



# Routing

---

Spring 22 TAs: Lakiotakis Manos, Plevridi Eleftheria  
manoslak@csd.uoc.gr, plevridi@csd.uoc.gr  
Computer Science Department, University of Crete

# IPv4 Addressing

- Lets first take a look at the IPv4 header

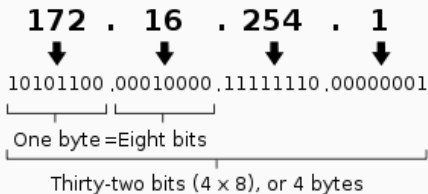
Bits	0	3	4	7	9	15	16	31
	Version		Header length		Type of service		Total length	
	Identification					Flags	Fragment offset	
	Time to live			Protocol		Header checksum		
32-bit source address								
32-bit destination address								
Options							Padding	

# IPv4 Addressing

- 32-bit addresses  $\rightarrow 2^{32}$  different IP addresses
- Not all of them can be used
- The address space of IP addresses is controlled by a global organization, the IANA ( <http://www.iana.org> )
- IPv4 address assignment can be found at the *IANA resource pages*
- IPv4 addresses are exhausted
- Solution: NAT, **IPv6**

# IPv4 Addressing

- In order IP addresses to be easily remembered, the decimal dot-notation is used
- 32 bits are divided into four octets
- We calculate the number of each octet
- After each octet (except the last!) we place a dot (.)
- For example:



# Classless Inter-Domain Routing

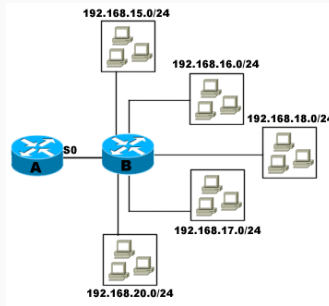
- CIDR → Classless Inter-Domain Routing
- CIDR is a method for allocating IP addresses and routing IP packets efficiently
- An IP address is divided into two parts:
  - Most significant bits, are called **network address** or **subnet**
  - Less significant bits are the **host identifier**
- CIDR notation: 192.168.1.0/24
  - 192.168.1.0 is the **network ID**
  - /x part indicates the number (x) of the most significant bits

# Reserved IP addresses

Range	Description
0.0.0.0/8	Current network (only valid as source address)
10.0.0.0/8	<a href="#">Private network</a>
100.64.0.0/10	Shared Address Space
127.0.0.0/8	<a href="#">Loopback</a>
169.254.0.0/16	<a href="#">Link-local</a>
172.16.0.0/12	<a href="#">Private network</a>
192.0.0.0/24	Reserved (IANA)
192.0.2.0/24	TEST-NET-1, documentation and examples
192.88.99.0/24	<a href="#">IPv6</a> to IPv4 relay
192.168.0.0/16	<a href="#">Private network</a>
198.18.0.0/15	Network benchmark tests
198.51.100.0/24	TEST-NET-2, documentation and examples
203.0.113.0/24	TEST-NET-3, documentation and examples
224.0.0.0/4	<a href="#">IP multicast</a> (former Class D network)
240.0.0.0/4	Reserved (former Class E network)
255.255.255.255	Broadcast

# Supernetting

- A subnet may contain several smaller subnets



- Easier routing, administration and more robust topologies
- Isolation (CSD may have its own smaller subnet rather than be in the 'huge' /16 subnet of UoC)

## Question 1

147.52.0.0/16 Which is the network ID? Which is the subnet?  
Which is the start and the end IP?

## Question 2

A /16 network how many IP addresses may have?

## Question 3

How many /24 subnets can have, a /16 subnet?

## Question 4

A network has a range of IP's from 10.0.0.0 – 10.255.255.255.  
Which is the network ID and the subnet?



- IP addresses can be assigned statically
  - Human interaction is needed
  - If the topology or the network change, modifications must be done again
- IP addresses can be also assigned dynamically
  - DHCP (Dynamic Host Configuration Protocol)
  - It's a server-client protocol
  - Only the server must be configured
  - Clients automatically get the proper IP (and other info) from the server

# What is routing?

- Routing is the process of selecting best paths in a network
- Packets from a specific source are propagated appropriately to router devices until the destination is reached
- In IP routing the procedure works in a **best-effort** way! No guarantees for the delivery of the packet
- Fundamental element of routing is the **routing table**

# Routing tables

- Each router keeps a routing table

Kernel IP routing table						
Destination	Gateway	Genmask	Flags	Metric	Ref	Use Iface
0.0.0.0	10.0.0.6	0.0.0.0	UG	0	0	0 eth1
10.0.0.0	0.0.0.0	255.255.255.0	U	0	0	0 eth1
10.0.1.0	0.0.0.0	255.255.255.0	U	0	0	0 eth2
10.0.2.0	0.0.0.0	255.255.255.0	U	0	0	0 eth1.120
127.0.0.0	0.0.0.0	255.0.0.0	U	0	0	0 lo
192.168.4.0	0.0.0.0	255.255.252.0	U	0	0	0 eth2.2

