Computer Networks - *Xarxes de Computadors*

Outline

- Course Syllabus
- Unit 1: Introduction
- **Unit 2. IP Networks**
- Unit 3. Point to Point Protocols -TCP
- Unit 4. LANs
- Unit 5. Data Transmission

Unit 2: IP Networks

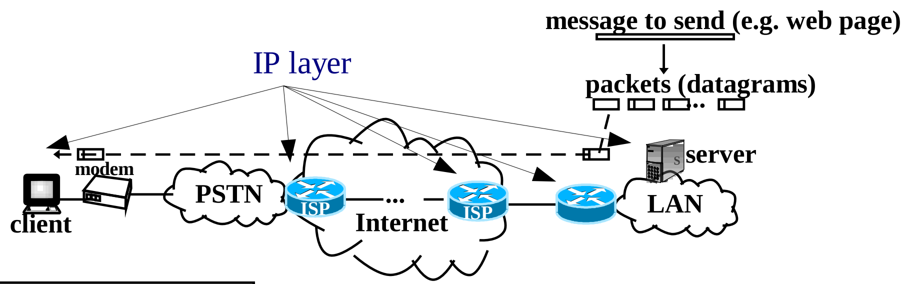
Outline

- **IP layer service**
- IP addresses
- Subnetting
- Routing tables
- ARP protocol
- IP header
- ICMP protocol
- DHCP protocol
- NAT
- DNS
- Routing algorithms
- Security in IP

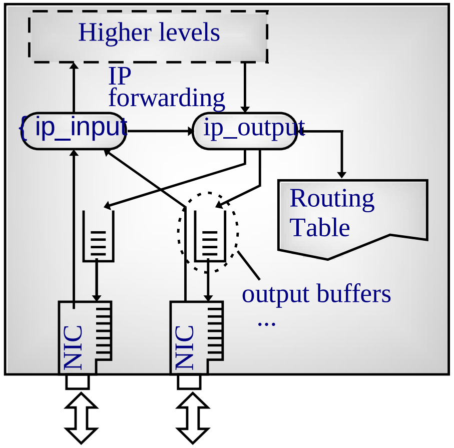
Unit 2: IP Networks

IP Layer Service

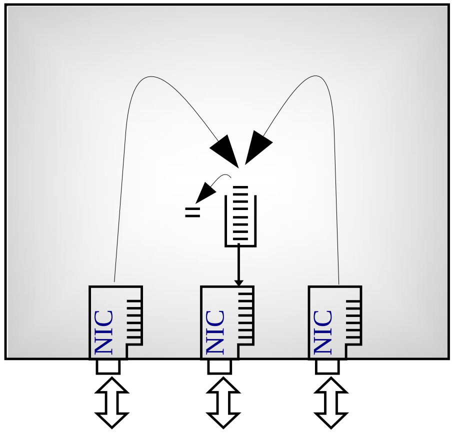
- Internet Protocol (IP) goal is routing datagrams.
- IP main design goal was interconnecting hosts attached to LANs/WANs of different technologies.
- IP was designed independently of any existing network technology.
- IP characteristics are:
  - Connectionless
  - Stateless
  - Best effort



Commercial routers



Basic router architecture



Looses may occur due to buffer overflow

Unit 2: IP Networks

High Performance Routers

Source: Juniper

"There is a major upgrade going on in service providers upgrading their core networks," Chris Komatas, director of service provider marketing at Juniper, said.

"The next-generation core network is all about having the agility to support any service. T1600 is delivering No. 1 in scale, No. 1 in service control and No. 1 in efficiency. All the metrics that are important for a service provider."

The keys to the performance throughput on the Juniper T1600 are the 100Gbps-capable slots that can support all the major connectivity options that carriers may have. Among them is support for OC-768 (40 Gbps), OC-192 (10Gbps) and 10GbE (10 Gigabit Ethernet).

Juniper (www.juniper.net)

Figure 1. Cisco XR 12000 and 12000 series routing portfolio

Table 1. Product Specifications

Product Specification	Cisco XR 12000 and 12000 Series 16-Slot Chassis	Cisco XR 12000 and 12000 Series 10-Slot Chassis	Cisco XR 12000 and 12000 Series 6-Slot Chassis	Cisco XR 12000 and 12000 Series 4-Slot Chassis
Slot capacity	16 slots	10 slots	6 slots	4 slots
Aggregate switching capacity	Cisco 12016: 80 Gbps Cisco 12416: 320 Gbps Cisco 12816: 1280 Gbps	Cisco 12010: 50 Gbps Cisco 12410: 200 Gbps Cisco12810: 800 Gbps	Cisco 12006: 30 Gbps Cisco 12406: 120 Gbps	Cisco 12404: 80 Gbps
Full-duplex throughput per slot	Cisco 12016: 2.5 Gbps/slot Cisco 12416: 10 Gbps/slot Cisco 12816: 40 Gbps/slot	Cisco 12010: 2.5 Gbps/slot Cisco 12410: 10 Gbps/slot Cisco 12810: 40 Gbps/slot	Cisco 12006: 2.5 Gbps/slot Cisco 12406: 10 Gbps/slot	Cisco 12404: 10 Gbps/slot

cisco (www.cisco.com)

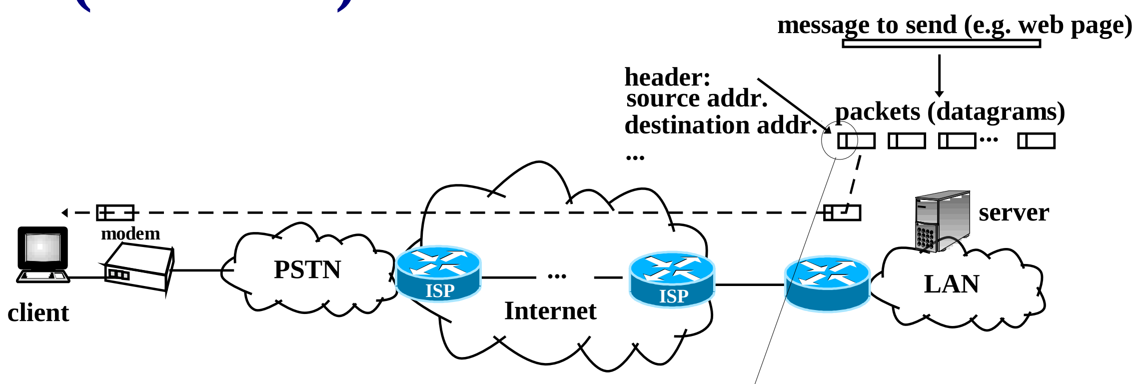
Unit 2: IP Networks

Outline

- IP layer service
- **IP addresses**
- Subnetting
- Routing tables
- ARP protocol
- IP header
- ICMP protocol
- DHCP protocol
- NAT
- DNS
- Routing algorithms
- Security in IP

Unit 2: IP Networks

IP Addresses (RFC 791)



Datagram packet switching

```
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 bits
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Version|  IHL  |Type of Service|          Total Length         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Identification        |Flags|      Fragment Offset    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Time to Live |    Protocol   |         Header Checksum       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Source Address                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Destination Address                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Options                    |    Padding    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

IP datagram header

Unit 2: IP Networks

IP Addresses

```
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 bits
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          netid                /              hostid            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

- 32 bits (4 bytes).
- Dotted point notation: Four bytes in decimal, e.g. 147.83.24.28
- netid identifies the network.
- hostid identifies the host within the network.
- An IP address identifies an *interface*: an attachment point to the network.
- All IP addresses in Internet must be different. To achieve this goal, Internet Assigned Numbers Authority, IANA (http://www.iana.net) assign address blocs to Regional Internet Registries, RIR:
  - RIPE: Europe, http://www.ripe.net.
  - ARIN: USA, http://www.arin.net.
  - APNIC: ASIA http://www.apnic.net.
  - LACNIC: Latin America, http://www.lacnic.net.
  - AFRINIC: Africa, http://www.afrinic.net.
- RIR assign addresses to ISPs, and ISPs to their customers.

Unit 2: IP Networks

IP Addresses - Classes

- The highest bits identify the class.
- The size of netid/hostid varies in classes A/B/C.
- D Class is for multicast addresses (e.g. 224.0.0.2: "all routers")
- E Class are reserved addresses.

Classe	netid (bytes)	hostid (bytes)	Codification	range
A	1	3	0xxxx···x	0.0.0.0 ∼ 127.255.255.255
B	2	2	10xxx···x	128.0.0.0 ∼ 191.255.255.255
C	3	1	110xx···x	192.0.0.0 ∼ 223.255.255.255
D	-	-	1110x···x	224.0.0.0 ∼ 239.255.255.255
E	-	-	1111x···x	240.0.0.0 ∼ 255.255.255.255

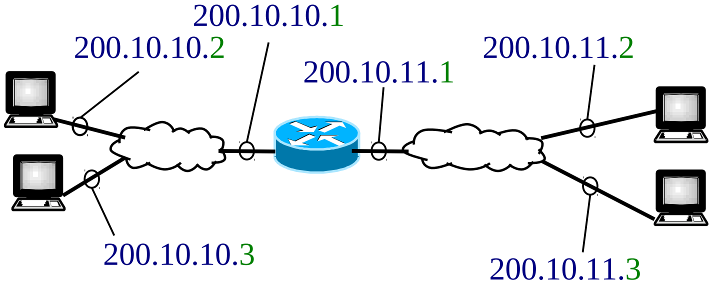
Unit 2: IP Networks

IP Addresses – Special Addresses

- Special addresses cannot be used for a physical interface.
- Each network has two special addresses: network and broadcast addresses.

netid	hostid	Meaning
xxx	all ‘0’	Identifies a network. It is used in routing tables.
xxx	all ‘1’	Broadcast in the net. xxx.
all ‘0’	all ’0’	Identifies “this host” in “this net.”. Used as source address in configuration protocols, e.g. DHCP.
all ‘1’	all ‘1’	broadcast in “this net.”. Used as destination address in configuration protocols, e.g. DHCP.
127	xxx	host loopback: interprocess communication with TCP/IP.

- Example:

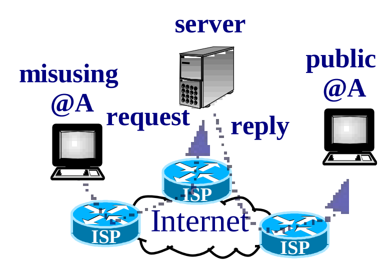


Unit 2: IP Networks

IP Addresses – Private Addresses (RFC 1918)

- Most commercial OSs include the TCP/IP stack.
- TCP/IP is used to network many kind of electronic devices:

- Addresses assigned to RIRs by IANA are called *public*, *global* or *registered*.
- What if we arbitrarily assign a registered address to a host?
  - It may be filtered by our ISP or cause trouble to the right host using that address.



- **Private addresses** has been reserved for devices not using public addresses. These addresses are not assigned to any RIR (are not unique). There are addresses in each class:
  - 1 class A network: 10.0.0.0
  - 16 class B networks: 172.16.0.0 ~ 172.31.0.0
  - 256 class C networks: 192.168.0.0 ~ 192.168.255.0

Unit 2: IP Networks

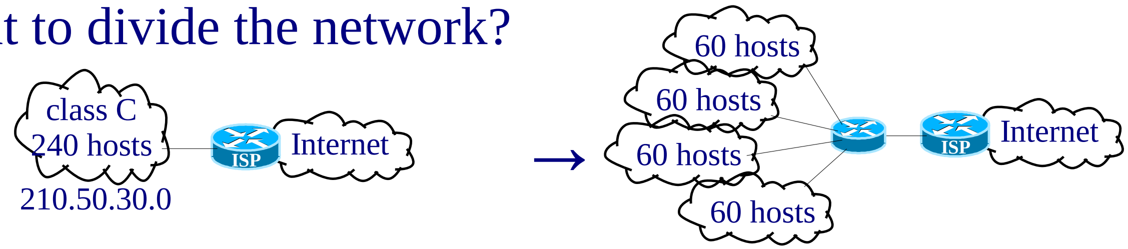
Outline

- IP layer service
- IP addresses
- **Subnetting**
- Routing tables
- ARP protocol
- IP header
- ICMP protocol
- DHCP protocol
- NAT
- DNS
- Routing algorithms
- Security in IP

Unit 2: IP Networks

Subnetting (RFC 950)

- Initially the netid was given by the address class: A with 2^{24} addresses, B with 2^{16} addresses and C with 2^{8} addresses.
- What if we want to divide the network?



- Subnetting allows adding bits from the hostid to the netid (called subnetid bits).
- Example: For the ISP the network prefix is 24 bits. For the internal router the network prefix is 26 bits. The 2 extra bits allows 4 “subnetworks”.
- A **mask** is used to identify the size of the netid+subnetid prefix.
- Mask notations:
  - dotted, as 255.255.255.192
  - giving the number of bits as 210.50.30.0/26

Unit 2: IP Networks

IP Addresses – Subnetting Example

- We want to subnet the address 210.50.30.0/24 in 4 subnets

B = 210.50.30

subnet	subnetid	IP net. addr.	range	broadcast	available
S1	00	B.0/26	B.0 $\sim$ B.63	B.63	$2^6 - 2 = 62$
S2	01	B.64/26	B.64 $\sim$ B.127	B.127	$2^6 - 2 = 62$
S3	10	B.128/26	B.128 $\sim$ B.191	B.191	$2^6 - 2 = 62$
S4	11	B.192/26	B.192 $\sim$ B.255	B.255	$2^6 - 2 = 62$

Unit 2: IP Networks

IP Addresses – Variable Length Subnet Mask (VLSM)

- Subnetworks of different sizes.
- Example, subnetting a class C address:
  - We have 1 byte for subnetid + hostid.
  - subnetid is green, chosen subnets addresses are underlined.

