Computer Networks - Lab Session 1
Part II - Static IP Routing

February 2013

1 Reminders

All the protocols of layer 3 of the OSI model (called the "network layer") allow communication between the various types of network of level 2 (the "links").

Several problems have to be solved, mainly with:

- different network types;
- general recognition of systems (universal addressing);
- packet routing (routing).

On the Internet, separate networks are interconnected and the protocols of the IP network layer provide a universal communications service. This service has to be independent of the structure and technology used locally on each of the networks.

Physically, two (or more) networks are interconnected by means of a device that has a point of attachment to each of the networks.

This device ? which plays a particular role in the networks ? is called a router (or gateway). Conversely, the user systems are called hosts. So, a router has several network interfaces and, therefore, Internet addresses. Here is an example with 2 Ethernet local area networks to which hosts and 2 routers are connected, each having two network interfaces. These two routers are connected via a network of another type.

The mechanism allowing the routing of packets to the right destination via the various networks is called routing. This routing can be of varying sophistication, such as "static" (routes are specified by an administrator) or "dynamic" (routes can vary over time according to different criteria: failures, load on links, etc.). The Internet address is divided into two parts:

- the address of the network on which the system is located;
- the system's address on this network.

Routing is performed at network address level. Routers base themselves on the first part of the address for forwarding messages to the destination network.

On broadcast networks (such as Ethernet), the ARP protocol handles the resolution of the Ethernet address that needs to be used to reach the proper interface.

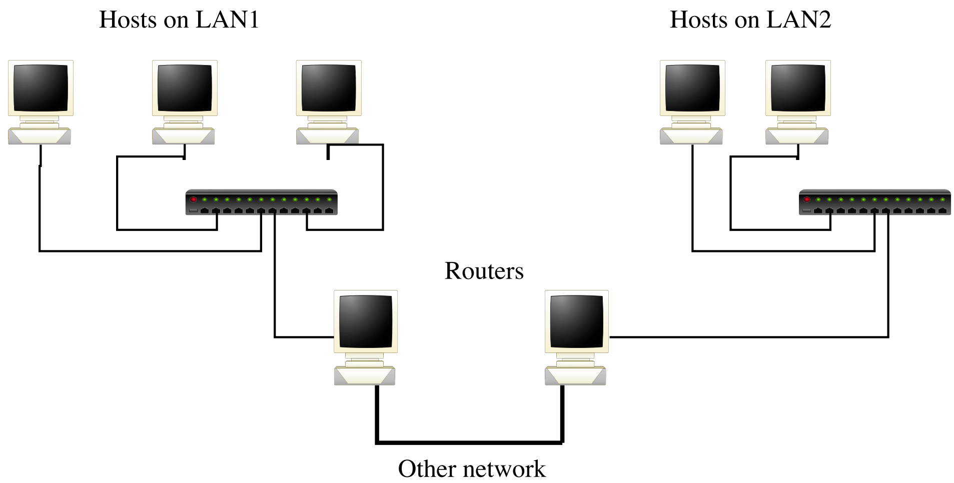

Figure 1:

To decide on the route to follow, routers manage a table called a "routing table" that lets them answer the question, "Given the destination address of the packet, which is the next router to send the packet to, so that it arrives at its destination?".

This router necessarily has a "neighbor". So, a routing table contains a list of pairs (network address, and neighboring router address) called routes. The complete path is not stored anywhere.

Let's take the analogy of a vehicle circulating on a road network. At each crossroads, the driver asks which road to take to get to the next crossroad, without ever knowing the complete path to the destination.

Host stations also have to manage such a routing table, without knowing which router they have to contact on their local network in order to reach the destination network.

We can distinguish two separate functionalities in routing:

- the decision-taking about the route to take, in view of the routing table and the destination address contained in the packet (IP header): this is packet switching;

- updating of routing tables, based on an often-partial knowledge of the network topology.

