. We have chosen the common DoD reference model, which is also known as the Internet reference model.
  - Process/Application Layer consists of applications and processes that use the network.
  - Host-to-host transport layer provides end-to-end data delivery * services.
  - Internetwork layer defines the datagram and handles the routing of data.
  - Network access layer consists of routines for accessing physical networks.

TCP/IP model – the “hourglass”



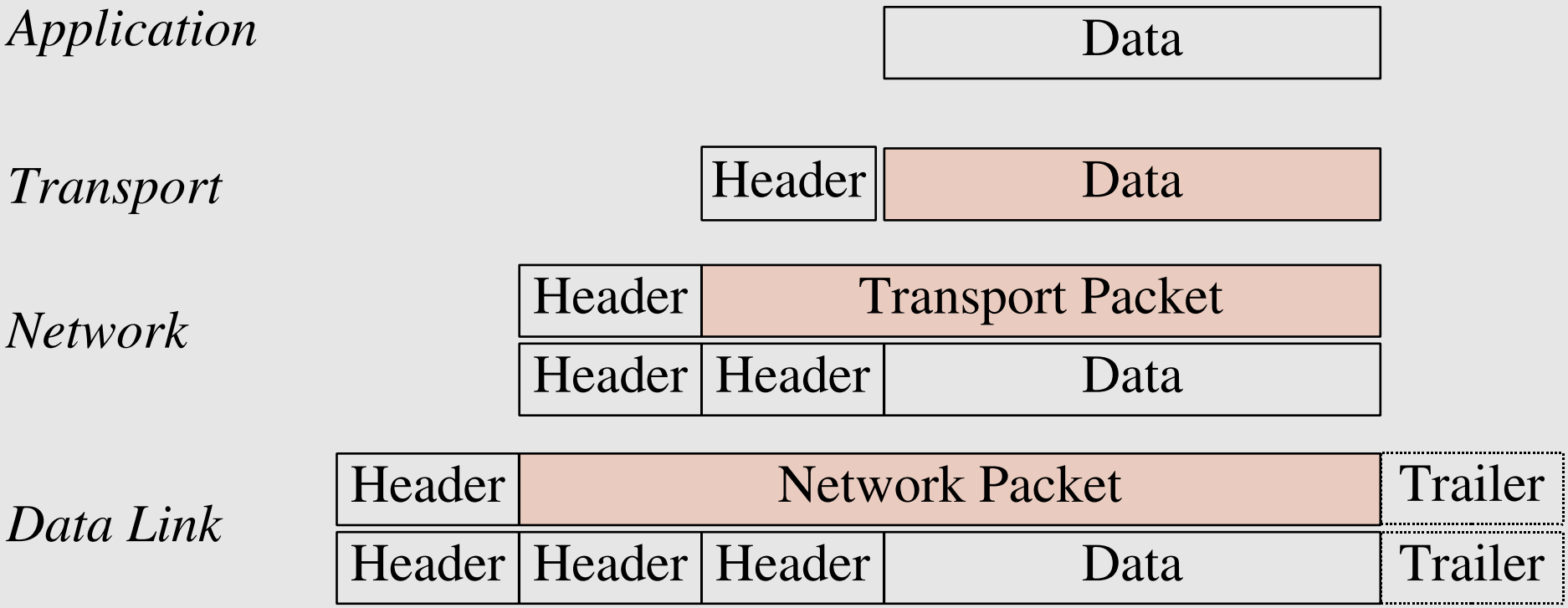
AfNOG

OSI and TCP/IP



Encapsulation & Decapsulation

- Lower layers add headers (and sometimes trailers) to upper layers packets



Frame, Datagram, Segment, Packet

- Different names for packets at different layers
  - Ethernet (link layer) frame
  - IP (network layer) datagram
  - TCP (transport layer) segment
- Terminology is not strictly followed
  - we often just use the term “packet” at any layer

So what is an IP address anyway?

