Genie Routing lab

Laboration in data communications
Department of Information Technology, Uppsala University

Overview

This lab deals with linux network setup and routing in the network layer

Administration

Student 1

Name: ______________________________

Email: ______________________________

Personal number: ______________________________

Student 2

Name: ______________________________

Email: ______________________________

Personal number: ______________________________

Agreement

I/we have independently worked on the following assignment solution. In the case of two people, we have both taken part in creating the solution, according to the assignment specification.

Sign 1: ______________________________

Sign 2: ______________________________

General

Course instance (e.g. Datakom DV1): ______________________

Date: ____

Notes to the lab assistant:

____

____

____

____

____

____

Comments from the lab assistant:

____

____

____

____

____

____

Grade: ____

Sign: ____

1 About the lab

This lab will give you an insight in intra-domain routing. Intra-domain routing is routing performed within a single administrative system (AS), e.g. within a company. It will concentrate on the network layer (i.e. layer 3 in the OSI model) and does not require any previous knowledge of other layers.

It is a "hands on" lab and you you will be using systems that are used today in the Internet.

First you will set up your machine manually, by setting the IP address, netmask, etc. and discover why routing is necessary. You will be given the opportunity to gain knowledge about how to allocate address space in an efficient way and see in practice that your network nodes are able to communicate accordingly to the specification.

Enjoy the lab!

Objectives

- IP addressing
- Routing
- Practicing network tools and command usage

Reading instructions

Computer Networking - a top-down approach featuring the Internet (2nd edition) Kurose, Ross

- 4.1-4.2 (routing)
- 4.4 (IPv4), especially 4.4.1 (datagram format)

Theoretical and practical questions

There are a number of questions marked with an asterisk (*) which do not require you to use your computer. I.e., these questions can be done after the lab, in case you run out of time in the lab room.

2 Preparatory assignments

General

Background information

IP - the Internet Protocol.

One of the purposes of the Internet is to enable communication between computers. To be able to distinguish between computers/nodes in the Internet, all nodes have unique IP addresses. An IP (version 4) address is four bytes long and is often written in dotted-decimal notation, e.g. 130.238.8.1.

One could also express the IP address in binary format.

10000010 10001010 00001000 000000001 is the binary representation of 130.238.8.1.

A portion of the IP address (the left part of the binary notation) will determine the network to which the node is connected. The rest of the IP address will determine a unique host on that IP network.

However, as you probably know there are networks of different sizes, from two nodes to several thousands or even millions of nodes. How can you tell how big the network portion of the IP address is? This is determined by the network mask. For the example address above it is determined that 254 computers could belong to this network. So to distinguish between 254 computers in this network we need 8 bits ($2^8 = 256$) of the address to be used for host identification. The right-most 8 bits are used for this purpose and the “rest” 24 bits are used to identify the network. We use something called the netmask to represent how big our network is, i.e., where the boundary is between the network and the host part of the IP address. All bits that are used for the networks are set to 1, the rest to 0.

In our example the netmask is:

11111111 11111111 11111111 0000000

or more commonly written 255.255.255.0.

“Internet - the network of networks”

The way Internet is designed and built is that is consists of many small LAN networks that are interconnected. This design requires some kind of mechanism to send data between networks. Traditionally, this has been done using some kind of forwarding mechanism, in the case of IP, IP forwarding. It’s basically a node which has connectivity to at least two networks, and whose task is to send traffic from one network to another (i.e. to forward data). A node with this task is called a router. In the simple case of two networks this is quite easy, but in more complex network topologies like the Internet, there has to be a systematic way to find a route from the source to the destination – typically by knowledge of which networks other routers are connected to. This can be formalized into a table, the routing table. Every machine in an IP network uses its own routing table to determine where to send packets; on normal workstations these are usually quite simple, but routing tables tend to be more complex in routers on the core/backbone network.