# Routing tables

- Each router keeps a routing table

Kernel IP routing table						
Destination	Gateway	Genmask	Flags	Metric	Ref	Use Iface
0.0.0.0	10.0.0.6	0.0.0.0	UG	0	0	0 eth1
10.0.0.0	0.0.0.0	255.255.255.0	U	0	0	0 eth1
10.0.1.0	0.0.0.0	255.255.255.0	U	0	0	0 eth2
10.0.2.0	0.0.0.0	255.255.255.0	U	0	0	0 eth1.120
127.0.0.0	0.0.0.0	255.0.0.0	U	0	0	0 lo
192.168.4.0	0.0.0.0	255.255.252.0	U	0	0	0 eth2.2

In Linux systems, routing table can be retrieved using the **ip route** command

# Routing tables

- Routing tables include the following fields:
  - **Destination:** The destination network or destination host
  - **Gateway:** The gateway address or '\*' if none set
  - **Genmask:** The subnet for the destination network
  - **Iface:** Interface to which packets for this route will be sent

Kernel IP routing table							
Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
0.0.0.0	10.0.0.6	0.0.0.0	UG	0	0	0	eth1
10.0.0.0	0.0.0.0	255.255.255.0	U	0	0	0	eth1
10.0.1.0	0.0.0.0	255.255.255.0	U	0	0	0	eth2
10.0.2.0	0.0.0.0	255.255.255.0	U	0	0	0	eth1.120
127.0.0.0	0.0.0.0	255.0.0.0	U	0	0	0	lo
192.168.4.0	0.0.0.0	255.255.252.0	U	0	0	0	eth2.2

- Depending on the maintainer of the routing table routing can be:
  - Static
  - Dynamic

- In static routing, routing tables are created and maintained by administrators
- Suitable for small networks
- Requires human interaction

If the network topology changes for any reason, reconfiguration and human intervention is required

# Dynamic Routing

- Dynamic routing relies on a protocol that automatically selects the best paths
- Dynamic routing protocols are based on two basic algorithms to find the best path:
  - **Link state** (e.g. Dijkstra's algorithm)
  - **Distance vector** (e.g. Bellman–Ford algorithm)
- Established and well known dynamic routing protocols are:
  - **RIP**
  - **OSPF**
  - **BGP**
  - **IGRP/EIGRP**
- These protocols constantly alter routing table entries on a router as the network conditions and topology changes

# Longest Prefix Matching

- A destination address may match in multiple destination entries of a routing table
- The rule of the **Longest Prefix Matching** dictates that the entry with the greatest number of identical bits between the destination address of the incoming packet and the destination network of the routing table entry will be used



# Longest Prefix Matching

#	Destination	Gateway
1	0.0.0.0/0	147.52.80.1
2	147.52.0.0/16	147.52.17.1
3	147.52.20.0/25	147.52.20.1
4	147.52.20.34/32	147.52.20.2

## Examples:

- 5.21.30.1

# Longest Prefix Matching

#	Destination	Gateway
1	0.0.0.0/0	147.52.80.1
2	147.52.0.0/16	147.52.17.1
3	147.52.20.0/25	147.52.20.1
4	147.52.20.34/32	147.52.20.2

## Examples:

- 5.21.30.1 → 147.52.80.1
- 147.52.1.3

# Longest Prefix Matching

#	Destination	Gateway
1	0.0.0.0/0	147.52.80.1
2	147.52.0.0/16	147.52.17.1
3	147.52.20.0/25	147.52.20.1
4	147.52.20.34/32	147.52.20.2

## Examples:

- 5.21.30.1 → 147.52.80.1
- 147.52.1.3 → 147.52.17.1
- 147.52.20.3

# Longest Prefix Matching

#	Destination	Gateway
1	0.0.0.0/0	147.52.80.1
2	147.52.0.0/16	147.52.17.1
3	147.52.20.0/25	147.52.20.1
4	147.52.20.34/32	147.52.20.2

## Examples:

- 5.21.30.1 → 147.52.80.1
- 147.52.1.3 → 147.52.17.1
- 147.52.20.3 → 147.52.20.1
- 147.52.20.140

# Longest Prefix Matching

#	Destination	Gateway
1	0.0.0.0/0	147.52.80.1
2	147.52.0.0/16	147.52.17.1
3	147.52.20.0/25	147.52.20.1
4	147.52.20.34/32	147.52.20.2

## Examples:

- 5.21.30.1 → 147.52.80.1
- 147.52.1.3 → 147.52.17.1
- 147.52.20.3 → 147.52.20.1
- 147.52.20.140 → 147.52.17.1
- 147.52.20.34

# Longest Prefix Matching

#	Destination	Gateway
1	0.0.0.0/0	147.52.80.1
2	147.52.0.0/16	147.52.17.1
3	147.52.20.0/25	147.52.20.1
4	147.52.20.34/32	147.52.20.2

## Examples:

- 5.21.30.1 → 147.52.80.1
- 147.52.1.3 → 147.52.17.1
- 147.52.20.3 → 147.52.20.1
- 147.52.20.140 → 147.52.17.1
- 147.52.20.34 → 147.52.20.2

# Routing debugging tools

- ping
- traceroute
- ip route
- ip route get (for testing which routing entry will be selected from the host)
- arping