- IPv4 :
  32 bit number (4 octet number) can be represented in lots of ways:

133	27	162	125

10000101	00011011	10100010	01111101

85	1B	A2	7D

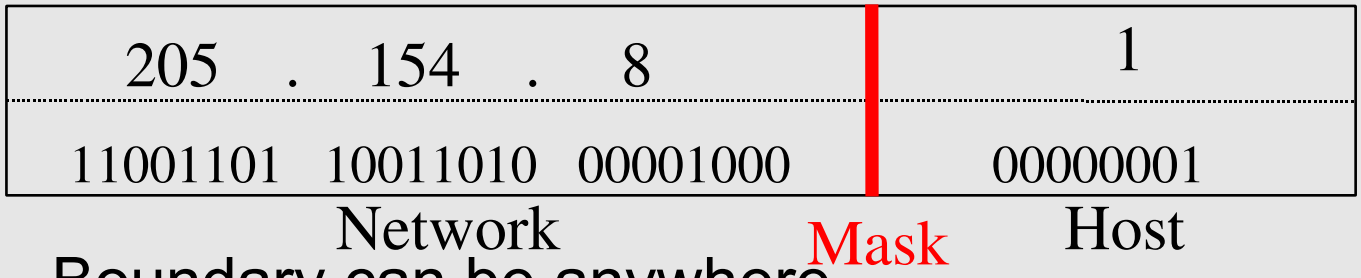
So what is an IP address anyway?

- IPv6 :
  16 bit fields in case insensitive colon hexadecimal representation:

- 2031:0000:130F:0000:0000:09C0:876A:130B
  <=>
- 2031:0:130f::9c0:876a:130b
  - Successive groups of 0 can be “compressed” into :: (only once)
  - Leading 0's can be dropped.

More to the structure

- Hierarchical Division in IP Address:
  - Network Part (Prefix)
    - describes which network
  - Host Part (Host Address)
    - describes which host on that network



  - Boundary can be anywhere
    - used to be a multiple of 8 (/8, /16/, /24), but not usual today
- Same idea in IPv6 – address and prefix.
- IPv6 introduces a concept of scopes
  - Global, link-local, unique-local (this one is unclear)

Network Masks

- Network Masks help define which bits are used to describe the Network Part and which for hosts
- Different Representations:
  - decimal dot notation: 255.255.224.0 (128+64+32 in byte 3)
  - binary: 11111111 11111111 111 00000 00000000
  - hexadecimal: 0xFFFFE000
  - number of network bits: /19 (8 + 8 + 3)
- IPv4: Binary AND of 32 bit IP address with 32 bit netmask yields network part of address
- IPv6: The prefix is expressed as /length (/48, /64, /80, ...). Everything else (including masking of prefix to find whether host is local) is the same.

Sample Netmasks

137.158.128.0/**17** (netmask **255.255.128.0**)

1111 1111	**1111 1111**	**1**	**000 0000**	**0000 0000**
1000 1001	1001 1110	1	000 0000	0000 0000

198.134.0.0/**16** (netmask **255.255.0.0**)

1111 1111	**1111 1111**	**0000 0000**	**0000 0000**
1100 0110	1000 0110	0000 0000	0000 0000

205.37.193.128/**26** (netmask **255.255.255.192**)

1111 1111	**1111 1111**	**1111 1111**	**11**	**00 0000**
1100 1101	0010 0101	1100 0001	10	00 0000