IP routing and IP forwarding are two things that are often mixed up, and used interchangeably. IP routing is the process of updating and calculating the routing table used for IP forwarding. Often, a routing protocol such as Routing Information Protocol (RIP) or Open Shortest Path First (OSPF) is used to maintain the routing table. The actual packet forwarding, i.e. receiving a packet on one interface and sending it out on another interface (as defined by the routing table) is called IP forwarding.

Basically, the motivation why routing/IP forwarding is needed is that one want to interconnect several networks, such as the Internet network of networks.

Routing in theory

The routing table consists of N rows of:
[IP address, netmask, cost, next hop, interface]

For each row *n* in the routing table calculate [IP(n) & mask(n)], "&" denotes the bitwise logic AND-operator. The result for each row *n* will be the network to which *IP(n)* belongs.

When a packet with destination address Y should be delivered, for each row *n* calculate [Y & mask(n)] and compare each [Y & mask(n)] to [IP(n) & mask(n)].

If [Y & mask(x)] is the same as [IP(x) & mask(x)], the row is considered to be a match. If several rows match, use the row whose mask contains the most 1:s in the binary representation.

If no row matches, a default route (if it exists) is used. The default route is where all packets not captured by any other rule is sent to. If no default route exists and no row matches, the packet is silently dropped.

Running the system

The lab environment runs from a Linux CD. Put the CD into the CD drive on your PC, and boot from it. This will start The Knoppix OS distribution. After a while, you will see a boot prompt. Simply press enter to continue booting the system. When the system is done booting, you will be presented with a graphical desktop.

On the desktop are a few icons. To start the Routing lab, click the "Start Routing Lab Stage 1" icon once. This will pop up four terminal windows, each running a virtual Linux computer. You can login to these computers as **root** with password **lab**.

Some useful commands for IP Networking in Linux

A substantial part of this lab will be about how to setup IP networking in Linux. There are some basic commands that you probably want to use, listed below.

In Linux the network interfaces are called eth*X*, where *X* is a number. The first network card is called eth0, the second eth1, etc. Remember that in the virtual lab environment ethX corresponds to a physical network card so it is important to setup the correct interface for the correct network. You can imagine the virtual network cables connecting the cards on the different virtual machines, if you configure two cards that are not connected with a cable to each other obviously nothing will work.

ifconfig - activate and/or configure a network card; set IP, netmask etc.

```
ifconfig eth0 192.168.200.1 netmask 255.255.255.0 up
```

route - show / manipulate the IP routing table

```
route add -net 192.168.205.0 gw 192.168.200.1
route -n
```

traceroute - print the route packets take to network host

```
traceroute -n 192.168.205.1
```

sysctl net/ipv4/ip_forward - show / manipulate IP forwarding rule.

`sysctl -w net/ipv4/ip_forward=1` - enable IP forwarding

The **-n** flag used for `route` and `traceroute` indicate that no DNS lookup will be made, this speeds up the output since no DNS is available in this network.

Use the man pages (`man <command>`) to retrieve additional information on how to use these commands. Man pages can seem a bit intimidating at first but they are a vital tool when working in any Linux/UNIX environment so it is a good idea to become comfortable reading and using them.

Note: The word gateway is in most cases synonymous with router, meaning a computer which acts as a forwarder of IP packets (or more generically; forwarder of *data*). More specifically a gateway is an exit-/entrypoint of a certain network.

Scenario

A simple network

A corporate business is located at two sites, Stockholm and Gothenburg (see figure on page 7). At each location there exists a LAN. Each LAN has several workstations but you are only allowed access to one (gbg-workstation in Gothenburg, sthlm-workstation in Stockholm) and one router (gbg-router and sthlm-router). Each such LAN can be used to communicate within the site, but not with the other site. A wire is setup between Stockholm and Gothenburg, but it's a leased line which costs per transferred amount of data. The simplest solution is to "extend" the LAN:s with a bridge so it can be seen as a single big LAN. However, this might not be what we want, as traffic inside an office will have to travel to the other office, causing us to pay for this unnecessary traffic flowing on the leased line.

