

# ITS 332 Networking Lab

## Layer 3 Networking

Prepared by Dr Steven Gordon, 25 December 2007, Revision 85.

First Name: \_\_\_\_\_ Last Name: \_\_\_\_\_

Date: \_\_\_\_\_ ID:

### 1 Overview

Although Ethernet is a common technology for layer 2 networks, in particular LANs, there are in fact many different technologies for layer 2 networks, including for WANs: Ethernet, ADSL, SDH, Wireless LAN, Bluetooth, Token Ring, Frame Relay, ATM, . . . . Therefore to allow a user to communicate with any other user, independent of the LAN/WAN technology, layer 3 networking is used. Today, the Internet Protocol (IP) is the most commonly used layer 3 network technology. At layer 3, *routers* are used to connect LANs and WANs together, e.g. connect an Ethernet LAN to a SDH WAN; connect two Ethernet LANs together; connect a ATM WAN to a Frame Relay WAN; and so on.

In this lab you will create layer 3 network. That is, you will connect different LANs/WANs together using routers. The main tasks will be configuring the hosts and routers to use the appropriate addresses and routing tables.

### 2 Working in Groups

You will work in a group of 4, forming a group with the nearest 4 seats next to you (e.g. 1 to 4).

In all cases of working with others, you should share the tasks and make sure all members of the group understand the tasks being performed. Remember, the knowledge from the tasks will be tested in the exams, where you must work individually.

### 3 Routers

An internet is a collection of many different computer networks (LANs and WANs) connected together. Routers are devices that connect these individual networks together.

A router has two main roles:

1. Routing. This is the process of discovering suitable routes throughout an internet. This is normally done automatically (using a routing protocol) but routes can be created manually (we will see how in this lab). Think of a route as the path through the internet.

2. Forwarding. This is the process undertaken when a router receives an IP datagram. The router looks at the destination IP address in the datagram, and from the routers routing table, determines what is the next router (or host) to send the datagram to in order to reach the final destination. Then the router sends the datagram.

### 3.1 Routers and Hosts

What is the difference between a host (e.g. your PC) and router?

- When a host receives an IP datagram destined to itself, then the host will process the datagram by sending it to the relevant application (e.g. web browser). If the host receives an IP datagram destined to an IP address other than itself, the datagram will be dropped.
- In the case of receiving an IP datagram destined to itself, the router will do the same as the host. But when a router receives an IP datagram destined to another IP address, the router will look up its routing tables and forward the datagram to another computer (host or router).

A simple example: an IP datagram with destination address 200.0.0.3 is received at a computer with IP address 192.168.1.1. If the computer is a host, the datagram will be dropped (discarded). If the computer is a router, the datagram will be forwarded to the next router in the path. In summary, a router will forward datagrams—a host will not forward datagrams.

A router knows where to forward an IP datagram based on its routing tables. The routing tables, in their simplest form, say: *if a datagram is destined to network X, then send it to the next router Y*. In fact, both a router and a host have a routing table. The table in the router may be quite complex (with many rows or entries), whereas in a host it is usually just a single entry specifying the default router (or have we have seen, default gateway—gateway and router are the same in this context).

For a router, the routing tables are created using routing protocols. The routing protocols are implemented as software applications. During network operation, the routers in the internet communicate with each other to discover the best paths through the internet. Alternatively, the routing tables can be created manually by adding entries to the routing table.

Figure 1 summarises the differences between routers and hosts.

### 3.2 Enabling Routing

There is only a small difference in functionality between a router and host (however, for a real network, there may be big differences in implementation and performance: for example, a commercial router often has hardware and an operating system dedicated to the task of routing, whereas a PC uses general purpose hardware and operating systems). In practice it is easy to make a host become a router. That is, most PCs can be configured as a router if:

1. They have two or more network interfaces (as the PCs in the Lab do)
2. The operating system is configured to enable *forwarding*

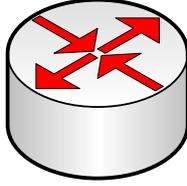
	Host	Router
		
Network interfaces	Usually 1	2 or more
Source of packets	Yes	Not common
Destination of packets	Yes	Not common
Forwards packets	No	Yes
Routing table	Yes Usually simple; 2 or 3 entries	Yes Usually complex; 100's or 1000's of entries
Routing protocol	Sometimes If so, simple protocol	Yes Complex (BGP, OSPF, RIP)

Figure 1: Comparison of Router and Host

On Ubuntu Linux, if the following file contains a 0 then forwarding is *off*, if it contains a 1 then forwarding is *on*:

```
/proc/sys/net/ipv4/ip_forward
```

By default, forwarding is *off*. You turn it on by using a text editor to change the value in the file from 0 to 1.