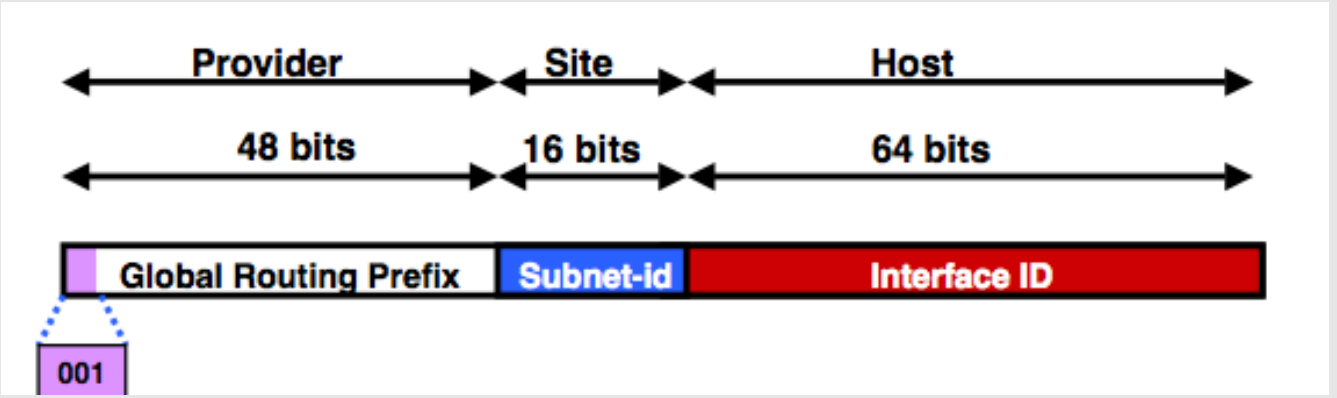
AfNOG

Allocating IP addresses

- The subnet mask is used to define size of a network
- E.g a subnet mask of 255.255.255.0 or /24 implies 32-24=8 host bits
  - 2^8 minus 2 = 254 possible hosts
- Similarly a subnet mask of 255.255.255.224 or /27 implies 32-27=5 hosts bits
  - 2^5 minus 2 = 30 possible hosts
- Same mechanism in IPv6 :
  - 2001:4348:0:218:196:200:218:1/64
  - Some differences though... (implicit prefix prefix length for example, /64 by default for segments. But IPv6 **IS** classless!)

Allocating IP addresses - IPv6

- The IPv6 Host part of the address (/64) is allocated in one of the following ways
  - Auto-configured from a 64-bit EUI-64, or expanded from a 48-bit MAC address (e.g., Ethernet address)
  - Auto-generated pseudo-random number (to address privacy concerns) – not widely implemented
  - Assigned via DHCP (not many implementations of DHCPv6)
  - Manually configured (most familiar)



- *Note: thanks to Phil Smith for his excellent presentation!*

Special IP Addresses - IPv4

- All 0's in host part: Represents Network
  - e.g. 193.0.0.0/24
  - e.g. 138.37.128.0/17
  - e.g. 192.168.2.128/25 (WHY ?)
- All 1's in host part: Broadcast (all hosts on net)
  - e.g. 137.156.255.255 (137.156.0.0/16)
  - e.g. 134.132.100.255 (134.132.100.0/24)
  - e.g. 192.168.2.127/25 (192.168.2.0/25) (WHY ?)
- 127.0.0.0/8: Loopback address (127.0.0.1)
- 0.0.0.0: Various special purposes (DHCP, etc...)

Special IP Addresses - IPv6

- IPv6 has a few changes
  - No more broadcast address – use of Multicast instead
  - FF02::1 → All nodes on the local network segment
  - :: → 0:0:0:0:0:0:0:0 → unspecified address
  - ::1 → 0:0:0:0:0:0:0:1 → loopback address

A word on ARP

- One mechanism which we don't really cover is ARP. ARP is the Address Resolution protocol. ARP is used in IPv4, to solve the problem of converting IPv4 address to ethernet MAC (Media Access Control), for finding hosts on the local link.
- ARP functions using the broadcast address on ethernet:
- `10:43:56.916523 00:1e:0b:b2:f7:e0 > ff:ff:ff:ff:ff:ff, ethertype ARP (0x0806), length 42: arp who-has 196.200.218.254 tell 196.200.218.1`
- `10:43:56.916906 00:1c:58:22:1c:e0 > 00:1e:0b:b2:f7:e0, ethertype ARP (0x0806), length 60: arp reply 196.200.218.254 is-at 00:1c:58:22:1c:e0`
-

IPv6: NDP

- IPv6 doesn't have ARP.
- Neighbor Discovery Protocol is used instead:
- 11:40:26.621642 00:1e:0b:b2:f7:e0 > 33:33:ff:18:02:00, ethertype IPv6 (0x86dd), length 86: 2001:4348:0:218:196:200:218:1 > ff02::1:ff18:200: ICMP6, neighbor solicitation, who has 2001:4348:0:218:196:200:218:200, length 32

- 11:40:26.622154 00:1e:0b:b5:a3:c9 > 00:1e:0b:b2:f7:e0, ethertype IPv6 (0x86dd), length 86: 2001:4348:0:218:196:200:218:200 > 2001:4348:0:218:196:200:218:1: ICMP6, neighbor advertisement, tgt is 2001:4348:0:218:196:200:218:200, length 32
- Note the used of ff02::1:ff multicast address to sollicit an answer (solicited-node address) – the last 24 bits are from the address being solicited.

