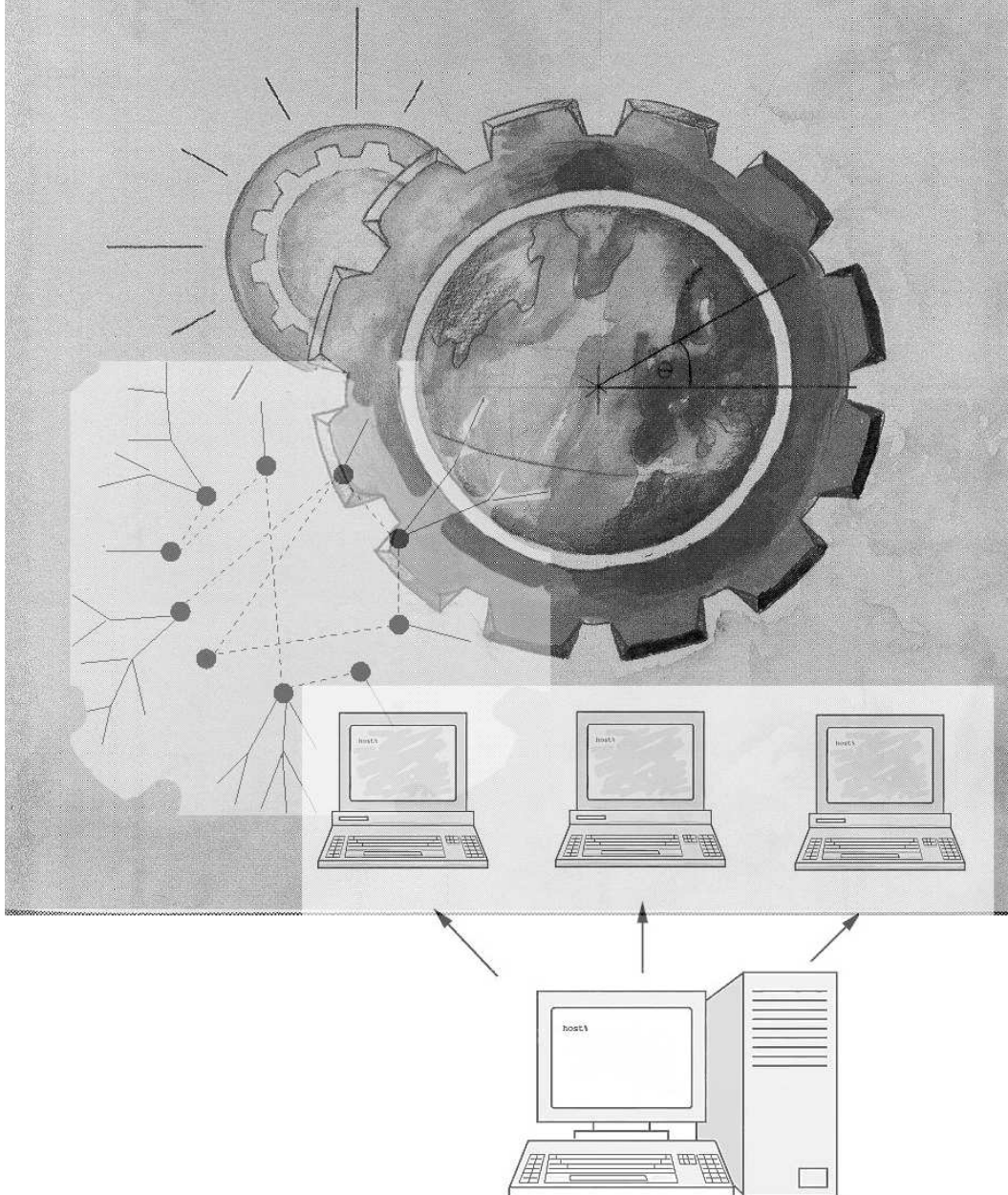

MS009A

Network Infrastructure and Security Lab



Network Infrastructure and Security Laboratory Workbook

Version 3.0

Oslo University College, Norway

Mark Burgess

January 15, 2007

Contents

I	Introduction	5
1	Introduction	7
1.1	Aims and requirements	7
1.2	Planning	8
2	Safety and responsibility in the Lab	9
2.1	General Laboratory Safety Rules	9
2.2	Electrical Safety Guidelines	10
2.3	Electrical Emergency Response	11
2.4	Handling equipment	11
3	When entering and leaving the laboratory	15
3.1	Planning your time	15
3.2	Sharing the lab	15
3.3	Resources	15
3.4	Security: the key	16
4	Documenting your lab-work	17
4.1	Keep a notebook	17
4.2	Plan your work	17
4.3	The main elements	18
4.4	Analysing results	19
4.5	Your grade	19
5	Plotting graphs	21
5.1	Xmgrace	21
5.2	Adding labels and legends	22
5.3	Trouble-shooting	24
5.4	Command summary	24
6	Troubleshooting	25
II	Mandatory Experiments	27
7	Host configuration management and web server	29
7.1	Setting up a virtual Unix server farm	29
7.1.1	The setup	29
7.1.2	The login process	30
7.1.3	Installing Apache	31
7.1.4	Important configuration files	32
7.1.5	Configuring Apache	32
8	Network Basics	35