Another solution is to use routing/IP forwarding. This allows us to make sure that out of all packets originating at the Stockholm office, only those destined for

Gothenburg will flow on the leased line to Gothenburg. Correspondingly, traffic inside an office will never leave that office. The way this is done is by using two IP networks.

A router is placed at each location, connected to both the local office network and the remote office network (via the leased line).

The Stockholm network is given the IP network address 10.1.0.0 and uses the netmask 255.255.0.0. This means that it can contain 2^{16} (-2 for technical reasons; one address is used to identify the network itself, and one address is used for broadcasts) nodes/computers. Similarly, the Gothenburg network is given the IP network address 10.2.0.0 and also uses the netmask 255.255.0.0.

Computers in the Stockholm network may have addresses ranging from 10.1.0.1 to 10.1.255.254. Recall that each octet of the IP address can take on 255 different values, so 10.1.353.3 is an example of an illegal address.

Similarly the Gothenburg network address range is 10.2.0.1 to 10.2.255.254.

By using the addressing scheme it is easy to determine whether we are communicating within our network or not. If it has the same numbers in the first two bytes of the IP address, we know it's inside our own network. More specifically, this is the purpose of the netmask. It decides how many bits that should be considered the network part and the host part, respectively. In our example, the netmask is 255.255.0.0. Converting it to binary format gives us 11111111 11111111 00000000 00000000. I.e. the first two bytes are all 1:s, meaning that the first two bytes are indicating the network part of the IP address.

Given an IP address and the netmask we can find out which IP network that address belongs to. For example a computer in Stockholm with IP address 10.1.10.10 belongs to network 10.1.0.0. We calculate this by taking the IP address and perform bitwise AND (&) with the netmask. 10.1.10.10 & 255.255.0.0 = 10.1.0.0.

The fact that we can distinguish which IP network an IP address belongs to makes it possible to decide which traffic should go over the leased line.

In practice

The first step is to setup IP addresses for all computers in the network. The next step is to setup routing at each node. This is done by stating which computers that we can communicate with directly, i.e., which computers that are located on the same LAN, and what to do with packets destined for other computers. This is done by setting up routing rules, to tell which path/route the packet should take.

Before you start you should check to see that you have internet access, try surfing to **https://joshua.it.uu.se:4443/genie_labs/routing** (there is a link on the page that popped up when you started the system). If you cannot reach the webpage you won't be able to hand in your report electronically, you will either have to write the report manually and hand in a paper-copy or make sure you have some other way of getting the report off of the computer (e.g. usb memory stick or a floppy disk).

