

Department of Computer Science
University of Pretoria

Computer Networks
COS 332

Study Guide (Practical Assignments)
Version 2024.01

2024

Contents

| | | |
|----------|-----------------------------------------|-----------|
| A | Some general requirements | 3 |
| B | Finding a host for your servers | 4 |
| B.1 | Servers and clients | 4 |
| B.2 | Options | 5 |
| B.2.1 | Standalone Unix | 5 |
| B.2.2 | Unix from the Microsoft store | 6 |
| B.2.3 | Virtual machines | 6 |
| 1 | Practical assignment 1 | 8 |
| 1.1 | Background | 8 |
| 1.2 | Your assignment | 11 |
| 1.3 | Assessment | 13 |
| 2 | Practical assignment 2 | 14 |
| 2.1 | Background | 14 |
| 2.2 | Your assignment | 15 |
| 2.3 | Assessment | 17 |
| 3 | Practical assignment 3 | 18 |
| 3.1 | Background | 18 |
| 3.2 | Your assignment | 20 |
| 3.3 | Assessment | 21 |
| 4 | Practical assignment 4 | 22 |
| 4.1 | Background | 22 |
| 4.2 | Your assignment | 23 |
| 4.3 | Assessment | 23 |
| 5 | Practical assignment 5 | 24 |
| 5.1 | Background | 24 |

| | | |
|-----------|-------------------------------|-----------|
| 5.2 | Your assignment | 25 |
| 5.3 | Assessment | 27 |
| 6 | Practical assignment 6 | 28 |
| 6.1 | Background | 28 |
| 6.2 | Your assignment | 29 |
| 6.3 | Assessment | 30 |
| 7 | Practical assignment 7 | 31 |
| 7.1 | Background | 31 |
| 7.2 | Your assignment | 31 |
| 7.3 | Assessment | 32 |
| 8 | Practical assignment 8 | 33 |
| 8.1 | Background | 33 |
| 8.2 | Your assignment | 33 |
| 8.3 | Assessment | 35 |
| 9 | Practical assignment 9 | 36 |
| 9.1 | Background | 36 |
| 9.1.1 | Layer 3 firewalls | 36 |
| 9.1.2 | Layer 7 firewalls | 38 |
| 9.2 | Your assignment | 39 |
| 9.3 | Challenges | 40 |
| 9.4 | Assessment | 41 |
| 10 | Practical schedule | 42 |

Chapter A

Some general requirements

Assignments need to be uploaded to clickUP by the due date. However, marks will only be awarded after the software has been demonstrated by its author(s). Dates of these demonstrations will be communicated in due course.

All programming assignments should be demonstrated as finished products: While you should be able to show your code on request, the development environment should normally not be visible during a practical demonstration. Do not hardcode any facets of your program that may have to change if you demonstrate it in a different environment (such as mail servers, IP addresses, network masks, etc.)

You are only allowed to use code that you found on the Web and elsewhere if the licence conditions of that code allows you to use it. If you use such code, it should be clearly documented that you used the code (as well as where you got it from). **Furthermore, that code should not provide the core functionality of the program you are submitting — in this course this means that such code should not provide any data communications functionality at all.**

Note that plagiarism and/or violation of copyright are serious offences. We accept that you are familiar with the remarks about plagiarism as set out in Part B of the COS332 study guide and the University Regulations in general.

All the assignments in this guide may be completed by an individual student or as a team consisting of two students. When two students collaborate, they are allowed to upload identical submissions. The names of both team members should be stated as part of the code they uploaded. Both should participate in the single demonstration of a given assignment. Under normal conditions both team members will be awarded the same mark.

Chapter B

Finding a host for your servers

B.1 Servers and clients

Network services are provided by servers that are accessed by clients. In some cases servers talk to one another in order to complete a request.

The term *server* is used with two distinct meanings. On the one hand a server may be a computer which runs the necessary software to service requests. On the other hand a server may be the software running on such a computer. Where necessary, we will refer to the computer as a *host* and the specific software as a *dæmon* (though both terms need further disambiguation to use them correctly).

To illustrate, consider a company who buys a computer to use as ‘server’ and then installs software on it to serve web pages, handle email and act as an FTP repository. They may refer to this computer as the ‘server’. The software that handles web requests will often be `httpd`, where the `http` part indicates that it deals with the HTTP protocol, and the `d` suffix indicates that it is the dæmon (server) for this protocol. Similarly, FTP requests may be handled by a program called `ftpd` — for similar reasons. On the other hand, the dæmon used to forward email may be `sendmail`, without any suffix to indicate that it is in fact a dæmon.

In this module you will be expected to develop software that, in some cases, work as dæmons to be used by specific client software. The client software will be readily available; since you will develop the dæmon, you will have everything you require to complete a working system.

In other cases you will be expected to write client software that should communicate with a dæmon. Unfortunately for this module (but, generally fortunately) organisations restrict access to their servers. Hence you will most probably need to supply your own server to complete a number of the assignments. This chapter is intended to guide you through the process of setting up your own server.

B.2 Options

A number of options exist to obtain such a server (computer). You may already have a Linux installation available. Or your computer may use a variant of Unix as (the foundation of) its operating system, that allows one to install server software. Microsoft Windows has traditionally been challenging in this regard: while server software is typically available, they often cost money, and it may be challenging to find information about them on the web.

Whatever solution you choose, verify that you are able to install well-known server software on your platform.

B.2.1 Standalone Unix

As already mentioned, your one option is to use a (standalone) Linux installation. (This includes the case of computers that are configured to dual boot Linux and some other operating system; to complete the assignments, such a machine should be booted as a Linux computer.)