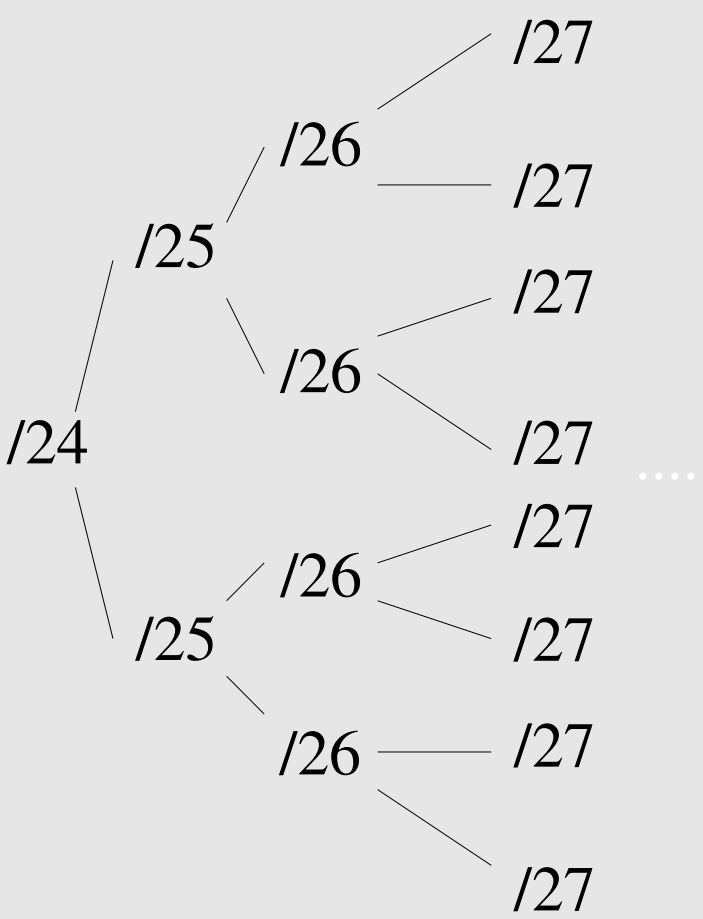
IPv6: NDP

- Another NDP, this time for duplicate address detection:

- 11:35:56.501752 00:1e:0b:b5:a3:c9 > 00:1e:0b:b2:f7:e0, ethertype IPv6 (0x86dd), length 86: 2001:4348:0:218:196:200:218:200 > 2001:4348:0:218:196:200:218:1: ICMP6, neighbor solicitation, who has 2001:4348:0:218:196:200:218:1, length 32

- 11:35:56.501767 00:1e:0b:b2:f7:e0 > 00:1e:0b:b5:a3:c9, ethertype IPv6 (0x86dd), length 78: 2001:4348:0:218:196:200:218:1 > 2001:4348:0:218:196:200:218:200: ICMP6, neighbor advertisement, tgt is 2001:4348:0:218:196:200:218:1, length 24

Networks – super- and subnetting

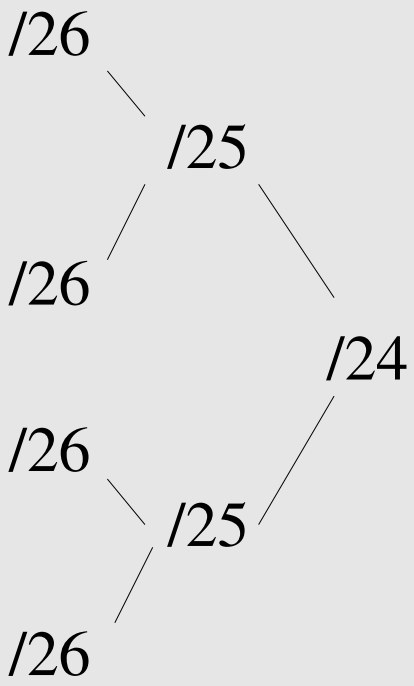


By adding one bit to the netmask, we subdivide the network into two smaller networks. This is *subnetting*.

i.e.: If one has a /26 network (32 – 26 = 6 => 2^6 => 64 addresses), that network can be subdivided into two subnets, using a /27 netmask, where the state of the last bit will determine which network we are addressing (32 – 27 = 5 => 2^5 => 32 addresses). This can be done recursively (/27 => 2 x /28 or 4 x /29, etc...).