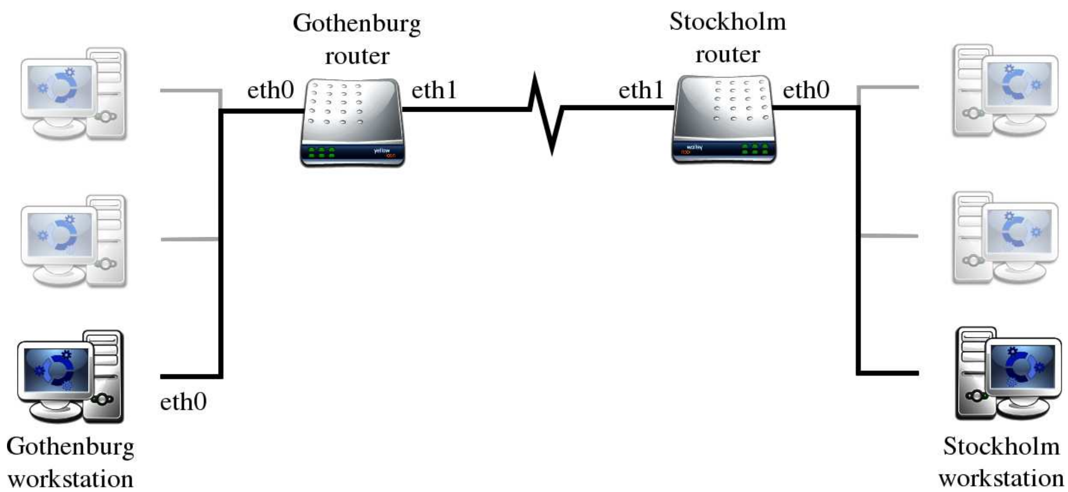

Figure 1: The network layout

Configuration of the workstations

When the lab-environment has started no network cards are activated so your first task will be to configure and activate them. The workstation(s) in Stockholm are connected to each other and the Stockholm router through their eth0 interfaces. Similarly, the workstations in Gothenburg connect to each other and the Gothenburg router through their eth0 interfaces. Start by bringing the eth0 interfaces up and assigning a suitable address to each interface

- `ifconfig eth0 10.1.0.10 netmask 255.255.0.0 up`

This configures and brings up the network card in the Stockholm Workstation

- `ifconfig eth0 10.2.0.10 netmask 255.255.0.0 up`

This configures and brings up the network card in the Gothenburg Workstation

Simply activating the network cards is not enough, we also need to add routing rules in order for packets to be forwarded correctly. First we want a rule that says that packets in our network should be sent directly to their destinations. This rule is often automatically setup when bringing an interface up with ifconfig. In the system used in this lab, this is true. You should check with `route -n` which routing rules exists before continuing.

Now the workstation knows how to reach its own network but it has no idea how to get to any other network. From our knowledge of the network layout we know that all packets originating in Stockholm destined for 10.2.0.0 (the Gothenburg network) should go through our router which has connectivity to the 10.2.0.0 network via the leased line. However, in practice one almost always sets up a different kind of route; a default route.

The default route is used for packets that do not match any previous rule, for a workstation the default route normally points to the closest router/gateway available.

To setup a route to a default router/gateway:

- `route add default gw 10.1.0.1`

To set up a *static* route pointing to the Gothenburg network one would enter the following command.

- `route add -net 10.2.0.0 netmask 255.255.0.0 gw 10.1.0.1`

10.1.0.1 is the IP address of the Stockholm router. Note that you only need *one* of the above rules to provide connectivity from Stockholm to Gothenburg.

Similarly, at Gothenburg's workstation, the rules will be either of:

- `route add default gw 10.2.0.1`
- `route add -net 10.1.0.0 netmask 255.255.0.0 gw 10.2.0.1`

The keyword "gw" in the lines above is short for gateway, which the attentative reader will know is usually synonymous to router. 10.2.0.1 is the IP address of the Gothenburg router.

Configuration of the routers

The routers/gateways in Stockholm and Gothenburg communicate with their respective network through their eth0 interface, to configure and bring these up enter the following commands:

- `ifconfig eth0 10.1.0.1 netmask 255.255.0.0 up`

This configures and brings up the network card in the Stockholm router.

- `ifconfig eth0 10.2.0.1 netmask 255.255.0.0 up`

This configures and brings up the network card in the Gothenburg router.

The leased line from Gothenburg to Stockholm is connected to the eth1 interfaces of the two routers.

We allocate a new network used for communication between the Stockholm and Gothenburg routers. It will only contain two nodes, so a small network is sufficient. We use the network 10.10.0.0 with netmask 255.255.255.252, which can contain 2 nodes (+ 1 network address + 1 broadcast address).

The Stockholm router eth1 interface is assigned the address 10.10.0.1. The corresponding Gothenburg eth1 interface is assigned 10.10.0.2.

In the Stockholm router:

- `ifconfig eth1 10.10.0.1 netmask 255.255.255.252 up`

In the Gothenburg router:

- `ifconfig eth1 10.10.0.2 netmask 255.255.255.252 up`

As with the workstations, we need to add some routing rules to let the routers know of the networks they are not directly connected with.

In the Stockholm router:

- `route add -net 10.2.0.0 netmask 255.255.0.0 gw 10.10.0.2`
  - to be able to communicate with the Gothenburg network.

In the Gothenburg router:

- `route add -net 10.1.0.0 netmask 255.255.0.0 gw 10.10.0.1`
  - to be able to communicate with the Stockholm network.