<u>0000</u>
1000 → <u>1000</u>
1100 → <u>1100</u>
<u>1101</u>
<u>1110</u>
<u>1111</u>

subnet	subnetid	IP net. addr.	range	broadcast	available
S1	0	B.0/25	B.0 ∼ B.127	B.127	$2^7 - 2 = 126$
S2	10	B.128/26	B.128 ∼ B.191	B.191	$2^6 - 2 = 62$
S3	1100	B.192/28	B.192 ∼ B.207	B.207	$2^4 - 2 = 14$
S4	1101	B.208/28	B.208 ∼ B.223	B.223	$2^4 - 2 = 14$
S5	1110	B.224/28	B.224 ∼ B.239	B.239	$2^4 - 2 = 14$
S6	1111	B.240/28	B.240 ∼ B.255	B.255	$2^4 - 2 = 14$

Unit 2: IP Networks

IP Addresses – Classless Inter-Domain Routing, CIDR (RFC 1519)

- Initially, Internet backbone routing tables did not use masks: netid was derived from the IP address class.
- When the number of networks in Internet started growing exponentially, routing tables size started exploding.
- In order to reduce routing tables size, CIDR proposed a “rational” geographical-based distribution of IP addresses to be able to “summarize routes”, and use masks instead of classes.
- Summarization example:

200.1.10.0/24
200.1.11.0/24 → 200.1.10.0/23

Unit 2: IP Networks

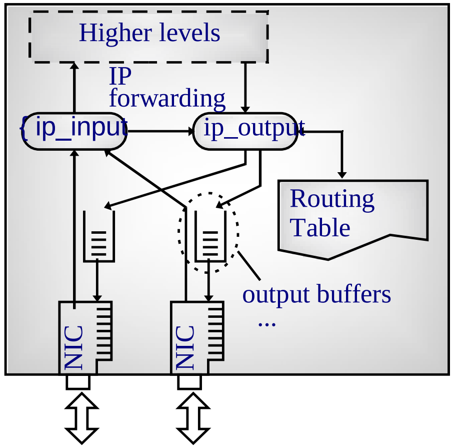
Outline

- IP layer service
- IP addresses
- Subnetting
- **Routing tables**
- ARP protocol
- IP header
- ICMP protocol
- DHCP protocol
- NAT
- DNS
- Routing algorithms
- Security in IP

Unit 2: IP Networks

Routing Table

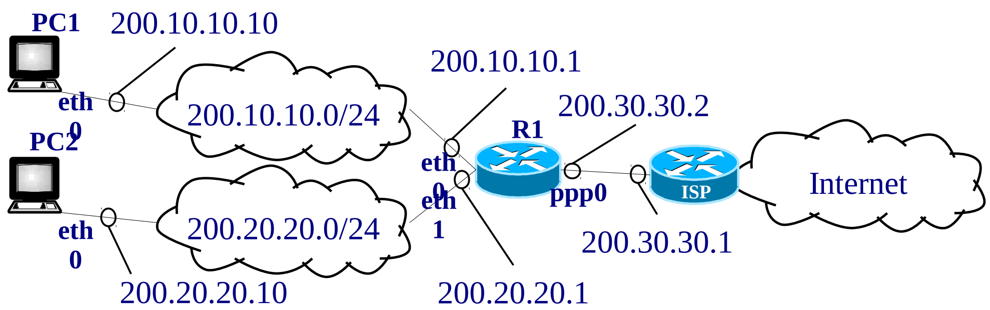
- ip_output() kernel function consults the routing table for each datagram.
- Routing can be:
  - Direct: The destination is directly connected to an interface.
  - Indirect: Otherwise. In this case, the datagram is sent to a router.
- Default route: Is an entry where to send all datagrams with a destination address to a network not present in the routing table. The default route address is 0.0.0.0/0.
- Hosts usually have two entries: The network where they are connected and a default route.



Basic router architecture

Unit 2: IP Networks

Routing Table – Unix Example



known destinations

how to reach the destinations

PC1 routing table:

Destination	**Genmask**	**Gateway**	**Iface**
200.10.10.0	255.255.255.0	0.0.0.0	eth0
0.0.0.0	0.0.0.0	200.10.10.1	eth0

PC2 routing table:

Destination	**Genmask**	**Gateway**	**Iface**
200.20.20.0	255.255.255.0	0.0.0.0	eth0
0.0.0.0	0.0.0.0	200.20.20.1	eth0

R1 routing table:

Destination	**Genmask**	**Gateway**	**Iface**
200.10.10.0	255.255.255.0	0.0.0.0	eth0
200.20.20.0	255.255.255.0	0.0.0.0	eth1
0.0.0.0	0.0.0.0	200.30.30.1	ppp0

Unit 2: IP Networks

Routing Table – Tiscali ISP, CISCO 7200 Router

- Telnet to route-server.ip.tiscali.net (see http://www.bgp4.net server list)

```
+-------------------------------------------------------------------+
|          TISCALI International Network - Route Monitor            |
|                            (AS3257)                               |
|   This system is solely for internet operational purposes. Any    |
|   misuse is strictly prohibited. All connections to this router   |
|   are logged.                                                     |
|   This server provides a view on the TISCALI routing table that   |
|   is used in Frankfurt/Germany. If you are interested in other    |
|   regions of the backbone check out http://www.ip.tiscali.net/lg  |
|   Please report problems to noc@tiscali.net                       |
+-------------------------------------------------------------------+
route-server.ip.tiscali.net>sh ip ro
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is 213.200.64.93 to network 0.0.0.0
B       85.27.76.0/22 [20/10] via 213.200.64.93, 4w2d
B       85.196.154.0/24 [20/10] via 213.200.64.93, 1d09h
B       85.158.216.0/21 [20/10] via 213.200.64.93, 2w6d
B       85.193.136.0/22 [20/10] via 213.200.64.93, 3d08h
B       85.121.48.0/21 [20/0] via 213.200.64.93, 1w4d
B       85.187.201.0/24 [20/10] via 213.200.64.93, 4d19h
B       85.114.0.0/20 [20/10] via 213.200.64.93, 1w5d
B       85.119.16.0/24 [20/10] via 213.200.64.93, 4w0d
B       85.119.16.0/21 [20/10] via 213.200.64.93, 4w0d
B       85.105.0.0/17 [20/10] via 213.200.64.93, 4w2d
B       85.93.52.0/24 [20/10] via 213.200.64.93, 4w0d
...
```

Tiscali Network Map
http://www.tiscali.net

thousands of entries

Unit 2: IP Networks

Routing Table – Datagram Delivery Algorithm

- 1. Check if it is the destination:

```
if(Datagram Destination == address of any of the interfaces) {
    send the datagram to upper layers
}
```

- 2. Consult the routing table:

```
for each routing table entry ordered from longest to shortest mask
  (Longest Prefix Match) {
    if((Datagram Destination IP address & mask) == Destination
      table entry) {
        return (gateway, interface) ;
    }
```

- 3. Forward the datagram

```
if(it is a direct routing) {
    send the datagram to the Datagram Destination IP address
} else { /* it is an indirect routing */
    send the datagram to the gateway IP address
}
```

Unit 2: IP Networks

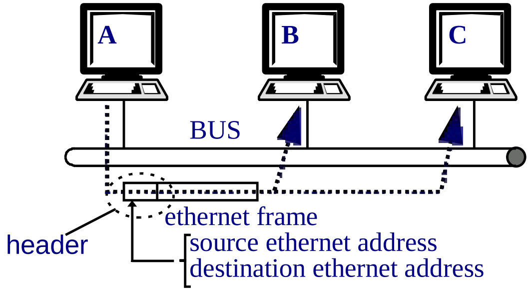
Outline

- IP layer service
- IP addresses
- Subnetting
- Routing tables
- **ARP protocol**
- IP header
- ICMP protocol
- DHCP protocol
- NAT
- DNS
- Routing algorithms
- Security in IP

Unit 2: IP Networks

Address Resolution Protocol, ARP (RFC 826)

- To send the datagram, IP layer may have to pass a "physical address" to the NIC driver.
- ARP translate IP addresses to "physical addresses" (used by the physical network).
- If needed, IP calls ARP module to obtain the "physical addresses" before the NIC driver call.
- Physical addresses are also called MAC or hardware addresses.
- Ethernet example:



Unit 2: IP Networks

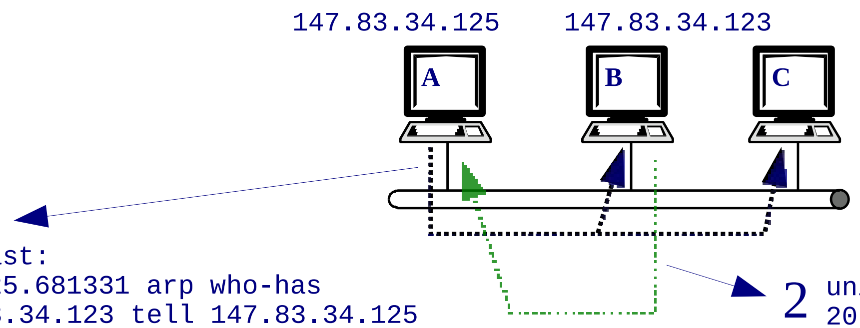
Address Resolution Protocol, messages

- When IP calls ARP:
  - If ARP table has the requested address, it is returned,
  - otherwise, a discovery mechanism is initiated:
    - IP stores the datagram in a temporal buffer, initiates a timeout and starts forwarding the next datagram in the transmission queue.
    - If the timeout triggers, the datagram is removed.
    - If ARP returns the requested address, IP calls the driver with it.
- ARP discovery protocol in an ethernet network (broadcast network):
  - An “ARP Request” broadcast message is sent indicating the IP address.
  - The station having the requested IP address send an “ARP Reply”, and stores the requesting address in the ARP table.
  - Upon receiving the “ARP Reply”, the requesting station return the IP call with it.
  - ARP entries have a timeout refreshed each time a match occurs.

Unit 2: IP Networks

Address Resolution Protocol, messages - Example

- Messages (tcpdump):



1 broadcast:
```
20:02:25.681331 arp who-has
 147.83.34.123 tell 147.83.34.125
```

2 unicast:
```
20:02:25.681490 arp reply 147.83.34.123
is-at 00:c0:49:d5:96:d8
```

- ARP tables:

```
A> /sbin/arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
147.83.34.123            ether   00:c0:49:d5:96:d8   C                     eth0

B> /sbin/arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
147.83.34.125            ether   00:14:F1:CC:59:00   C                     eth0
```

"Completed" flag

Unit 2: IP Networks

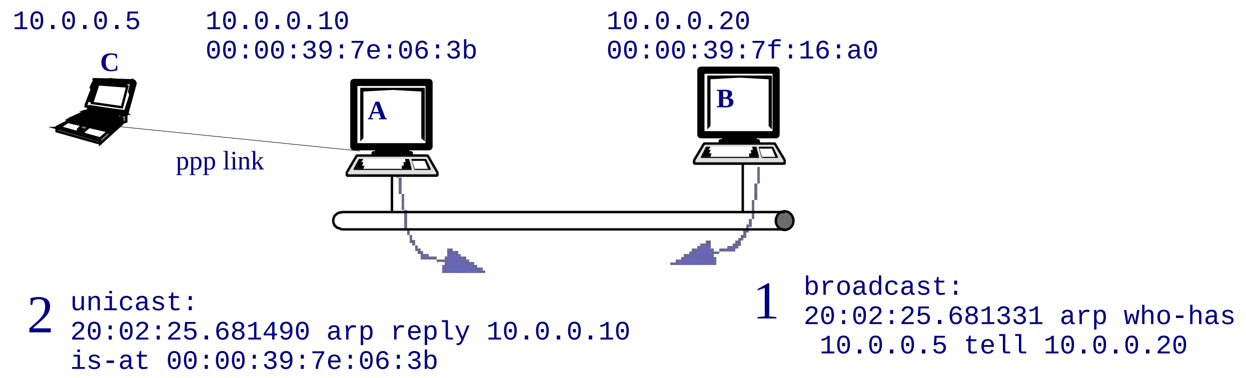
Address Resolution Protocol – Message format (ethernet)

- ARP messages are encapsulated directly in a data-link frame.

```
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 bits
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Hardware Type (16)            |  Protocol Type (16)           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Hard. Length(8)|Prot. Length(8)|  Opcode (16)                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|       Sender Hardware                                         |
+       Address (48)            +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                               | Sender Protocol Address (32)  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Sender Protocol Address (cont)|     Target Hardware           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+     Address (48)              +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|       Target Protocol Address (32)                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Unit 2: IP Networks

Address Resolution Protocol – Proxy ARP



- ARP tables:

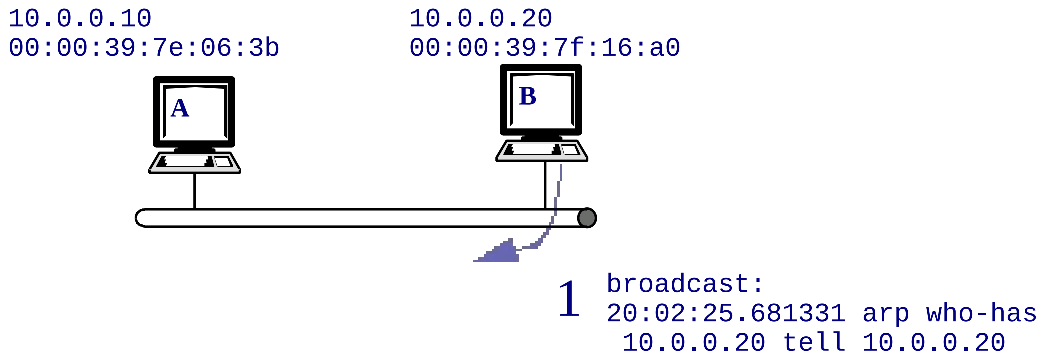
```
A # /sbin/arp -i eth0 -s 10.0.0.5 00:00:39:7e:06:3b pub
A # /sbin/arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
10.0.0.20                ether   00:00:39:7f:16:a0   C                     eth0
10.0.0.5                 ether   00:00:39:7e:06:3b   MP                    eth0
                                                        “Completed” flag
                                                       “Manual” and “Permanent” flags
B # /sbin/arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
10.0.0.5                 ether   00:00:39:7e:06:3b   C                     eth0
```

- Routing table of host A:

```
A # route -n
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
10.0.0.5        0.0.0.0         255.255.255.255 U     0      0        0 ppp0
10.0.0.0        0.0.0.0         255.255.255.0   U     0      0        0 eth0
```

Unit 2: IP Networks

Address Resolution Protocol – Gratuitous ARP



- Goals:
  - Detect duplicated IP addresses.
  - Update ARP table MAC addresses after an IP or NIC change.

Unit 2: IP Networks

Outline

- IP layer service
- IP addresses
- Subnetting
- Routing tables
- ARP protocol
- **IP header**
- ICMP protocol
- DHCP protocol
- NAT
- DNS
- Routing algorithms
- Security in IP

Unit 2: IP Networks

IP Header (RFC 791)

- Version: 4
- IP Header Length (IHL): Header size in 32 bit words.
- Type of Service: (ToS): *xxxdtrc0.*
- Total Length: Datagram size in bytes.
- Identification/Flags/Fragment Offset: used in fragmentation.
- Time to Live (TTL): if(--TTL==0) { discard ; }.
- Protocol: Encapsulated protocol (/etc/protocols in unix).
- Header Checksum: Header error detection.
- Source and Destination Addresses: End nodes addresses.
- Options: Rcord Route, Loose Source Routing, Strict Source Routing.

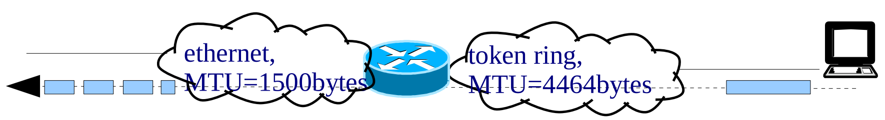
20 bytes

```
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 bits
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Version|  IHL  |Type of Service|          Total Length         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Identification        |Flags|      Fragment Offset    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Time to Live |    Protocol   |         Header Checksum       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Source Address                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Destination Address                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Options                    |    Padding    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Unit 2: IP Networks

IP Fragmentation

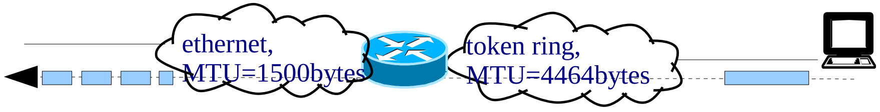
- Router: Fragmentation may be needed when two networks with different *Maximum Transfer Unit* (MTU) are connected.
- Host: Fragmentation may be needed using UDP. TCP segments are ≤ MTU.
- Datagrams are reconstructed at the destination.
- Fields:
  - Identification (16 bits): identify fragments from the same datagram.
  - Flags (3 bits):
    - D, don't fragment. Used in MTU path discovery
    - M, More fragments: Set to 0 only in the last fragment
  - Offset (13 bits): Position of the fragment first byte in the original datagram in 8 byte words (indexed at 0).

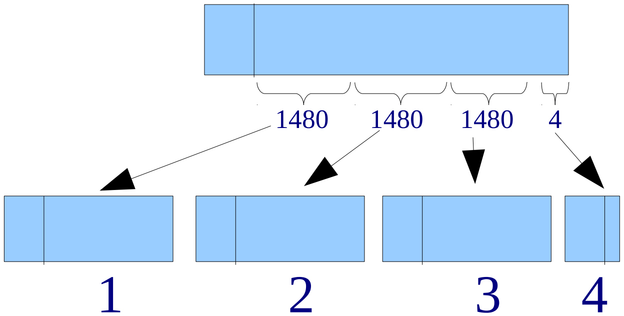


Unit 2: IP Networks

IP Fragmentation - Example

- Original datagram = 4464 bytes (4Mbps Token Ring): 20 header + 4444 payload.
- Fragment size = $\left\lfloor \frac{1500-20}{8} \right\rfloor = 185$ 8-byte-words (1480 bytes)
  - 1^{st} fragment: offset = 0 , M = 1. 0~1479 payload bytes.
  - 2^{nd} fragment: offset = 185, M = 1. 1480~2959 payload bytes.
  - 3^{rd} fragment: offset = 370, M = 1 . 2960~4439 payload bytes.
  - 4^{th} fragment: offset = 555, M = 0 . 4440~4443 payload bytes.

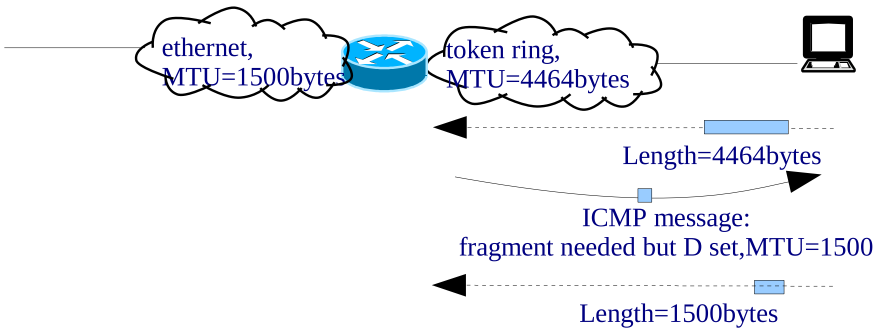




Unit 2: IP Networks

MTU Path Discovery

- Used in modern TCP implementations.
- TCP by default choses the maximum segment size, to avoid headers overhead (TCP payload / (TCP payload + Σ TCP,IP,Data-link,Physical headers)
- Goal: avoid fragmentation:



Unit 2: IP Networks

Outline

- IP layer service
- IP addresses
- Subnetting
- Routing tables
- ARP protocol
- IP header
- **ICMP protocol**
- DHCP protocol
- NAT
- DNS
- Routing algorithms
- Security in IP

Unit 3: IP Networks

Internet Control Message Protocol, ICMP (RFC 792)

- Used for attention and error messages.
- Can be generated by IP, TCP/UDP, and application layers.
- Are encapsulated into an IP datagram.
- Can be: (i) query, (ii) error.
- An ICMP error message cannot generate another ICMP error message (to avoid loops).

Unit 3: IP Networks

ICMP general format message (RFC 792)

```
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 bits
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |     Code      |          Checksum             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    contingut (variable)                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

Identifies the message                        Is computed using all the message
```

- Query type messages have an identifier field, for request-reply correspondence.
- Error messages have a field where the first 8 bytes of the datagram payload causing the error are copied. These bytes capture the TCP/UDP ports. E.g. Destination Unreachable Message:

```
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |     Code      |          Checksum             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             unused                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      Internet Header + 64 bits of Original Data Datagram      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Unit 2: IP Networks

Common ICMP messages

Type	Code	query/error	Name	Description
0	0	query	echo reply	Reply an echo request
3	0	error	network unreachable	Network not in the RT.
	1	error	host unreachable	ARP cannot solve the address.
	2	error	protocol unreachable	IP cannot deliver the payload
	3	error	port unreachable	TCP/UDP cannot deliver the payload
	4	error	fragmentation needed but DF set	MTU path discovery
4	0	error	source quench	Sent by a congested router.
5	0	error	redirect for network	When the router send a datagram by the same interface it was received.
8	0	query	echo request	Request for reply
11	0	error	TTL=0 during transit	Sent by a router when `--TTL=0`

Unit 2: IP Networks

Outline

- IP layer service
- IP addresses
- Subnetting
- Routing tables
- ARP protocol
- IP header
- ICMP protocol
- **DHCP protocol**
- NAT
- DNS
- Routing algorithms
- Security in IP

Unit 3: IP Networks

Dynamic Host Configuration Protocol, DHCP (RFC 2131)

- Improves and can interoperate with previous BOOTP protocol.
- Used for automatic network configuration:
  - Assign IP address and mask,
  - Default route,
  - Hostname,
  - DNS domain,
  - Configure DNS servers,
  - etc.
- IP address configuration can be:
  - Dynamic: During a leasing time.
  - Automatic: Unlimited leasing time.
  - Manual: IP addresses are assigned to specific MAC addresses.

Unit 3: IP Networks

DHCP – Protocol Messages (RFC 2131)

```
DHCPDISCOVER -  Client broadcast to locate available servers.

DHCPOFFER    -  Server to client in response to DHCPDISCOVER with
                offer of configuration parameters.

DHCPREQUEST  -  Client message to servers either (a) requesting
                offered parameters from one server and implicitly
                declining offers from all others, (b) confirming
                correctness of previously allocated address after,
                e.g., system reboot, or (c) extending the lease on a
                particular network address.

DHCPACK      -  Server to client with configuration parameters,
                including committed network address.

DHCPNAK      -  Server to client indicating client's notion of network
                address is incorrect (e.g., client has moved to new
                subnet) or client's lease as expired

DHCPDECLINE  -  Client to server indicating network address is already
                in use.

DHCPRELEASE  -  Client to server relinquishing network address and
                cancelling remaining lease.

DHCPINFORM   -  Client to server, asking only for local configuration
                parameters; client already has externally configured
                network address.
```

Unit 3: IP Networks

DHCP – Message Fields (RFC 2131)

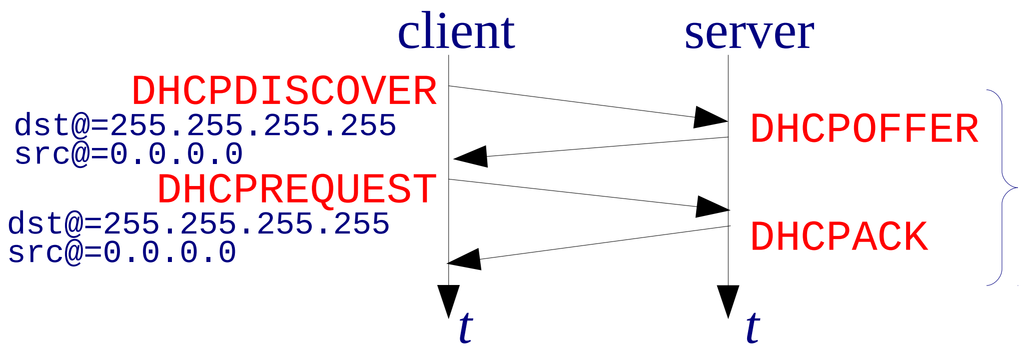
```
FIELD      OCTETS       DESCRIPTION
op            1  Message op code / message type. 1 = BOOTREQUEST, 2 = BOOTREPLY.
htype         1  Hardware address type.
hlen          1  Hardware address length.
hops          1  Client sets to zero, optionally used by relay agents
                 when booting via a relay agent.
xid           4  Transaction ID, a random number chosen by the
                 client, used by the client and server to associate
                 messages and responses between a client and a
                 server.
secs          2  Filled in by client, seconds elapsed since client
                 began address acquisition or renewal process.
flags         2  Flags.
ciaddr        4  Client IP address; only filled in if client is in
                 BOUND, RENEW or REBINDING state and can respond
                 to ARP requests.
yiaddr        4  'your' (client) IP address. Set by the server in
                 a DHCPOFFER message.
siaddr        4  IP address of next server to use in bootstrap;
                 returned in DHCPOFFER, DHCPACK by server.
giaddr        4  Relay agent IP address, used in booting via a
                 relay agent.
chaddr       16  Client hardware address.
sname        64  Optional server host name, null terminated string.
file        128  Boot file name, null terminated string; "generic"
                 name or null in DHCPDISCOVER, fully qualified
                 directory-path name in DHCPOFFER.
options     var  Optional parameters field.
```