Example: 192.168.10.0/25 (.0 - .127) can be subnetted into 192.168.10.0 / 26 and 192.168.10.64 / 26

Networks – super- and subnetting



Inversely, if two networks can be "joined" together under the same netmask, which encompasses both networks, then we are *supernetting*.

Example:

Networks 10.254.4.0/24 and 10.254.5.0/24 can be "joined" together into one network expressed: 10.254.4.0/23.

Note: for this to be possible, the networks must be *contiguous*, i.e. it is not possible to supernet 10.254.5.0/24 and 10.254.6.0/24

Subnetting in IPv6

- Concept is the same as in v4
- Network administrator can subnet as he wants
  - /126 for point-to-point if one wants
  - /96 for a subnet
  - Etc...
- Only constraint is use of /64 for local net if autoconfiguration is to be used
- Otherwise there are no restrictions

Fun with subnets

Numbering Rules - IPv4

- Private IP address ranges (RFC 1918)
  - 10/8 (10.0.0.0 – 10.255.255.255)
  - 192.168/16 (192.168.0.0 – 192.168.255.255)
  - 172.16/12 (172.16.0.0 – 172.31.255.255)
- Public Address space available from AfriNIC
- Choose a small block from whatever range you have, and subnet your networks (to avoid problems with broadcasts, and implement segmentation policies – DMZ, internal, etc...)

Numbering Rules - IPv6

- No private IP space in IPv6 like in IPv4
- Concept of unique-local (FC00::/7), previously site-local
- Not really used (for the site only)
- At the link level: link local addresses
  - FE80::/10
- Public Address space available from AfriNIC as well
- Allocated as /32s for ISPs
- ISPs assign /48 to customers
- This is a policy choice, not a technical constraint

FreeBSD IP related settings

- In /etc/rc.conf (manual configuration)

```
hostname=“pcX.sae.ws.afnog.org”

# IPv4
ifconfig_em0=“196.200.218.x”
defaultrouter=“196.200.218.254”

# IPv6
ipv6_enable="YES"
ipv6_ifconfig_le0="2001:4348:0:218:196:200:218:x"
ipv6_defaultrouter=”
2001:4348:0:218:196:200:218:254"
```

Reaching hosts on the local net

- If you want to talk to other computers on the same network (e.g: withing the same IP subnet, not necessarily the same physical network!), this is automatically possible the moment you assign an IP address to your network card.
- We will see this later with the hands-on

Reaching hosts on other networks

- If a computer isn't on your subnet, to reach it packets must be sent via a “gateway” connected to your network (“next hop”).
- If not explicit route (“direction”) is given on how to reach a particular network you want to talk to, then the computer will try a last resort “default gateway” for all packets that are not local
- defaultrouter (v4) option in /etc/rc.conf sets the default gateway for *this* system, and ipv6_defaultrouter (v6)
  - Note: on IPv6, the next hop *could* be a link-local address, even though you have manually configured your interface for v6 – but it's still directly reachable

Forwarding packets

- Any UNIX-like (and other) operating system can function as gateway (e.g.: forwarding packets from one interface to another)
- IP forwarding on a FreeBSD box turned on with the gateway_enable option in /etc/rc.conf (v4) or ipv6_gateway_enable (v6)
  - Remember that v4 and v6 are *different* protocol stacks
- Without forwarding enabled, the box will not forward packets from one interface to another: it is simply a host with multiple interfaces.

Packet Routing Exercise

Client – Server Architecture

- Client makes requests, Server serves requests – e.g HTTP for transferring “web pages”. This is the easiest way to provide services on demand and provides a means of sharing resources more effectively.
- Example: Mimicking the browser with telnet (client) talking to a web server (server)

  # telnet www.google.com 80

  GET / HTTP/1.0
  Host: www.google.com
  <blank line>

Debugging

- ping
- traceroute
- tcpdump
- arp
- route

AfNOG

References

- IPv6:
  - http://www.nanog.org/mtg-0802/presentations/smith-ipv6.pdf
  - http://www.nanog.org/mtg-0802/wmv/NANOG-42-Tutorial-Intro-v6.wmv

AfNOG