Not all computers act as routers in the world, in fact most don't so for this network setup to work we also have to enable forwarding of IP packets on the routers. This is done by manipulating the apropriate kernel parameter with the command sysctl (see 'Useful commands' earlier in this document).

- `sysctl -w net/ipv4/ip_forward=1` - to enable IP forwarding.

Now we should be able to communicate with both networks, and only traffic destined for the other network goes over the expensive and slow leased line.

A traceroute from the Stockholm workstation to the Gothenburg workstation should look something like this:

```
sthlm-workstation:~# traceroute -n 10.2.0.10
traceroute to 10.2.0.10 (10.2.0.10), 30 hops max, 38 byte packets
 1  10.1.0.1  2.600 ms  0.831 ms  0.802 ms
 2  10.10.0.2  3.517 ms  1.161 ms  1.156 ms
 3  10.2.0.10  7.695 ms  1.528 ms  1.514 ms
sthlm-workstation:~#
```

Make sure that you have setup the network and that it is functioning properly before you continue to assignment 1!

3 Lab assignments

Assignment 1

The company recently created a new office in Uppsala, which needs to be connected to the other offices in Stockholm and Gothenburg. Uppsala is connected to Stockholm via a leased line, but is not connected to Gothenburg. The office in Uppsala is small, so it will only use 250 computers.

To add the Uppsala office, click on the "Start Routing Lab Stage 2" icon once. This will produce two more terminals running virtual Linux computers. Allow