```
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     op (1)    |   htype (1)   |   hlen (1)    |   hops (1)    |
+---------------+---------------+---------------+---------------+
|                            xid (4)                            |
+-------------------------------+-------------------------------+
|           secs (2)            |           flags (2)           |
+-------------------------------+-------------------------------+
|                          ciaddr  (4)                          |
+---------------------------------------------------------------+
|                          yiaddr  (4)                          |
+---------------------------------------------------------------+
|                          siaddr  (4)                          |
+---------------------------------------------------------------+
|                          giaddr  (4)                          |
+---------------------------------------------------------------+
|                                                               |
|                          chaddr  (16)                         |
|                                                               |
|                                                               |
+---------------------------------------------------------------+
|                                                               |
|                          sname   (64)                         |
+---------------------------------------------------------------+
|                                                               |
|                          file    (128)                        |
+---------------------------------------------------------------+
|                                                               |
|                          options (variable)                   |
+---------------------------------------------------------------+
```

Unit 3: IP Networks

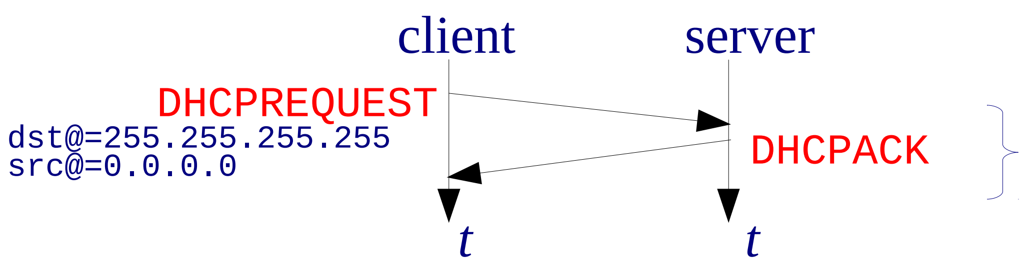
DHCP – Client-server interaction (RFC 2131)

- UDP, server port = 67, client port = 68.



Can be unicast or broadcast, if requested by the client.

- The client can directly send `DHCPREQUEST`:
  - After rebooting if it remembers and wishes to reuse a previously allocated network address.
  - Extending the lease on a particular network address.



Can be unicast or broadcast, if requested by the client.

DHCP – Example: tcpdump/dhcpdump capture

```
linux # tcpdump -lenx -s 1500 -i eth0 port bootps or port bootpc | dhcpdump
  TIME: 17:09:24.616312
    IP: 0.0.0.0.68 (00:30:1b:b4:6d:78) > 255.255.255.255.67 (ff:ff:ff:ff:ff:ff)
    OP: 1 (BOOTPREQUEST)
 HTYPE: 1 (Ethernet)
   XID: 181f0139
 FLAGS: 0
CIADDR: 0.0.0.0
YIADDR: 0.0.0.0
SIADDR: 0.0.0.0
GIADDR: 0.0.0.0
CHADDR: 00:30:1b:b4:6d:78:00:00:00:00:00:00:00:00:00:00
OPTION:  53 (  1) DHCP message type         3 (DHCPREQUEST)
OPTION:  57 (  2) Maximum DHCP message size 576
OPTION:  50 (  4) Request IP address        192.168.1.100
OPTION:  51 (  4) IP address leasetime      -1 ()
OPTION:  55 ( 21) Parameter Request List      1 (Subnet mask)
                                              3 (Routers)
                                              6 (DNS server)
                                             12 (Host name)
                                             15 (Domainname)
                                             23 (Default IP TTL)
                                             28 (Broadcast address)
                                             29 (Perform mask discovery)
                                             42 (NTP servers)
                                              9 (LPR server)
                                            119 (Domain Search)
                                            ...
---------------------------------------------------------------------------
  TIME: 17:09:24.619312
    IP: 192.168.1.1.67 (00:18:39:5d:74:9d) > 192.168.1.100.68 (00:30:1b:b4:6d:78)
    OP: 2 (BOOTPREPLY)
 HTYPE: 1 (Ethernet)
   XID: 181f0139
 FLAGS: 0
CIADDR: 0.0.0.0
YIADDR: 192.168.1.100
SIADDR: 192.168.1.1
GIADDR: 0.0.0.0
CHADDR: 00:30:1b:b4:6d:78:00:00:00:00:00:00:00:00:00:00
OPTION:  53 (  1) DHCP message type         5 (DHCPACK)
OPTION:  54 (  4) Server identifier         192.168.1.1
OPTION:  51 (  4) IP address leasetime      86400 (24h)
OPTION:   1 (  4) Subnet mask               255.255.255.0
OPTION:   3 (  4) Routers                   192.168.1.1
OPTION:   6 (  4) DNS server                192.168.0.1
OPTION:  15 (  3) Domainname                lan
```

Unit 2: IP Networks

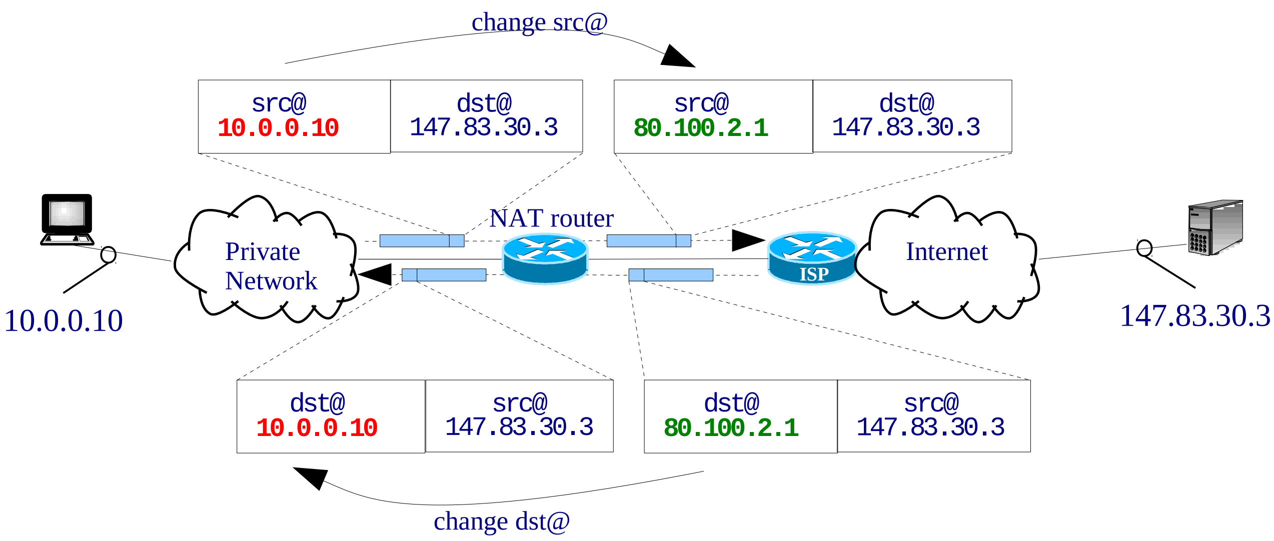
Outline

- IP layer service
- IP addresses
- Subnetting
- Routing tables
- ARP protocol
- IP header
- ICMP protocol
- DHCP protocol
- **NAT**
- DNS
- Routing algorithms
- Security in IP

Unit 3: IP Networks

Network Address Translation, NAT (RFCs 1631, 2663 3022)

- Typical scenario: Private addresses (internal addresses) are translated to public addresses (external addresses).
- A NAT table is used for address mapping.
- Advantages:
  - Save public addresses.
  - Security.
  - Administration, e.g. changing ISP does not imply changing private network addressing.



Unit 3: IP Networks

NAT – Types of translations

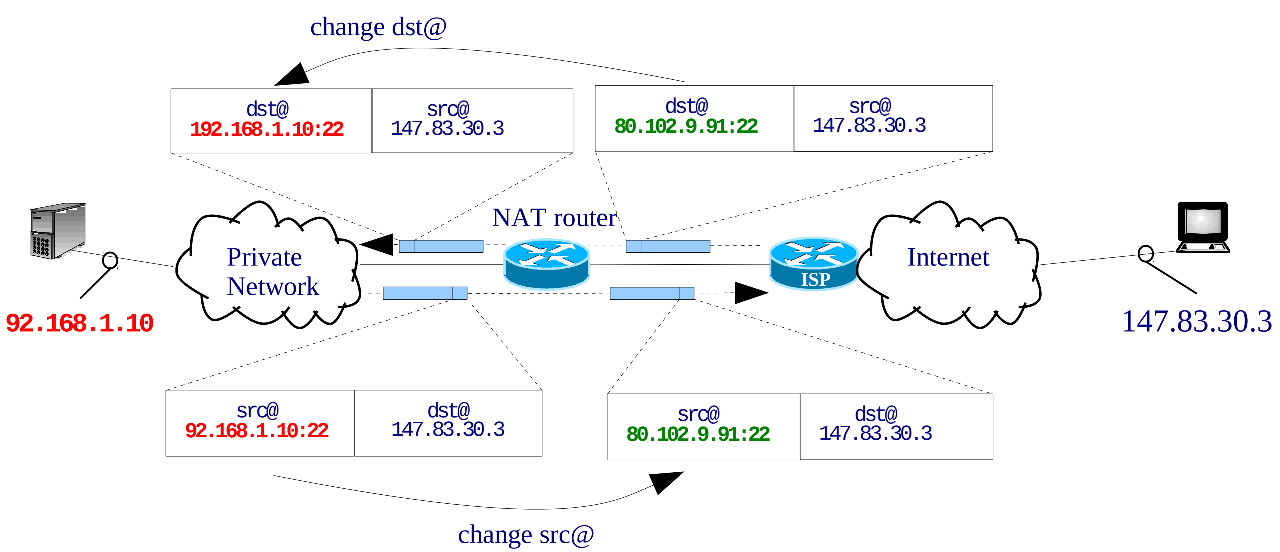
- NOTE: NAT is a technique, not a protocol. Implementations and terminology may change from one manufacturer to another.
- Basic NAT:
  - A different external address is used for each internal address → a different public IP address is needed for each hosts accessing Internet.
  - Each NAT table entry has the tuple: (internal address, external address).
  - Each host requires one NAT table entry.
- Port and Address Translation, PAT:
  - The same external address can be used for each internal address → a unique public IP address can be used for all hosts accessing Internet.
  - Each NAT table entry has the tuple: (int. address/port, ext. address/port)
  - Each connection requires one NAT table entry.
- The NAT table entries can be:
  - Static: Manually added.
  - Dynamic:
    - Entries are automatically added when an internal connection is initiated.
    - External addresses are chosen from a pool.
    - Table entries have a timeout.

Unit 3: IP Networks

DNAT

- What if we want external connections to internal servers? (DNAT in linux-iptables terminology).
- The address translation is exactly the same as NAT, but, the connection is initiated from an external client.
- Typically, some static configuration is needed to configure the server IP/port.

```
Static entry in the NAT router:
Inside-address:Port  Outside-address:Port
192.168.1.10:22     80.102.9.91:22
```



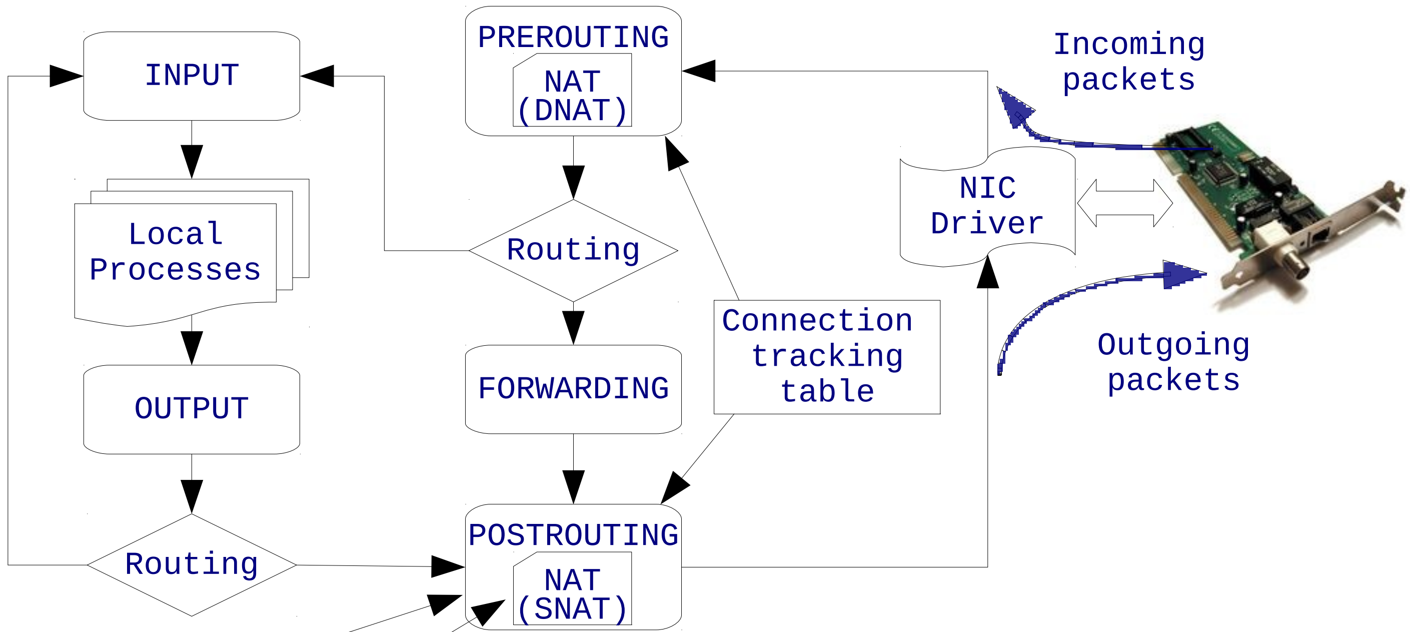
Unit 2: IP Networks

NAT – Linux example

- iptables is used to add NAT/DNAT rules, e.g.