One of the more exciting options is to use a Raspberry Pi as your server. Any Raspberry Pi would suffice for these assignments. In the case of the earlier versions of the Pi, it is better to get the B version — that is, a 1B, 2B, 3B or 4B — since these versions are, amongst others, intended to be used as general-purpose computers. The Pi 5 is only available as a model that is similar to the earlier B versions, so any Pi 5 will suffice.

The Pi 1 to 5 provides an Ethernet port, which dramatically simplifies connection between your client computer (such as your laptop) and your server (the Pi). Some versions of the Pi Zero include a WiFi interface, and with enough patience, it is not too hard to get your client computer to talk to the Pi Zero as a server. At the time of writing this, the Pi Zero W is available for about R310.

If you plan to buy a Pi, note that some shops sell Pi boards at exorbitant prices. As a general rule I recommend that you compare prices (and availability) of boards or kits that are of interest with offers on [pishop.co.za](https://www.pishop.co.za) — one of the four official distributors of the Pi in South Africa. As a computer science student the odds are that you will only need a board and a micro-SD card, since you probably have access to spare keyboards and other components. (For most of our work you won't even need a monitor for the Pi.) Note that typical cell phone chargers are sufficient to use as a PSU for most of the Pi computers. Note that you need a somewhat more powerful charger (2A) for the Pi 4B and an above-average one (3A) for the Pi 5.

The other authorised South African distributors are

- DigiKey (<https://www.digikey.co.za/>),

- Farnell (<https://export.farnell.com/>) and
- Mouser (<https://www.mouser.co.za/>),

where prices and stock levels are somewhat harder to determine.

B.2.2 Unix from the Microsoft store

Note that a number of Linux for Windows distributions are available in the Microsoft Store. The well-known ones are all free.

B.2.3 Virtual machines

A solution that is known to work well is based on Oracle’s VirtualBox virtual machine. Some additional details are therefore provided for this option.

In order to create your own server on VirtualBox you will need a computer and a disc. VirtualBox should be installed on the computers in the Informatorium. To install VirtualBox on your own computer, you can download it from <https://www.virtualbox.org>

One good option is to use the server edition of Ubuntu as the operating system of our virtual host. The system may be downloaded from various mirrors around the Internet. Expect a download of a few megabytes.

As noted, you also need a disc for your host. Any USB drive that can hold a file of about 16GB should work fine. (The contents of this drive will not be erased; a new set of files will simply be added.) Proceed to install the operating system on this removable disc after you created a new virtual machine with VirtualBox. Do not install any of the servers (daemons) that the installation process offers to install. Note that this installation process takes a few minutes that tend to feel longer than it is. Also be prepared to repeat the installation if necessary — such processes do not always proceed as planned. There are copious amounts of information available on the web that should guide you through any problems you encounter.

Be aware of the different uses of the term *host* in networking in general, and in VirtualBox. In networking a host is simply a computer connected to a network (possibly acting as a server). In VirtualBox, the computer that you create is known as a *guest*, which is ‘hosted’ on the physical computer. Hence, our Ubuntu computer will in VirtualBox be known as the guest, while the real computer (possibly running Windows) will be known as the *host*. Install your virtual computer with networking configured such that the host (Windows?) can talk to your Ubuntu computer. If possible, other real computers on the network should also be able to talk to your Ubuntu computer. For details, read more about “virtual networking” on VirtualBox. To repeat: In this context you want the host

to be able to talk to the guest. The guest should also be able to talk to the host, but that is not a challenge.

Note that, after you have set up your network adapter, we will revert to talking about our Ubuntu computer as the host.

Once your installation is complete, you should be able to boot your virtual host, and log in. To shut it down, simply enter the following command:

```
sudo shutdown -h now
```

To boot your host on another (physical) computer you may have to create a new virtual machine, pointing to the disc you already have. It's best to try it early in the process to be certain that you can retrieve your host whenever required.

Chapter 10

Practical schedule

Practical assignments have to be uploaded to clickUP by the due date. The assignment then has to be demonstrated during the indicated practical sessions to earn marks.

The practical assignments are due on the following dates:

Wednesday, 6 March 2024 Assignment 1

Wednesday, 3 April 2024 Assignment 2

Wednesday, 10 April 2024 Assignment 3

Wednesday, 17 April 2024 Assignment 4

Wednesday, 24 April 2024 Assignment 5

Wednesday, 1 May* 2024 Assignment 6

Wednesday, 8 May 2024 Assignment 7

Wednesday, 22 May 2024 Assignment 8

Wednesday, 29 May 2024 Assignment 9

Assignments are due at 13:00 on the days noted. In all cases an automatic extension of 24 hours will be granted. That means the assignment will be accepted for another 24 hours without any penalty. However, beyond that, assignments are too late to be accepted. (Note, due to technical limitations, this means that clickUP will display the due date and time for Assignment 1 as 7 March 2024 by 13:00. In the lecturer's mind it is actually due on 6 March 2024 by 13:00.)

Since 1 May is a public holiday, submissions will be accepted until 22:00 on 2 May without penalty.

Submissions that are due on a Wednesday are to be demonstrated during the practical period on the next Friday (10:30–12:30) *or* the immediately following Tuesday (14:30–17:30).

Demonstrations not only involve showing what your software does, but also include answering some questions about reasons you decided to solve a problem in some particular way, or about the way an aspect of your program works. Answers to such questions may affect your mark. (In practice, many students ‘volunteer’ the expected information during the demonstrations, such that additional questions about the work are not necessary.)

These tight schedules explain why extensions cannot be granted beyond the automatic 24-hour extension.

You are allowed to upload a given assignment any number of times and only the final submission will be deemed to have been submitted. Rather submit work as soon as it is almost ready, and then again before the due date, than missing the opportunity to upload your work.

When two students collaborate on a project, both have to be present simultaneously to demonstrate their project.