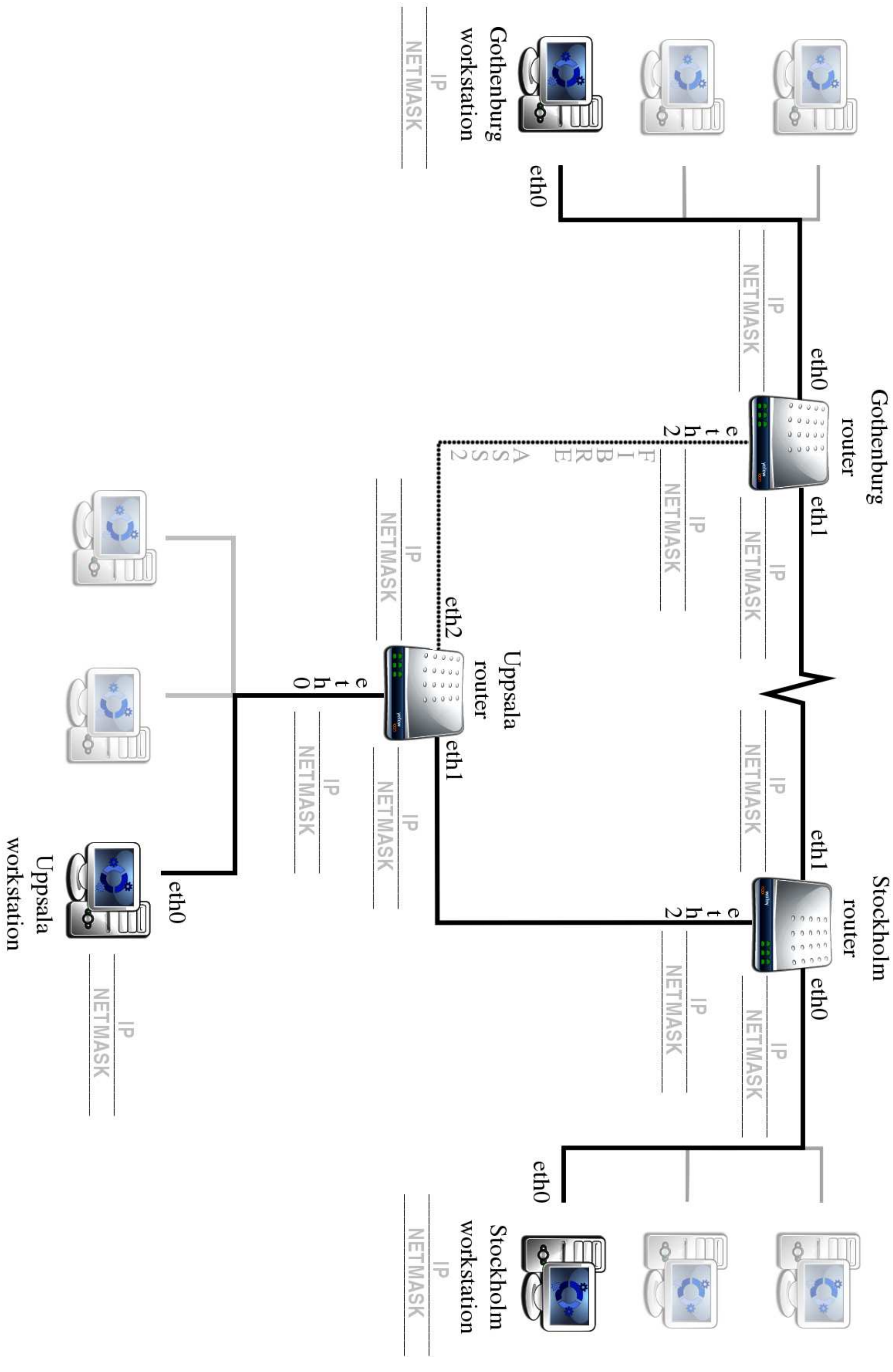

Figure 2: The network layout for assignment 1, 2 and 3

them to boot, and you will be able to log in as **root**, with password **lab**. The machines are connected to each other on eth0 like in Stockholm and Gothenburg, and eth1 on the Uppsala router is connected to eth2 on the Stockholm router.

Task: Allocate a suitable IP network and the longest possible netmask. Make necessary changes to the routing tables in order to allow the networks to communicate with each other. Test your setup by performing a traceroute from the workstation in Uppsala to the workstation in Gothenburg.

Report: On each virtual computer, run the command `source route_report 1` which will generate files in /root/report/routing/1 on the virtual machines. You can show the contents of a file by entering `more <filename>`. Also you should write down information about IP and netmask for the network you have created so far in the figure on page 11 .

Assignment 2

The company is offered a free fibre using Gigabit technology between Uppsala and Gothenburg. This means that all traffic preferably should flow over this fibre rather than the expensive and slow leased line. However, we don't want to remove the leased line altogether, in case the fibre line goes down. (The fibre line is rather unreliable, but what do you expect when you get things for free?). There is something called "metrics" which come in handy in situations like this.

The fibre is connected to eth2 on the routers in Uppsala and Gothenburg.

Task: Configure the interfaces and make changes in the routing tables to achieve the desired behaviour. Try to change the metric and see if you get the desired result. Test before and afterwards with traceroute to see that your change really worked.

Report: On each virtual machine, run the command `source route_report 2` which will generate files in /root/report/routing/2 on the virtual machines. You should also complement the figure on page 11 with IP and netmask for your newly configured interfaces.

Assignment 3

Finally the price for Internet connectivity has dropped enough that the company can afford to get an Internet connection! However, to save money, there is only one connection in Stockholm. Based on your recently acquired knowledge of routing, you should have a solution to how all offices can share the Internet connection.

From your Internet Service Provider (ISP) you get the following information on how to setup your network card:

IP	81.153.32.4
Gateway	81.153.32.1
Netmask	255.255.255.0

Task: Configure and activate the eth3 interface on the Stockholm router (which is connected to the ISP). Change the routing tables so that all computers can access the internet, try tracerouting to the ISP gateway to check if you have succeeded (note: adding a static route to the ISP gateway is not a valid solution).

Hint: default gateway/router

You can ignore the fact that the IP addresses used in this scenario are private. If you don't understand what this means, don't care.

Report: On each virtual machine, run the command `source route_report 3` which will generate files in /root/report/routing/3 on the virtual machines.