```
iptables -t NAT -A POSTROUTING -j SNAT --to-source 80.102.191.191
```

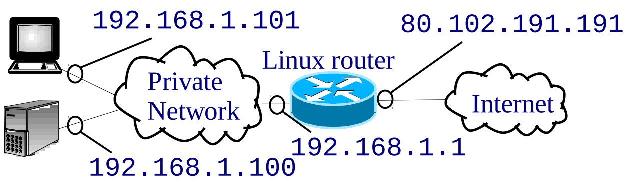
- Information of established connections is recorded by the "connection tracking" module. Connection information is used as "NAT table".



- chains
- tables with routing rules

Linux routing *chains*.

Unit 2: IP Networks



NAT – Linux example

- NAT outgoing packets to 80.102.191.191

```
iptables -t NAT -A POSTROUTING -j SNAT --to-source 80.102.191.191
```

- DNAT incoming packets, port 22 (ssh) to 192.168.1.100

```
iptables -t NAT -A PREROUTING -p tcp –dport 22 -j DNAT --to-destination 192.168.1.100
```

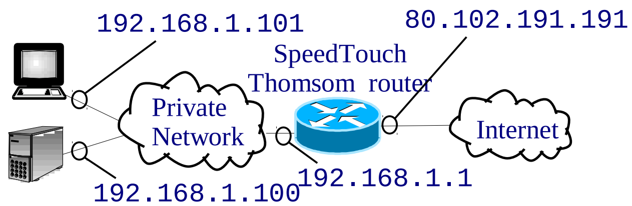
```
        linux-router # cat /proc/net/ip_conntrack
        tcp      6 103 TIME_WAIT src=192.168.1.101 dst=84.120.112.212 sport=1730 dport=1755
                 src=84.120.112.212 dst=80.102.191.191 sport=1755 dport=1730 [ASSURED]
        udp      17 3591 src=192.168.1.101 dst=217.125.101.197 sport=5770 dport=4941
                 src=217.125.101.197 dst=80.102.191.191 sport=4941 dport=5770 [ASSURED]
SNAT    tcp      6 112 SYN_SENT src=192.168.1.101 dst=85.59.94.22 sport=1795 dport=13392 [UNREPLIED]
                 src=85.59.94.22 dst=80.102.191.191 sport=13392 dport=1795
        tcp      6 3598 ESTABLISHED src=192.168.1.101 dst=82.158.227.48 sport=4598 dport=4662
                 src=82.158.227.48 dst=80.102.191.191 sport=4662 dport=4598 [ASSURED]
DNAT    tcp      6 3599 ESTABLISHED src=147.83.30.137 dst=80.102.191.191 sport=1096 dport=22
                 src=192.168.1.100 dst=147.83.30.137 sport=22 dport=1096 [ASSURED]
        ...
```

Legend:
protocol-name, protocol-number, timeout(seconds), [tcp-state], received IP/port src/dst, expected return IP/port src/dst, [UNREPLIED: first packet|ASSURED: packets in both directions]

Unit 3: IP Networks

NAT – ADSL commercial router example

- NAT outgoing packets to 80.102.191.191
- DNAT incoming packets, port 22 (ssh) to 192.168.1.100




```
      linux # telnet 192.168.1.1
      Trying 192.168.0.1...
      Connected to 192.168.1.1.
      =>nat
      [nat]=>list
      Indx Prot Inside-address:Port  Outside-address:Port  Foreign-address:Port Flgs  Expir  State  Control
DNAT    2   6   192.168.1.100:22      80.102.191.191:22          0.0.0.0:0      instance
        6   6   192.168.1.101:1420    80.102.191.191:10079   83.60.122.22:45730   1   14m48      1
       11   6   192.168.1.101:1337    80.102.191.191:10060  85.56.136.231:16000   1   14m30      1
SNAT   12   6   192.168.1.101:1402    80.102.191.191:10064   82.159.8.187:1755    1     14s      5
       ...
```

Unit 2: IP Networks

Outline

- IP layer service
- IP addresses
- Subnetting
- Routing tables
- ARP protocol
- IP header
- ICMP protocol
- DHCP protocol
- NAT
- **DNS**
- Routing algorithms
- Security in IP

Unit 2. Network applications

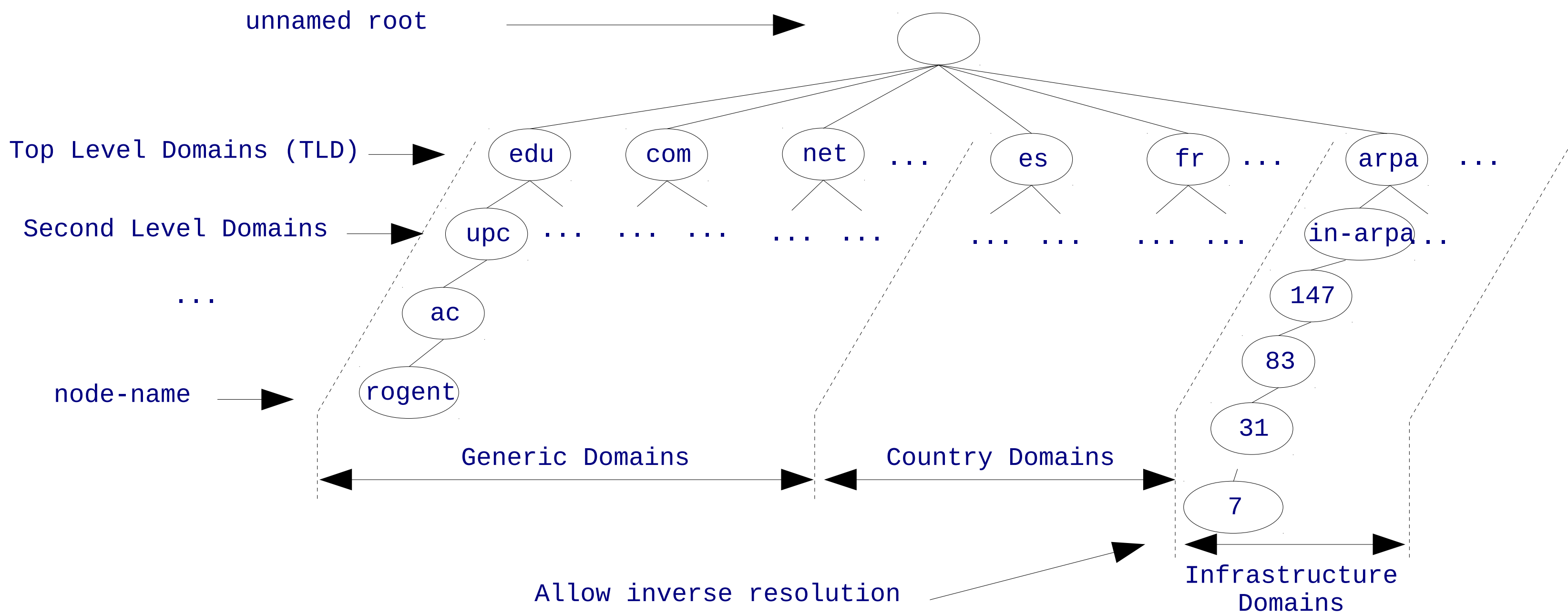
Domain Name System DNS (RFC 1034, 1035)

- Allows users to use names instead of IP addresses: e.g. rogent.ac.upc.edu instead of 147.83.31.7, www.upc.edu instead of 147.83.194.21, etc.
- Names consists of a node-name and a domain-mane: rogent.ac.upc.edu, www.upc.edu
- DNS consists of a worldwide distributed data base.
- DNS data base entries are referred to as *Resource Records* (RR).
- The information associated with a name is composed of 1 or more RRs.
- Names are case insensitive (e.g. www.upc.edu and WWW.UPC.EDU are equivalent).

Unit 2. Network applications

DNS – Domain Hierarchy

- DNS data base is organized in a tree:



Unit 2. Network applications

DNS – Domain Hierarchy

- The *Internet Corporation for Assigned Names and Numbers* (ICANN) is responsible for managing and coordinating the DNS.
- ICANN delegates Top Level Domains (TLD) administration to registrars: http://www.internic.net
- Domains delegate the administration of subdomains.

Unit 2. Network applications

DNS – Data Base Organization

- Access to DNS data base is done using *Name Servers* (NS).
- NSs may hold permanent and cached RRs. Cached RRs are removed after a timeout.
- Each subdomain has an *authority* which consists of a primary and backup NSs.
- In this context, subdomains are referred to as *zones*, and delegated subdomains *subzones*.
- An authority has the complete information of a zone:
  - Names and addresses of all nodes within the zone.
  - Names and addresses of all subzone authorities.

Unit 2. Network applications

DNS – Data Base Organization

- Root Servers are the entry point to the domain hierarchy.
- Root Servers are distributed around the world and have the TLD addresses: http://www.root-servers.org
- Root server addresses are needed in a NS configuration.

Source: http://www.root-servers.org

Unit 2. Network applications

DNS - Unix example: The resolver

- The applications use the calls (*resolver* library):

```
struct hostent *gethostbyname(const char *name) ;
struct hostent *gethostbyaddr(const void *addr, int len, int type);
```

- The resolver first looks the /etc/hosts file:

```
# hosts         This file describes a number of hostname-to-address
#               mappings for the TCP/IP subsystem.  It is mostly
#               used at boot time, when no name servers are running.
#               On small systems, this file can be used instead of a
#               "named" name server.
# Syntax:
# IP-Address  Full-Qualified-Hostname  Short-Hostname
127.0.0.1       localhost
10.0.1.1        massanella.ac.upc.edu massanella
```

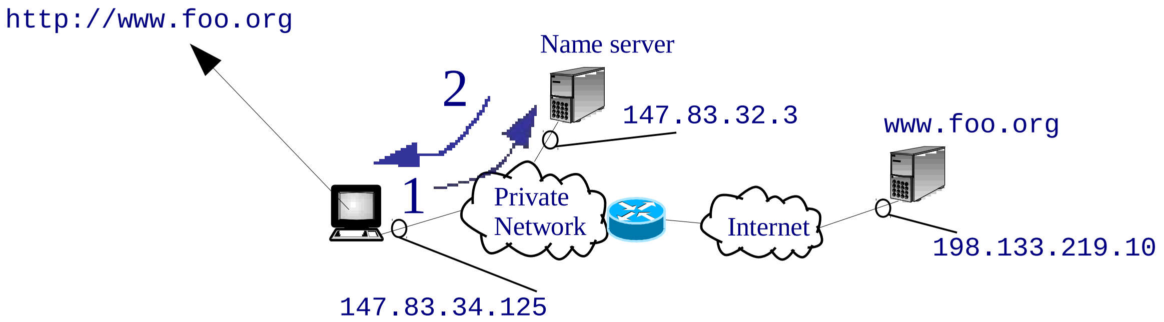
- Otherwise a *name server* is contacted using /etc/resolv.conf file:

```
search ac.upc.edu
nameserver 147.83.32.3
nameserver 147.83.33.4
```

Unit 2. Network applications

DNS - Protocol

- Client-server paradigm
- UDP/TCP. Short messages uses UDP.
- well-known port: 53



1 `18:36:00.322370 IP (proto: UDP) 147.83.34.125.1333 > 147.83.32.3.53: 53040+ A? www.foo.org. (31)`

2 `18:36:00.323080 IP (proto: UDP) 147.83.32.3.53 > 147.83.34.125.1333: 53040 1/2/2 www.foo.org. A 198.133.219.10 (115)`

Unit 2. Network applications

DNS – Unix example: Basic NS configuration

- Unix NS implementation is BIND (Berkeley Internet Name Domain), http://www.isc.org.
- named is the BIND NS daemon.
- BIND basic configuration files:
  - `/etc/named.conf` global configuration
  - `/var/lib/named/root.hint` root servers addresses
  - `/var/lib/named/*.db` zone files

Unit 2. Network applications

DNS – Unix example: zone file

The domain name

The domain NS

The domain maintainer mail address (the @ is written as a '.')

comments

```
linux # cat /var/lib/named/foo.db
; BIND data file for foo.org
; /var/lib/named/foo.db
;
foo.org. IN      SOA     dns.foo.org. root.foo.org. (
                       1998121401          ; Serial
                          604800           ; Refresh
                           86400           ; Retry
                         2419200           ; Expire
                          604800 )         ; Default TTL
         IN      NS      dns.foo.org.
         IN      MX      10      mail.foo.org.
server   IN      A       198.133.219.10
www      IN      CNAME   server
ftp      IN      CNAME   server
news     IN      A       198.133.219.20
mail     IN      A       198.133.219.30
dns      IN      A       198.133.219.40
dns2     IN      A       198.133.219.50
…
sub.foo.org. IN NS       dns3.sub.foo.org.
dns3     IN      A       10.10.0.24
…
```

configuration

MX preference value (used if multiple servers are available)

NS name

domain mail server

IP addresses and alias names

Resource Records (RR)

delegated sub-domain

name (type A or CNAME), domain (type NS of MX). If the domain is missing, it is automatically added.

class:
IN: Internet System.

type:
- SOA: Start Of Authority.
- NS: NS name.
- MX: the domain mail exchange.
- A: A host address.
- CNAME: Canonical Name Record. E.g. the real hostname of www.foo.org is server.foo.org.

address (type A), name (type NS or CNAME)...

Unit 2. Network applications

DNS – Unix example: root servers addresses

```
linux # cat /var/lib/named/root.hint

;       This file holds the information on root name servers needed to
;       initialize cache of Internet domain name servers
;       (e.g. reference this file in the "cache  .  <file>"
;       configuration file of BIND domain name servers).
;
;       This file is made available by InterNIC
;       under anonymous FTP as
;           file                /domain/named.root
;           on server           FTP.INTERNIC.NET
;       -OR-                    RS.INTERNIC.NET
.                        3600000  IN  NS    A.ROOT-SERVERS.NET.
A.ROOT-SERVERS.NET.      3600000  IN  A     198.41.0.4
.                        3600000  IN  NS    B.ROOT-SERVERS.NET.
B.ROOT-SERVERS.NET.      3600000  IN  A     192.228.79.201
.                        3600000  IN  NS    C.ROOT-SERVERS.NET.
C.ROOT-SERVERS.NET.      3600000  IN  A     192.33.4.12