On Microsoft Windows XP, in Registry Editor, navigate to the following registry key:

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters
```

Select the following entry:

```
IPEnableRouter: REG_DWORD: 0x0
```

To enable IP forwarding for all network connections installed and used by this computer, assign a value of 1.

### 3.3 Editing the Routing Table

In large internets, routing protocols are used to automatically create the routing tables. In small internets, we can manually configure the routes. To do this, we must add routes to the routing tables.

In Ubuntu, the `route` command shows the current routing table. Usually, for a host there will be a single entry like:

Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
default	192.168.1.1	0.0.0.0	UG	0	0	0	eth1

This says, for all IP datagrams that are destined outside of this network, send them to the default router (192.168.1.1).

And for a simple, two interface router, there may be more entries to specify the routes to different networks. In order to add a route, you can use the `add` option:

```
route add -net NETWORKADDRESS netmask SUBNETMASK gw NEXTROUTER dev INTERFACE
```

where:

- `NETWORKADDRESS` is the IP address representing the destination network, e.g. 192.168.1.0
- `SUBNETMASK` is the subnet mask for the destination network, e.g. 255.255.255.0
- `NEXTROUTER` is the IP address next router in the path to the destination network, e.g. 192.168.3.1
- `INTERFACE` is the network interface to send the datagram on, e.g. eth2

Similarly, you can delete an entry with:

```
route del -net NETWORKADDRESS netmask SUBNETMASK
```

In Microsoft Windows XP, the `route` command can be used in a similar manner (although some of the options differ).

## 4 Network Design

When designing an IP based network there are many options to consider. Below are some of the options that you should consider (and document) during the tasks:

- LAN/WAN technology and appropriate cabling. What technology/cabling do you use and why?
- IP address assignment. What IP addresses (and associated net masks) are assigned to interfaces?
- Route selection. What entries are needed in the routing tables?
- Node configuration. What other configuration changes did you make to the hosts and/or routers?
- Test cases. What do you do to make sure the network is working correctly? What are the results of the tests?

Some of the design information can be shown on a diagram, similar to that in Figure 2. For example, you could add the LAN/WAN types and IP address information to the figure. Other information should be described or given in tables.

Note that it is often convenient to refer to a network interface by a short label, such as *IF1* of Router C, instead of the full IP address (and netmask). *IF* means *InterFace*. An alternative label you can use is the operating system label such as *eth0* or *eth1*. Then the mapping between interface and IP address can be given in a separate table.

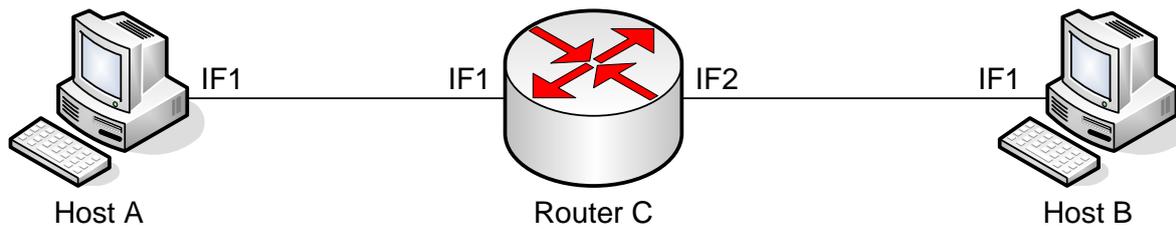


Figure 2: Single Router Network

## 5 Tasks

**Task 1.** Setup an internet that has two hosts and one router, as in Figure 2. Test the network using ping between hosts, as well as web access, while capturing the traffic at the router. Document the network design.

**Task 2.** Setup an internet that has two hosts and two routers, as in Figure 3. Test the network using ping between hosts, as well as web access, while capturing the traffic at the router. Document the network design.

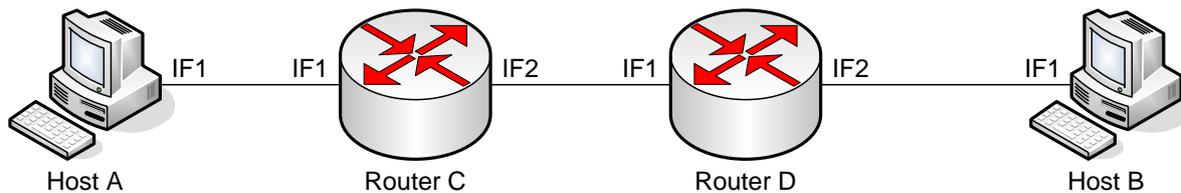


Figure 3: Two Router Network

## 6 Cleaning IP

As other students (both in this lab class and other classes) must use these computers, it is very important you set the computer configuration back to the default. This includes:

- Forwarding is turned off.
- Routing tables are set to their default.
- Both interfaces should be configured to use DHCP.
- Ethernet cables connected from each network card to the appropriate switch.
- All personal files deleted from the computer (e.g. source code, results).

Failure to leave your computer in an appropriate configuration may lead to a loss of marks. Leaving source code and results on the computer may result in penalties for cheating!

## A Notes

Record any additional notes here (e.g. important points made by the instructor, summary of things you learned, mistakes you made). You should use this in future labs, as well as in preparation for assessment items like exams.