8.1	Equipment	35
8.1.1	Router capabilities	35
8.1.2	Cable types	35
8.1.3	Quick configuration	36
8.1.4	Catalyst switches	36
8.1.5	Nortel BayStack switch	37
8.1.6	Juniper routers	37
8.2	Connecting computers together	38
8.2.1	Connect two PCs together using a direct Ethernet to Ethernet connection . . .	38
8.2.2	Connect two PCs together by a switch	38
8.2.3	Showing that broadcast traffic is unrestricted with a basic switch setup	39
8.2.4	Splitting our 3 PCs into 2 connected VLANs with Cisco	40
8.3	Security features	41
8.4	Remote connectivity	41
8.5	Creating VLANs with the Nortel-Juniper collaboration.	43
III	Suggested In-depth Experiments	47
9	Advanced routing	49
9.1	RIP V1/V2	49
9.2	EIGRP - an improved RIP?	50
9.3	Routing loops	51
10	BGP Policies and Virtual Tunnels	53
10.1	Autonomous systems	53
10.2	Peering	54
10.3	Configuration errors	54
10.4	Equilibration	54
11	Traffic Shaping and Quality of Service	55
11.1	Scenario 1	56
11.2	Measurements	56
11.3	Scenario 2	57
12	Wireless Networking	59
13	VPN and IPSec	61
14	Virtual machines and large installation configuration	63
14.1	Virtual networks	63
14.2	Managing the network	63
15	Sensor monitoring	65
16	XEN Virtual machines	67
17	XORP router	69
18	Linux Terminal Server Project	71
19	High availability web services	73
20	Disk and RAID performance	75
21	SNMP and Network Operations Centre	77

22 Multiseat system Under Linux	79
23 Miscellaneous	81

Part I

Introduction

Chapter 1

Introduction

This booklet is your guide to the laboratory work for the Master's degree in Network and System Administration.

1.1 Aims and requirements

MS009A is designed to demonstrate and practise the concepts of design, configuration and maintenance of networked systems with a hands-on approach. The course is meant to foster *independent and scientific/diagnostic thinking*. You will work with the same hardware and development tools used by enterprise and service providers.

The course covers the four practical skills.

- System administration (installation, configuration management etc)
- Services (Web, ssh, etc.)
- Networking (routing, switching, etc.)
- Security (host and network)

You should aim to learn the following in this course:

- To be familiar with cables and equipment and how to connect network devices together.
- How to install operating systems and set up web services.
- How to take measurements, understand and present the results.
- Understand the meaning of mean, standard deviation and error bars.
- How to document experimental work.
- How to use virtual machines.
- How to connect switches and routers using RIP, OSPF.

Requirement 1 (Required knowledge). *You should be familiar with the coursework in Network and System Administration I, i.e. the contents of ref. [?]. If you have not taken this course or are not familiar with this material, you must perform the exercises for installing GNU/Linux and Web Services as part of your mandatory work. All students leaving our programme must be able to set up a web server.*

Requirement 2 (Lab notebook). *When working in the lab you must keep a notebook, e.g. spiral bound in which you write the details of what you are doing as you do it. Your notebook will be examined each week for your progress. It should be readable and carefully documented.*

If you want to include screen data in your notebook, you should glue or edit it in. You must deliver your notebook in hard copy. If you edit your notebook electronically, you need to keep a computer available at all times for noting the data.

You should not wait until after the experiment to make your notes. Notes are taken as you work. Do not trust your memory. From your notes it should be possible for someone to reproduce what you did.

In the lab, you will be required to complete a certain number of mandatory exercises which you should make notes about in your notebooks.

Requirement 3 (Reports). *You must deliver at least two full experimental reports for the in-depth experiments.*

1.2 Planning

In order to carry out the experimental work, you will need to develop your planning skills and your diagnostic (trouble shooting) skills. The experiments are deliberately described in little detail to make you devise your own plan to succeed. In each experiment ask yourself — how can I measure whether this is working? How can I test this?

You will work in pairs in the lab, but everyone must write an individual report. It is not acceptable to submit the same report as another student. The estimated time for each experiment is noted in the text, but this is not a fixed limit.