...
.                        3600000  IN  NS    M.ROOT-SERVERS.NET.
M.ROOT-SERVERS.NET.      3600000  IN  A     202.12.27.33
```

comments

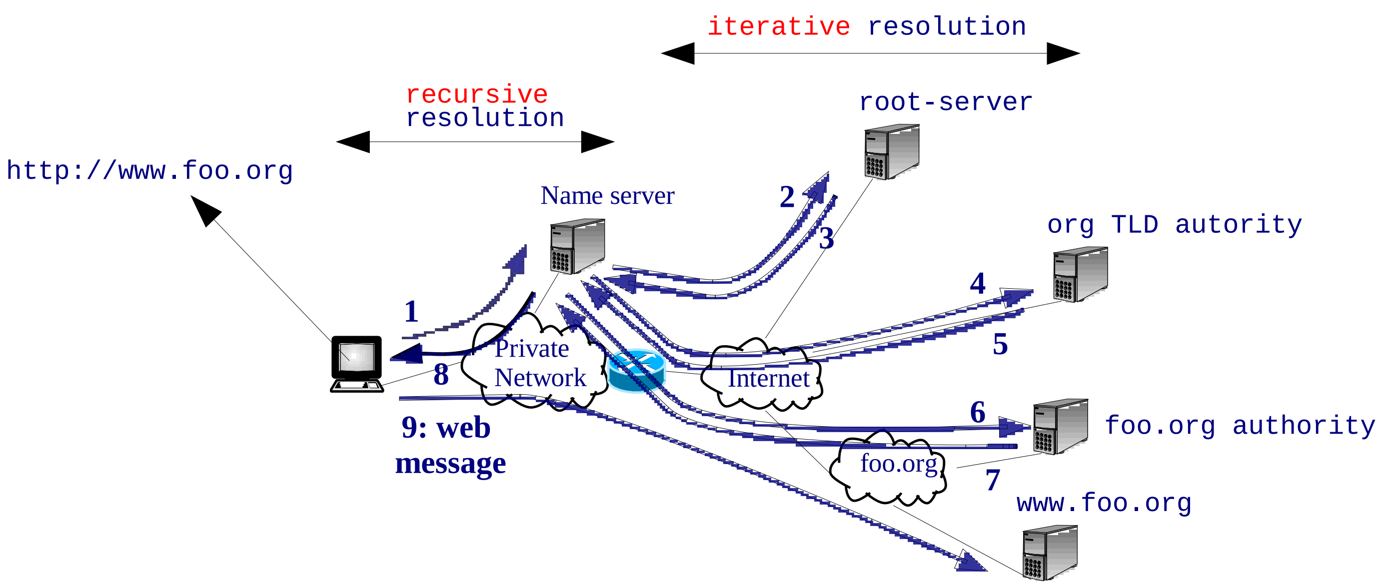
Resource Records (RR)
pointing to root-servers

address of a name
NS name

Unit 2. Network applications

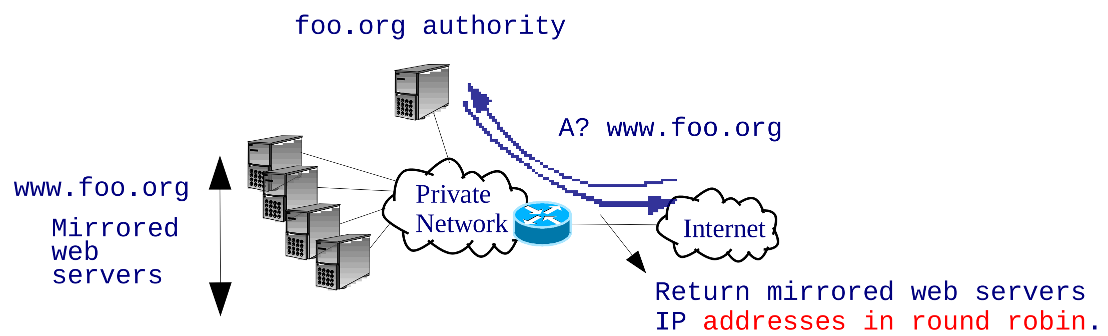
DNS – Resolution

- NSs cache name resolutions.
- A cached RR is returned without looking for in the NS authority.
- The same name may be associated with several IP addresses (e.g. load balancing).
- The addresses of a common domain may not belong to the same IP network (e.g. Content Distribution Networks).



Unit 2. Network applications

DNS – Load balancing, example



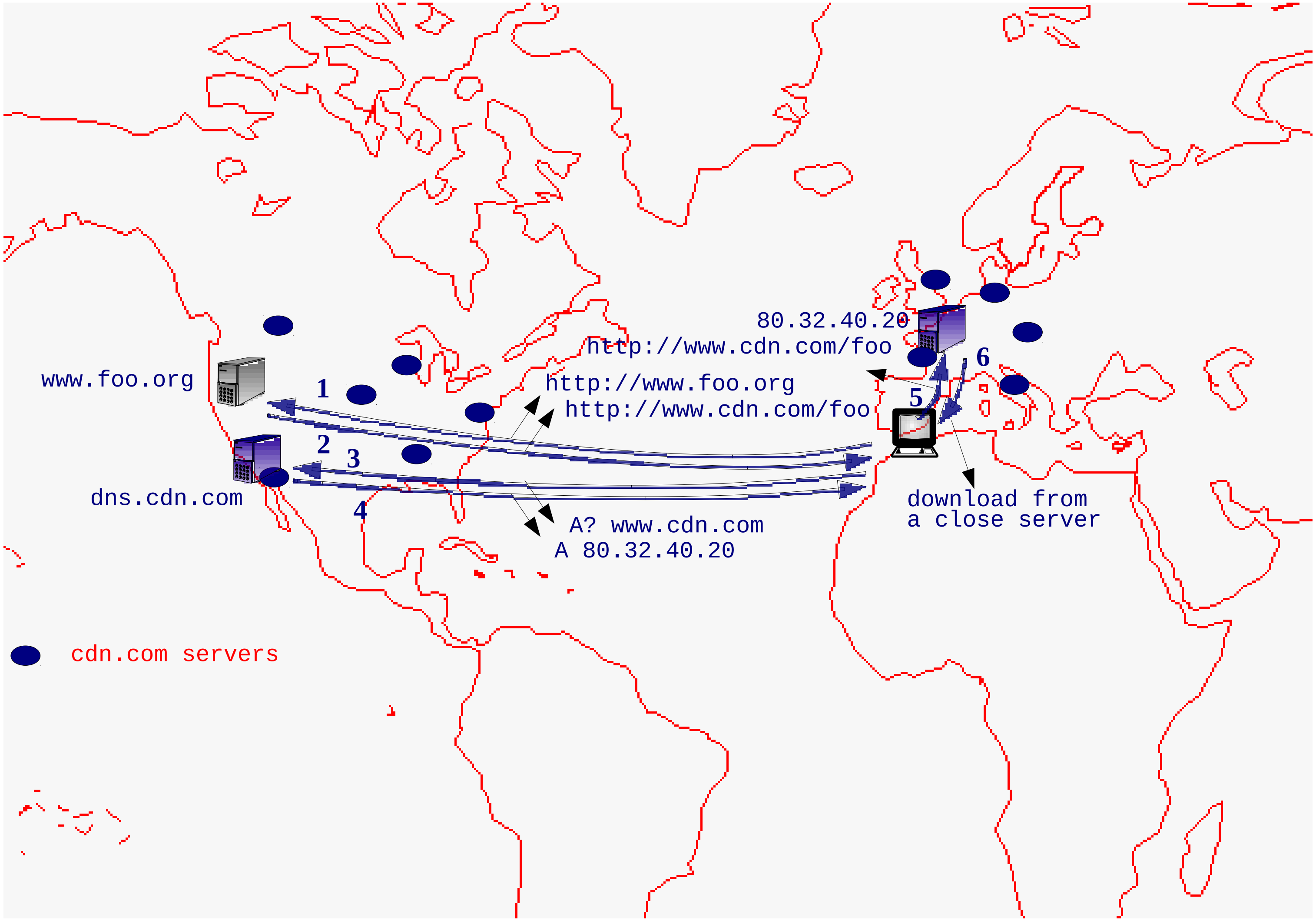
- Example using dig:

```
linux ~> dig www.microsoft.com

; <<>> DiG 9.3.2 <<>> www.microsoft.com
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 31808
;; flags: qr rd ra; QUERY: 1, ANSWER: 9, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;www.microsoft.com.            IN      A

;; ANSWER SECTION:
www.microsoft.com.       3135    IN      CNAME   toggle.www.ms.akadns.net.
toggle.www.ms.akadns.net. 181    IN      CNAME   g.www.ms.akadns.net.
g.www.ms.akadns.net.     181     IN      CNAME   lb1.www.ms.akadns.net.
lb1.www.ms.akadns.net.   181     IN      A       207.46.19.60
lb1.www.ms.akadns.net.   181     IN      A       207.46.18.30
lb1.www.ms.akadns.net.   181     IN      A       207.46.20.60
lb1.www.ms.akadns.net.   181     IN      A       207.46.19.30
lb1.www.ms.akadns.net.   181     IN      A       207.46.198.30
lb1.www.ms.akadns.net.   181     IN      A       207.46.225.60

;; Query time: 42 msec
;; SERVER: 192.168.1.1#53(192.168.1.1)
;; WHEN: Sun Mar 11 10:48:11 2007
;; MSG SIZE  rcvd: 203
```

```
linux ~> dig www.microsoft.com

; <<>> DiG 9.3.2 <<>> www.microsoft.com
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 17923
;; flags: qr rd ra; QUERY: 1, ANSWER: 9, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;www.microsoft.com.            IN      A

;; ANSWER SECTION:
www.microsoft.com.       3469    IN      CNAME   toggle.www.ms.akadns.net.
toggle.www.ms.akadns.net. 215    IN      CNAME   g.www.ms.akadns.net.
g.www.ms.akadns.net.     215     IN      CNAME   lb1.www.ms.akadns.net.
lb1.www.ms.akadns.net.   215     IN      A       207.46.198.30
lb1.www.ms.akadns.net.   215     IN      A       207.46.199.30
lb1.www.ms.akadns.net.   215     IN      A       207.46.18.30
lb1.www.ms.akadns.net.   215     IN      A       207.46.19.60
lb1.www.ms.akadns.net.   215     IN      A       207.46.198.60
lb1.www.ms.akadns.net.   215     IN      A       207.46.20.60

;; Query time: 43 msec
;; SERVER: 192.168.1.1#53(192.168.1.1)
;; WHEN: Sun Mar 11 10:42:38 2007
;; MSG SIZE  rcvd: 203
```

Unit 2. Network applications

DNS - Content Distribution Networks, example



Unit 2. Network applications

DNS – Messages: Message Format

- All DNS messages have the same format:
  - Header: type of message.
  - Question: What is to be resolved.
  - Answer: Answer to question.
  - Authority: Domain authority names.
  - Additional: Typically, the authority name's addresses.

```
-----------------------------------------------
|              Header (12 bytes)              |
-----------------------------------------------
/              Question (variable)            /
-----------------------------------------------
/              Answer (variable)              /
-----------------------------------------------
/              Authority (variable)           /
-----------------------------------------------
/              Additional (variable)          /
-----------------------------------------------
```

Unit 2. Network applications

DNS – Messages: Header

- Identification: 16 random bits used to match query/response
- Flags. Some of them:
  - Query-Response, QR: 0 for query, 1 for response.
  - Authoritative Answer, AA: When set, indicates an authoritative answer.
  - Recursion Desired, RD: When set, indicates that recursion is desired.
- The other fields indicate the number of Questions, Answer, Authority and Additional fields of the message.