(Theoretical) Assignment 4

Your task is to assign network addresses in an optimal manner. In this exercise you should always use the longest possible netmask when assigning IP ranges. The layout of the physical network is illustrated in figure 3. The router in 'Net1' serves as the entrypoint for the entire network and should therefore announce a suitable network identifier and netmask so that it can address all its own hosts as well as distinguish between the subnets ('Net2' and 'Net4') and their hosts. Similarly the router in 'Net2' must be able to distinguish between its own hosts and the hosts in 'Net3' and the same goes for the router in 'Net4'. 'Net3' and 'Net5' have no subnets and only need address-space for their own hosts.

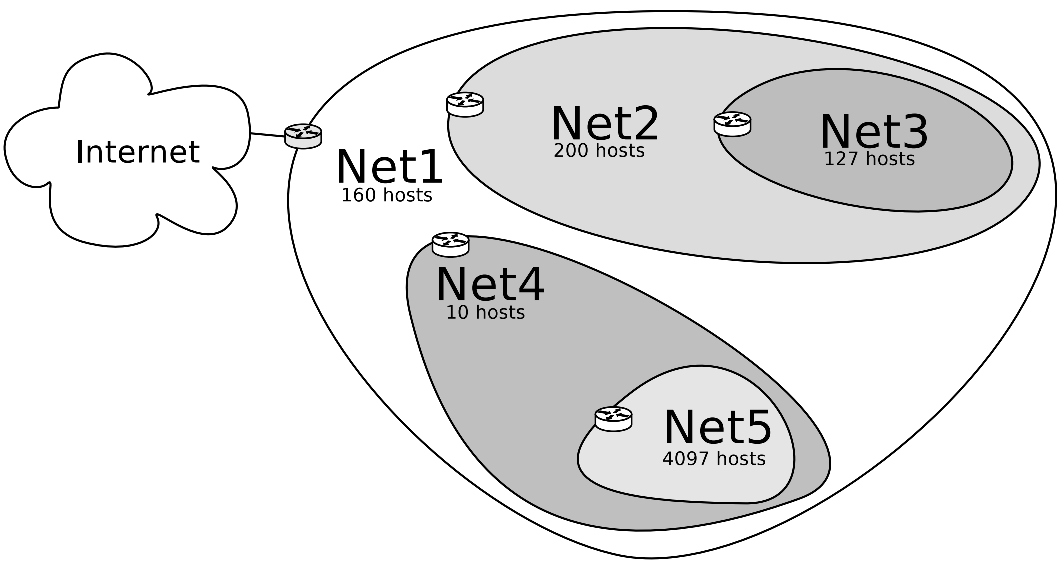

Figure 3: The network layout

Report: Fill in the tables below with suitable networks and bit-/netmasks. Assign IP addresses for the routers and fill in the routing table for the router in 'Net1'. "Aggregated bitmask" denotes the bitmask needed to address and properly distinguish between two or more *aggregated* networks.

	Net1	Net2	Net3	Net4	Net5
number of hosts	160	200	127	10	4097
bitmask *(/x)*					
aggregated bitmask	"				
aggregated bitmask					

	Router IP	Network	Netmask	/x
Net1				
Net2				
Net3				
Net4				
Net5				

4 The report

You should answer all of the questions above. Also make sure that you covered all subquestions.

You may write in English or in Swedish.

To hand in the electronic part of the report you click the icon "Hand in routing report" on the desktop once. This will direct you to the hand-in page. You will have to login using the ID you have been given for this course. The report file is located in /home/knoppix/report/routinglab.tar. If you cannot access the web you must transfer the file to a diskette/usb-stick or similar and access the hand-in page later. *No files can be saved to the harddrive! If you shut down the system without transferring your reportfile you will lose it!*

Also, you should hand in this document with your answers filled in (in a legible style). Submit the papers to the lab assistant's pigeonhole in Polacksbacken house 1, 4th floor. This should be done before the date announced at the web page for your course instance.

Routing table for router in Net1