A handful of lectures will be given on special topics:

- General orientation.
- Setting up an experiment.
- Statistics and measurements.
- Troubleshooting methods.

This is not a book of recipes but a guide to help you to make your own decisions and investigations.

Chapter 2

Safety and responsibility in the Lab

Oslo University College is committed to providing a safe and healthy working environment for its students and staff, and to teaching safe and healthy practices in your later employment. You will learn working practices that you will take with you when you leave the College. It is your ethical duty to spread safe working practices and procedures beyond this College in the workplace.

Oslo University College requires its students, staff, and faculty to adhere to basic General Laboratory Safety Rules in the Electrical Engineering laboratories.

These rules apply to everyone. If you see someone ignoring the rules, it is your duty to remind them of their responsibilities.

This discipline is a mark of professionalism – it is not for annoyance or for show. You might find other labs that do not care about safety, but as a student of Oslo University College, we expect the highest standards.

Requirement 4 (Accept terms). *You must sign the terms and conditions for working in the laboratory before you can use the equipment.*

2.1 General Laboratory Safety Rules

1. The lab is a dry and noisy environment. You should try to organize your work so that you do not spend more than three or four hours in the lab at any one time. If you need to talk to your work partners, go out of the lab into a quiet area rather than shouting.
2. Food and drink must not be brought into, stored in or consumed in a laboratory.
3. Never wear personal headphones or play music to drown out the noise: you can damage your hearing by playing loud music, and cause accidents by being distracted. Your full attention should be on your work.
4. Shoes must be worn, and appropriate personal clothing shall be worn in laboratories.
5. Do not wear items of metal jewelry when using electronic equipment. You might cause a short-circuit, get a shock or get the jewelry caught in something.
6. Bags and coats should be left at the doorway.
7. *BE TIDY AND TREAT ALL EQUIPMENT WITH RESPECT.* Do not tread on or run over cables with chairs or heavy objects.
8. Never drop items on the floor. Always put equipment in the correct box or storage place provided for it. Fold cables neatly to avoid long-term damage to them.
9. Learn and know what to do in an emergency.

10. Do not expose yourself or others to risk. Do not joke around in the lab.
11. Un-authorized persons **MUST NOT** be admitted to the laboratory.
 'Authorized' means having business in the laboratory with the permission of the College. No-one under the age of eighteen may enter the lab unsupervised. Only trained individuals and students enrolled in courses using the lab may touch the equipment.
12. Laboratory shall remain locked when unoccupied.
13. Never open (remove cover) of any equipment in the laboratories, unless you are instructed to do so.
14. Report all problems to drift@iu.hio.no (extensions 3268/3264/3263)

2.2 Electrical Safety Guidelines

1. Be familiar with the electrical hazards associated with your workplace.
2. Be as careful for the safety of others as for yourself. Think before you act. Be tidy and systematic.
3. Avoid bulky, loose or trailing clothes. Avoid long loose hair. Remove metal bracelets or watch-straps.
4. Do not take food or drink into the laboratory. Avoid wet hands and clothing.
5. Use extension cords (skjøteledninger) only when necessary and only on a temporary basis. Request new power outlets if your work requires equipment in an area without an outlet.
6. **ALWAYS** report damaged cords and cables, including cables that become hot, or with exposed wiring. Never ignore these or replace them for others to use.
7. Experiments left unattended should be isolated. If for a special reason, it must be left on, a barrier and a warning notice are required.
8. Equipment found to be faulty in any way should be reported immediately and not used until it is inspected and declared safe.
9. Voltages above 50 V RMS AC and 120 V DC are always dangerous. Extra precautions should be considered as voltage levels are increased.
10. Never make any changes to circuits or mechanical layout without first isolating the circuit by switching off and removing connections to supplies.
11. Know what you must do in an emergency.
12. Be acquainted with the Emergency Power Off:
13. When touching equipment that might be electrically "live" always use the *back of your hand*. Never grip anything with your fingers, since an electric shock will tend to make your grip tighter.
14. Electrical power sockets should not be overloaded.
15. No cables should ever be stretched. Tape them firmly to a table or floor. Remember that people can trip over cables.
16. Hold the plug firmly by the insulating cover when plugging it in or unplugging it. **NEVER** unplug an electrical device by pulling on its cord!
17. Do not attempt to lift or move heavy equipment alone.

2.3 Electrical Emergency Response

The following instructions provide guidelines for handling two types of electrical emergencies:

When someone suffers serious electrical shock, he or she may be knocked unconscious. If the victim is still in contact with the electrical current, immediately turn off the electrical power source. If you cannot disconnect the power source, push in the Emergency Power Off button.