```
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 bits
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          Identification         |       Flags                 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          #Questions             |       #Answers              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          #Authorities           |       #Additional           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Unit 2. Network applications

DNS – Messages: Question

- QName: Indicates the name to be resolved.
- QType: Indicates the question type:
  - Address, A.
  - Name Server, NS.
  - Pointer, PTR: For an inverse resolution.
  - Mail Exchange, MX: Domain Mail Server address.
- Qclass: For Internet addresses is 1.

```
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 bits
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
/                        QName (variable)                       /
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          QType                |       QClass                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

```
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 bytes
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|6|r|o|g|e|n|t|2|a|c|3|u|p|c|3|e|d|u|0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Codification example of `rogent.ac.upc.edu`

Unit 2. Network applications

DNS – Messages: Resource Records (RRs)

- The fields Answer, Authority and Additional are composed of RRs:
  - Name, Type, Class: The same as in the Question field.
  - TTL (Time To Live): Number of seconds the RR can be cached.
  - RDLenth: RR size in bytes.
  - Rdata: An IP address if the Type is 'A', or a name if the Type is 'NS'.

```
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 bits
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
/                        Name (variable)                        /
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            Type               |          Class                |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             TTL                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          RDLenth              |         RData (variable)      /
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Unit 2. Network applications

DNS – Messages: Example

```
# tcpdump -s1500 -vvpni eth0 port 53
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 200 bytes
11:17:30.769328 IP (UDP, length: 55) 147.83.30.137.1042 > 147.83.30.70.53: 36388+ A? ns.uu.net. (27)
11:17:30.771324 IP (UDP, length: 145) 147.83.30.70.53 > 147.83.30.137.1042: 36388
                 q: A? ns.uu.net. 1/2/2 ns.uu.net. A 137.39.1.3
                 ns: ns.uu.net. NS auth00.ns.uu.net., ns.uu.net. NS auth60.ns.uu.net.
                 ar: auth00.ns.uu.net. A 198.6.1.65, auth60.ns.uu.net. A 198.6.1.181 (117)
```

Query message:

- **36388: Identifier.**
- **+: Recursion-Desired is set.**
- **A?: Qtype = A.**
- **ns.uu.net.: Name to resolve.**

Response message:

- **36388: Identifier.**
- **q: A? ns.uu.net.: Repeat the Question field.**
- **1/2/2: 1 Answers, 2 Authorities, 2 Additional follows.**
- **ns.uu.net. A 137.39.1.3: The answer (RR of type A, address: 137.39.1.3).**
- **ns: ns.uu.net. NS auth00.ns.uu.net., ns.uu.net. NS auth60.ns.uu.net.: 2 Authorities (RRs of type NS: the domain ns.uu.net. authorities are auth00.ns.uu.net. and auth60.ns.uu.net).**
- **ar: auth00.ns.uu.net. A 198.6.1.65, auth60.ns.uu.net. A 198.6.1.181: 2 Additional (RRs of type A: authorities IP addresses).**

Unit 2: IP Networks

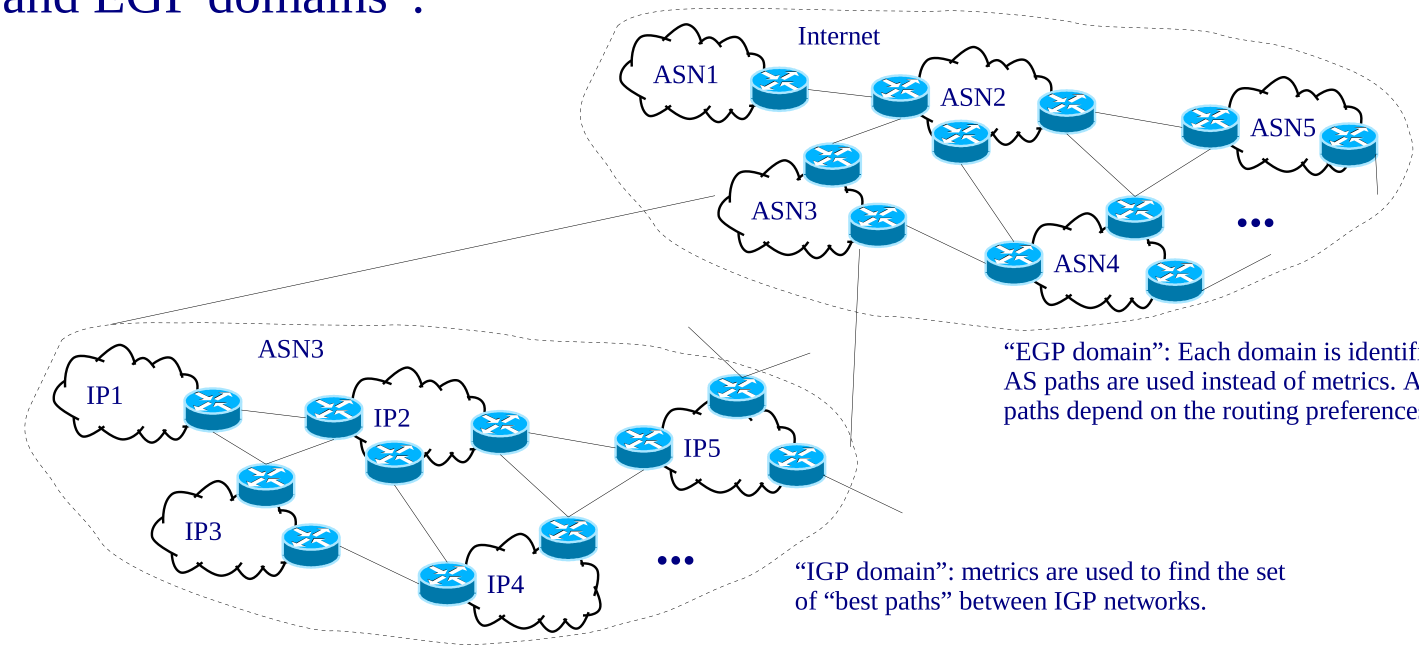
Outline

- IP layer service
- IP addresses
- Subnetting
- Routing tables
- ARP protocol
- IP header
- ICMP protocol
- DHCP protocol
- NAT
- DNS
- **Routing algorithms**
- Security in IP

Unit 3: IP Networks

Routing algorithms - Autonomous Systems (AS)

- AS definition (RFC 1930): “An AS is a connected group of one or more IP prefixes run by one or more network operators which has a SINGLE and CLEARLY DEFINED routing policy”.
- Each AS is identified by a 16 bits AS Number (ASN) assigned by IANA.
- ASs facilitate Internet routing by introducing a two-level hierarchy: “IGP and EGP domains”.



“EGP domain”: Each domain is identified by a ASN. AS paths are used instead of metrics. Advertised AS paths depend on the routing preferences between ASs.

“IGP domain”: metrics are used to find the set of “best paths” between IGP networks.

Unit 3: IP Networks

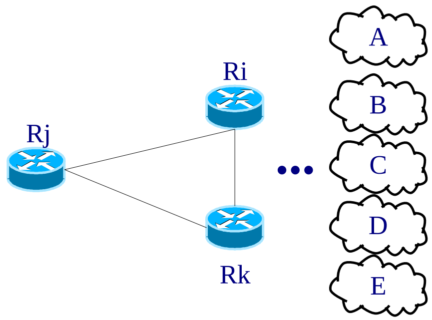
Routing Information Protocol, RIP (RFC 2453)

- The metric (distance) to a destination is the number of hops (i.e. transmissions) to reach the destination: 1 if the destination is attached to a directly connected network, 2 if 1 additional router is needed ...
- Routers send RIP updates every 30 seconds to the neighbors.
- RIP updates use UDP, src./dst. well-known port = 520, broadcast dst. IP addr.
- RIP updates include destinations and metrics tuples.
- A neighbor is considered down if no RIP messages are seen during 180 seconds.
- Infinite metric is 16.
- Two versions of RIP: Version 2 allows variable masks ans uses the multicast dst. address 244.0.0.9 (all RIPv2 routers).
- The routing algorithm is known as “distance-vector” or “Bellman-Ford algorithm”.

Unit 3: IP Networks

RIP – Routing Table (RT) Update Example

- Example: When Ri receives an update message from Rj:
  - Increase the message metrics.
  - Add new destinations.
  - Change entries with other routers with larger metrics.
  - Update metrics using Rj's gateway.



D	G	M
A	Rk	4
B	Rj	3
C	Rk	5
D	Rj	2

Ri's RT

➡

D	M
A	1
B	4
C	5
D	1
E	3

Ri receives Rj's update message

➡

D	M
A	2
B	5
C	6
D	2
E	4

Rj's metrics increased

➡

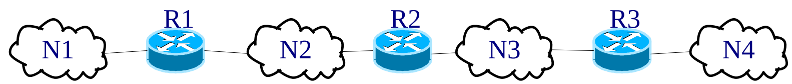
D	G	M
A	Rj	2
B	Rj	5
C	Rk	5
D	Rj	2
E	Rj	4

Ri's RT updated

Unit 3: IP Networks

RIP – Count to Infinity

- Depending on the route update message order, convergence problems may arise:



D	G	M
N1	*	1
N2	*	1
N3	R2	2
N4	R2	3

R1's RT

D	G	M
N1	R1	2
N2	*	1
N3	*	1
N4	R3	2

R2's RT

D	G	M
N1	R2	3
N2	R2	2
N3	*	1
N4	*	1

R3's RT

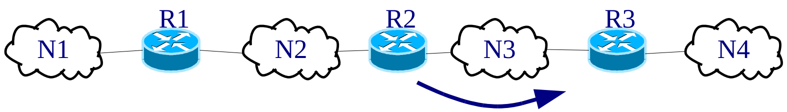
- Evolution of D=N4 entry when R3 fails:

	G	M	R3 fails	G	M	R1 upd	G	M	R2 upd	G	M	R1 upd	G	M		G	M
R1:	R2	3	→	R2	3	→	R2	3	→	R2	**5**	→	R2	5	...	R2	**16**
R2:	R3	2	→	R3	**16**	→	R1	**4**	→	R1	4	→	R1	**6**	...	R1	**16**

Unit 3: IP Networks

RIP – Count to Infinity Solutions

- Split horizon: When the router sends the update, removes the entries having a gateway in the interface where the update is sent:



D	G	M
N1	R1	2
N2	*	1
N3	*	1
N4	R3	2

R2's RT

D	M
N1	2
N2	1

update sent by R2

- Split horizon with Poisoned Reverse: Consists of adding the entries having a gateway with M=16.
- Triggered updates: Consists of sending the update before the 30 seconds timer expires when a metric change in the routing table.
- Hold down timer (CISCO): When a route becomes unreachable (metric = 16), the entry is placed in *holddown* during 280 seconds. During this time, the entry is not updated.

Unit 3: IP Networks

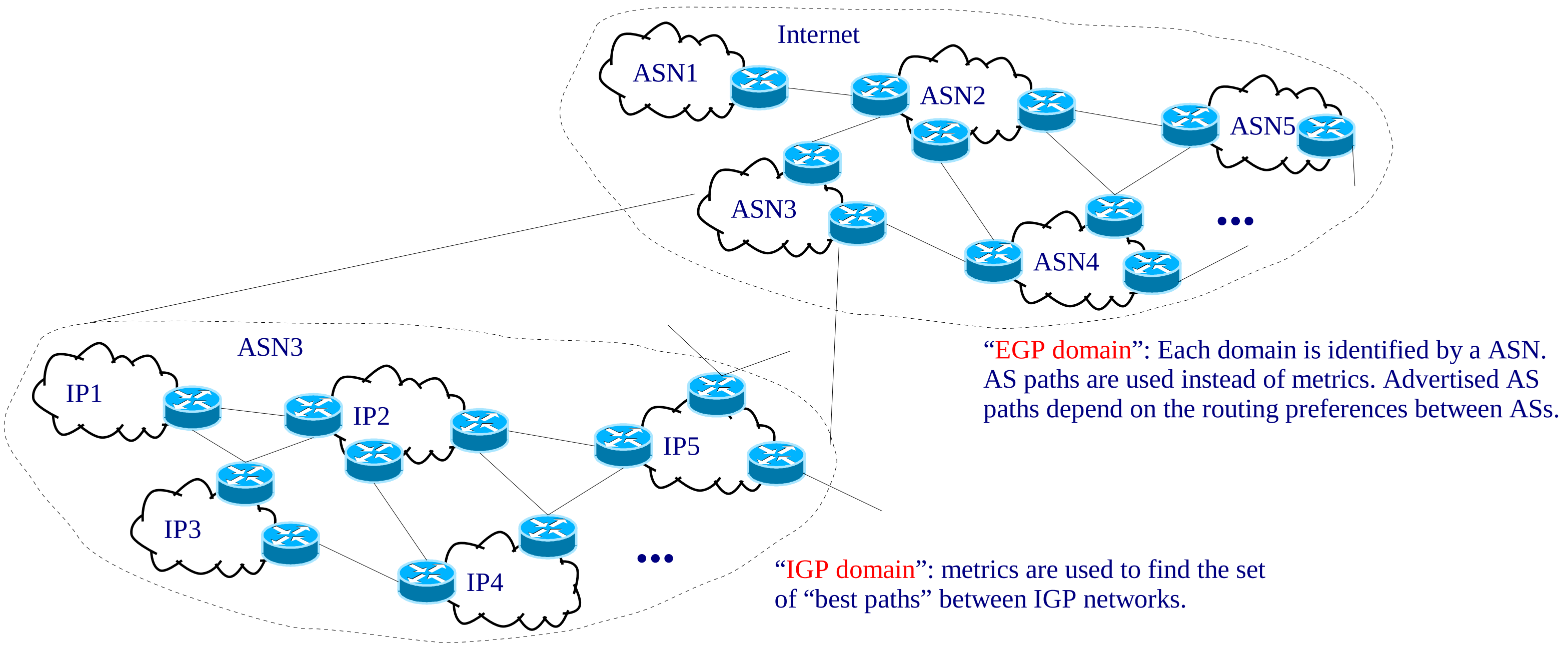
Open Shortest Path First, OSPF (RFC 2328)

- IETF standard for high performance IGP routing protocol.
- *Link State* protocol: Routers monitor neighbor routers and networks and send this information to all OSPF routers (*Link State Advertisements*, LSA).
- LSA are encapsulated into IP datagrams with multicast destination address 224.0.0.5, and routed using *flooding*.
- LSA are only sent when changes in the neighborhood occur, or when a LSA Request is received.
- Neighbor routers are monitored using a *hello protocol*.
- OSPF routers maintain a LS database with the information received with LSA. The Shortest Path First algorithm (Dijkstra algorithm) is used to optimal build routing table entries.
- The metric is computed taking into account link bitrates, delays etc.
- The infinite metric is the maximum metric value.
- There is no convergence (count to infinity) problems.

Unit 2: IP Networks

Border Gateway Protocol, BGP (RFC 1771, 1772)

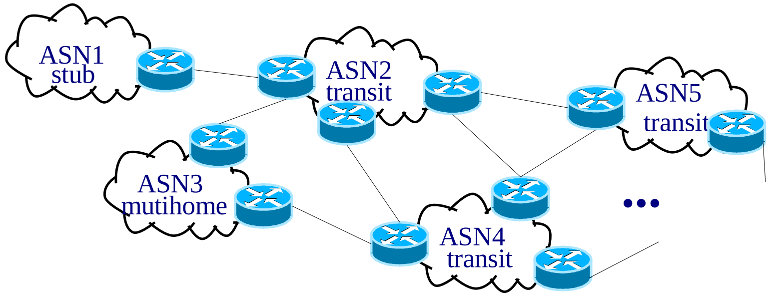
- BGP is the routing protocol used among ASs in Internet:



Unit 2: IP Networks

BGP, ASs Classification

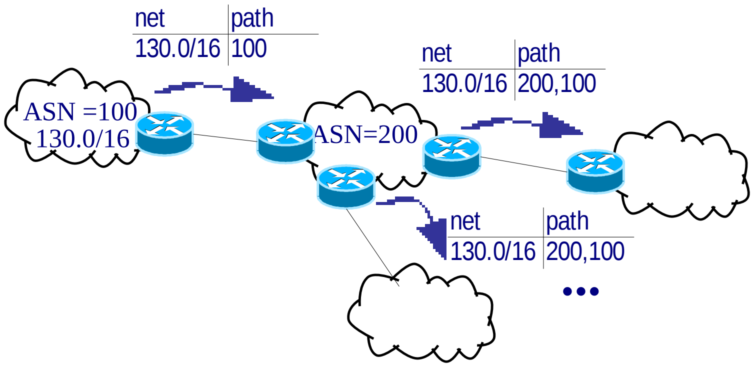
- Stub: Only carries local traffic and is connected to only one AS.
- Multihomed: Only carries local traffic and is connected to more than one AS.
- Transit: Route traffic from other ASs.



Unit 2: IP Networks

BGP, Basis

- BGP peers establish a permanent TCP connection, well-known port: 179.
- BGP peers exchange messages with network prefixes where they are willing to send traffic, and the *ASN path* to reach them.
- ASN path and other BGP Attributes are computed depending on the AS policies.
- Loops are detected and avoided by checking the own ASN with the ASN received in the BGP messages.
- BGP information is distributed among internal AS BGP routers.
- BGP message information is stored in a Routing Information Base (RIB). RIB is used to add routing table entries.



Unit 2: IP Networks

Outline

- IP layer service
- IP addresses
- Subnetting
- Routing tables
- ARP protocol
- IP header
- ICMP protocol
- DHCP protocol
- NAT
- DNS
- Routing algorithms
- **Security in IP**

Unit 3: IP Networks

Security in IP

- Goals:
  - Confidentiality: Who can access.
  - Integrity: Who can modify the data.
  - Availability: Access guarantee.
- Vulnerabilities:
  - Technological: Protocols (e.g. ftp and telnet send messages in “clear text”) and networking devices (routers...)
  - Configuration: Servers, passwords, ...
  - Missing security policies: Secure servers, encryption, firewalls, ...

Unit 3: IP Networks

Security in IP – Attacks

- Reconnaissance: Previous to an attack.
  - Available IP addresses.
  - Available servers and ports.
  - Types of OSs, versions, devices...
  - Eavesdropping
- Access: Unauthorized access to an account or service.
- Denial of Service: Disables or corrupts networks, systems, or services.
- Viruses, worms , trojan horses...: Malicious software that replicate itself.

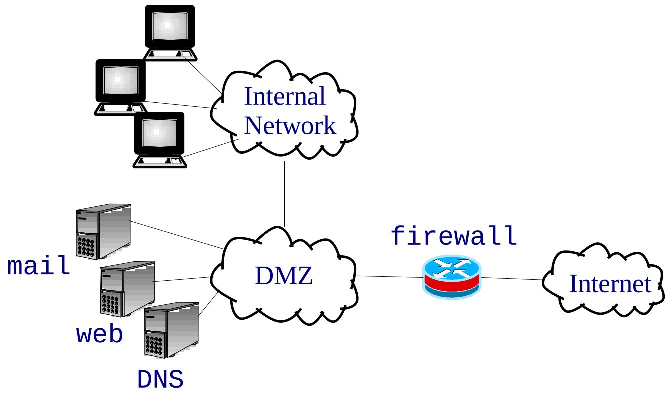
Security in IP – Basic Solutions

- Firewalls.
- Virtual Private Networks (VPN).

Unit 3: IP Networks

Security in IP – Firewalls

- Firewall: System or group of systems that enforces an access control policy to a network.
- There are many firewall types: From simple packet filtering based on IP/TCP/UDP header rules, to state-full connection tracking and application-based filtering, defense against network attacks, ...

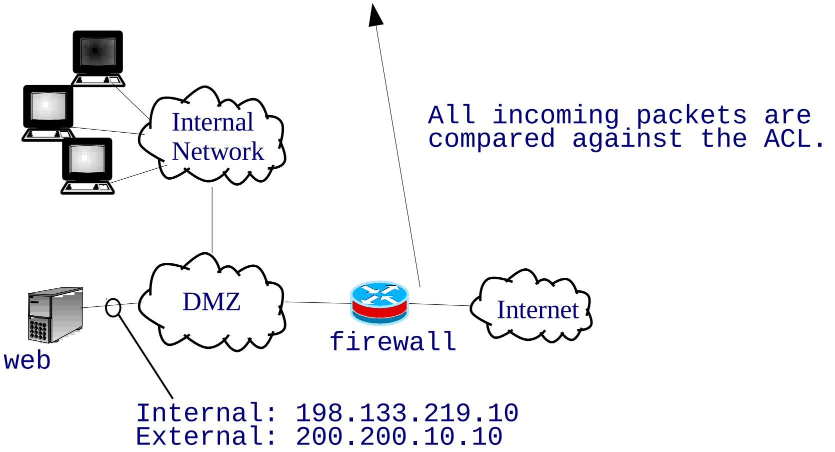


Unit 3: IP Networks

Security in IP – Basic Firewall Configuration

- NAT
- Access Control List, ACL

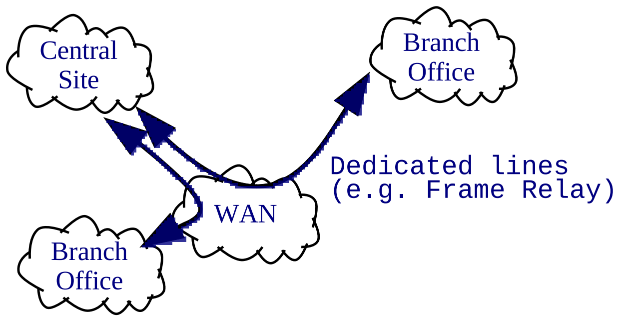
Protocol	IP-src	IP-dst	Port-src	Port-dst	Action
TCP	*any*	200.200.10.10/32	*any*	80	*accept*
TCP	*any*	*any*	< 1024	≥ 1024	*accept*
ICMP	*any*	*any*	–	–	*accept*
IP	*any*	*any*	–	–	*deny*

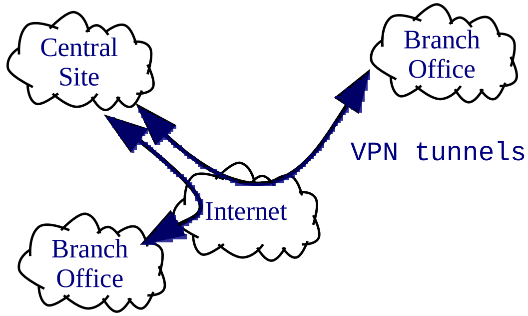


Unit 3: IP Networks

Security in IP – Virtual Private Network, VPN

- Provides connectivity for remote users over a public infrastructure, as they would have over a private network.





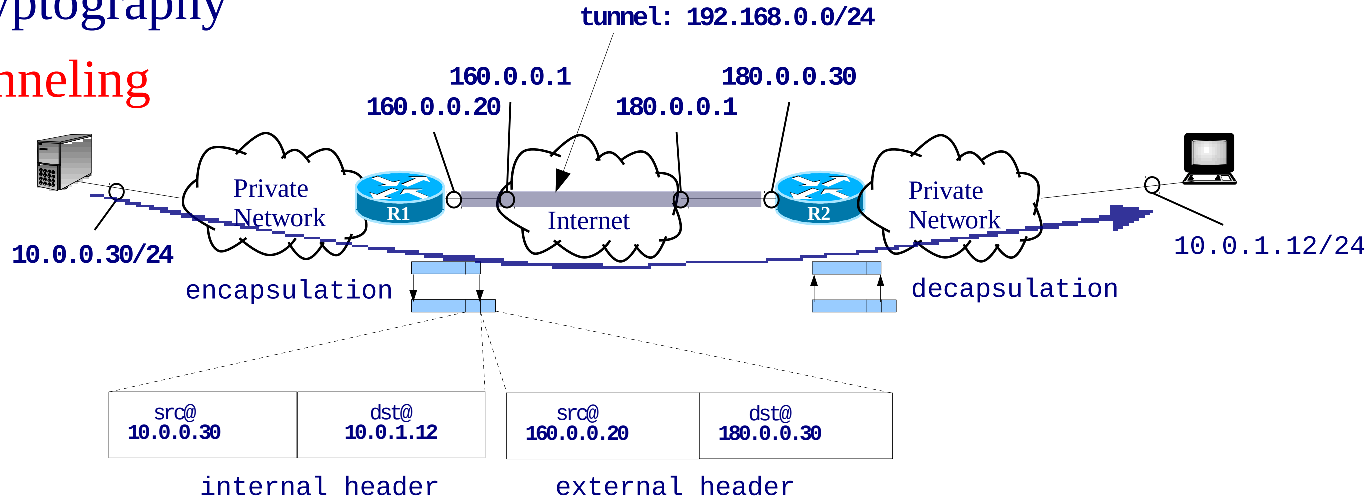
Conventional Private Network
- More cost.
- Less flexible.
- WAN management.

VPN
- Less cost.
- More flexible.
- Simple management.
- Internet availability.

Unit 3: IP Networks

Security in IP – VPN Security

- Authentication
- Cryptography
- Tunneling