2 Performing operations with routing tables

We have seen during the lectures that an IP node is either a host or a router. Besides, nodes have a routing table to find the correct route for any packet. We will here manipulate this routing table manually. You will see in the next labs how you can use a routing protocol to populate these routing tables dynamically and how to react to network failures.
To display the routing table under FreeBSD, the command is:
`netstat -rn -f inet`
To update the table:

```
route  [add|delete] destination_prefix gateway
```
Example: route add 172.16.0.0/16 192.168.2.4

To enable a system to become a router (stations are not by default supposed to switch packets):

```
sysctl net.inet.ip.forwarding=1
```

3 Using the routing tables

Suppose that you have granted a single Internet address of class C (192.168.1/24, for example) for the local area network you have to manage. For a variety of reasons, you want to break your LAN down into 2 subnetworks (think about what such reasons might be).

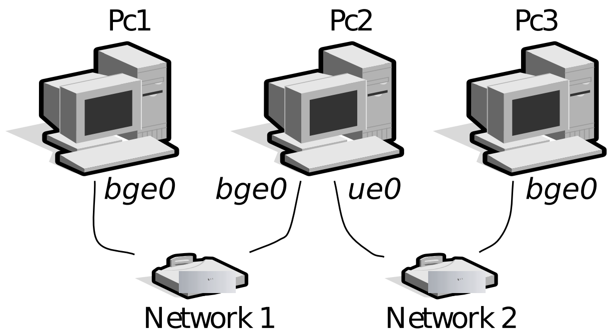

Figure 2: The network

As you can see, only one ethernet interface is avialable for experiments on the computers. Therefore, you can use a USB ethernet card in order to reproduce the platform shown in Figure 2. Once plugged, the new interface will appear as `ue0`

1. *Connect the three stations within the platform as shown in Figure 2. You can use crossover cables.*
2. *Produce an addressing plan for the platform, noting the interface numbers and addresses. Configure the interfaces of the computers and the router* .
3. *Using ping command determine which systems cannot communicate? why?*
4. *Configure the routing table of Pc1 and try to ping Pc3 from Pc1. What does happen? I suggest you to use wireshark (wireshark) between Pc2 and Pc3 to analyze the problem.*
5. *Finish the configuration so that any node can ping any destination in the network. Let describe the resulting routing tables.*

Notes: To specify a netmask that is different from the standard netmasks associated with classes (A, B, C)[1], you have to add the option -netmask followed by the netmask in decimal notation, or specify the number of bits of the network part.

[1]There are several **classes of Internet addresses**, which define the default length of the mask, for historical reasons. Here is a summary, for each class, of the bits reserved for encoding the network

Example: route -v add -net 192.168.0.64 192.168.0.66 -netmask 255.255.255.192 or else route -v add -net 192.168.0.64/26 192.168.0.66

6. *Run "arp -a -d" to delete arp tables on all stations. Now ping Pc3 from Pc1 and list the packets interchanged during the ping. You can in particular verify the behavior of ARP in a multi-hop route: is ARP used one time, for the first packet only, for each hop, etc. ?*

7. *Using the command traceroute, measure the transfer times between Pc1 and Pc2 and between Pc1 and Pc3. Sniff the packets sent by* `traceroute`*. Verify how* `traceroute` *manages to trigger an answer from the intermediate systems.*

4 Routing in the real Internet

Connect back on station to the school network. Re-enable the default interface for this network.

Launch a `traceroute` *to a distant system (on a different continent). Check that the delays that you observe make sense.*

address (r bits) and those reserved for the encoding of the hardware system's address on the network (m bits).
Class A:`0rrrrrrr.mmmmmmmm.mmmmmmmm.mmmmmmmm`; Class B: `10rrrrrr.rrrrrrrr.mmmmmmmm.mmmmmmmm`; Class C: `110rrrrr.rrrrrrrr.rrrrrrrr.mmmmmmmm`