```
Example: creating a tunnel in linux:
R1# ip tunnel add tun0 mode gre remote  180.0.0.30 local 160.0.0.20 ttl 255
```

Destination	Gateway	Genmask	Iface
10.0.0.0	0.0.0.0	255.255.255.0	eth0
160.0.0.1	0.0.0.0	255.255.255.255	ppp0
0.0.0.0	160.0.0.1	0.0.0.0	ppp0
192.168.0.0	0.0.0.0	255.255.255.0	tunl0
10.0.1.0	192.168.0.2	255.255.255.0	tunl0

R1 Routing Table

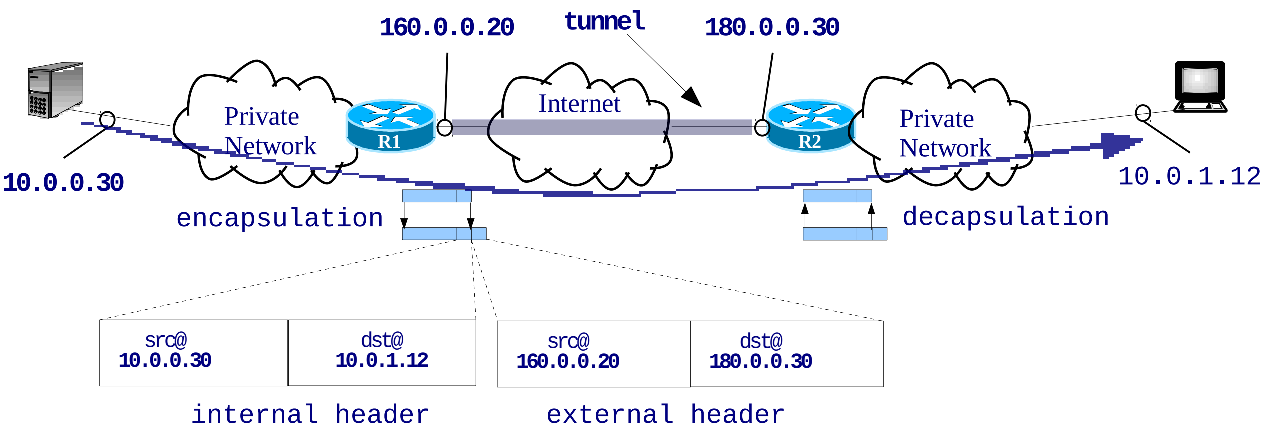
Destination	Gateway	Genmask	Iface
10.0.1.0	0.0.0.0	255.255.255.0	eth0
180.0.0.1	0.0.0.0	255.255.255.255	ppp0
0.0.0.0	180.0.0.1	0.0.0.0	ppp0
192.168.0.0	0.0.0.0	255.255.255.0	tunl0
10.0.0.0	192.168.0.1	255.255.255.0	tunl0

R2 Routing Table

Unit 3: IP Networks

Security in IP – VPN Tunneling Problems

- Fragmentation inside the tunnel will use the external header, thus, the exit router of the tunnel may reassemble fragmented datagrams.
- ICMP messages sent inside the tunnel are addressed to the tunnel entry.
- MTU path discovery may fail.
- Solution: the router entry maintains a "tunnel state", e.g. the tunnel MTU, and generate ICMP messages that would be generated inside the tunnel. Furthermore, the tunnel entry router typically fragment the datagrams, if needed, before encapsulation, to avoid the exit router having to reassemble fragmented datagrams.



Unit 3: IP Networks

Security in IP – VPN Tunneling

- IP over IP (RFC 2003): Basic encap.
- Generic Routing Encapsulation, GRE (RFC 1701): There is an additional GRE header: different protocol encap (not only IP).
- Point-to-Point Tunneling Protocol (RFC 2637): Add the ppp functionalities.
- IPsec (RFC 2401): Standards to introduce authentication and encryption and tunneling to IP layer.