1. Do not touch a victim that is still in contact with a power source; you could electrocute yourself.
2. Have someone call for emergency medical assistance immediately. Administer first-aid, as appropriate.
3. If an electrical fire occurs, try to disconnect the electrical power source, if possible.
4. In case of fire, pull out power sockets if possible, taking care not to be electrocuted.
5. Use of **clean water** to put out electrical fire is acceptable.

2.4 Handling equipment

All electronic equipment should be treated as highly fragile and easily damaged, regardless of how sturdy it is. Today we are far too blasé towards electronic equipment.

- Never insert or remove power cords to equipment without ensuring that it is switched off.
- Take care when inserting multi-pin connectors that the pins are oriented the right way up and that no pins are bent on insertion.

Moreover:

- *Read instructions:* When dealing with hardware, one should always look for and *read* instructions in a manual. It is foolish to make assumptions about expensive purchases. Instructions are there for a reason.
- *Interfaces and connectors:* Hardware is often connected to an interface by a cable or connector. Obtaining the correct cable is of vital importance. Many manufacturers use cables which look similar, superficially, but which actually are different. An incorrect cable can result in damage to an interface. Modem cables in particular can damage a computer or modem if they are incorrectly wired, since some computers supply power through these cables which can damage equipment which does not expect to find a power supply coming across the cable.
- *Handling components:* Modern day CMOS chips work at low voltages (typically 5 volts or lower). Standing on the floor with insulating shoes, you can pick up a static electric charge of several thousand volts. Such a charge can instantly destroy computer chips. Before touching any computer components, earth yourself by touching the metal casing of the computer. If you are installing equipment inside a computer, wear a conductive wrist strap. Avoid wearing rubber sandals or shoes that insulate you from Earth when dealing with open case equipment, since these cause the body to build up charge that can discharge through that equipment; on the other hand this might be good advice when working around high voltage or current sources.
- *Disks:* Disk technology has been improving steadily for two decades. The most common disk types, in the workplace, fall into two families: ATA (formerly IDE) and SCSI. The original IDE (Integrated Drive Electronics) and SCSI (Small Computer Software Interface) had properties that have since evolved faster than the prejudices about them. ATA (formerly IDE) disks are now generally cheaper than SCSI disks (due to volume sales) and excel at sequential access, but SCSI disks have traditionally been more efficient at handling multiple accesses due to a multitasking bus design, and are therefore better in multitasking systems, where random access is important. However, filesystem design also plays an important role in determining the perceived performance

of each; i.e. how operating systems utilize buses during updates is at least as important as bus performance itself. Interesting comparisons show that IDE technology has caught up with the head start that SCSI disks once had[?] for many purposes, but not all.

SCSI[?] comes in several varieties, SCSI 1, SCSI 2, wide SCSI, fast-wide etc. The difference has to do with the width of the data-bus and the number of disks which can be attached to each controller. There are presently three SCSI standards SCSI-1, SCSI-2 and SCSI-3. The SCSI-2 standard defines also wide, fast and fast/wide SCSI. Each SCSI disk has its own address (or number) which must be set by changing a setting on the disk-cabinet or by changing jumper settings inside the cabinet. Newer disks have programmable identities. Disk chain buses must be terminated with a proper terminating connector. Newer disks often contain automatic termination mechanisms integrated into the hardware. The devices on the SCSI bus talk to the computer through a controller. On modern PCs the SCSI controller is usually connected to the PCI bus either as an on-board solution on motherboards or as a separate card in a PCI slot. Other buses are also used as the carrier of the SCSI protocol, like FireWire (IEEE 1394) and USB. The SCSI standard also supports removable media devices (CD-ROM, CD-R, Zip drives), video frame grabbers, scanners and tape streamers (DAT, DLT).

- *Memory*: Memory chips are sold on small pluggable boards. They are sold in different sizes and with different speeds. A computer has a number of slots where they can be installed. When buying and installing RAM, remember
 - The physical size of memory plug-ins is important. Not all of them fit into all sockets.
 - Memory is sold in units with different capacities and data rates. One must find out what size can be used in a system. In many cases one may not mix different types.
 - There are various incompatible kinds of RAM that work in different ways. Error correcting RAM, for instance, is tolerant to error from external noise sources like cosmic rays and other ultra short wave disturbances. It is recommended for important servers, where stability is paramount.
 - On some computers one must fill up RAM slots in a particular order, otherwise the system will not be able to find them.

Another aspect of hardware is the extent to which weather and environment are important for operation..

- *Lightning*: strikes can destroy fragile equipment. No fuse will protect hardware from a lightning strike. Transistors and CMOS chips burn out much faster than any fuse. Electronic spike protectors can help here, but nothing will protect against a direct strike.
- *Power*: failure can cause disk damage and loss of data. A UPS (Un-interruptible power supply) can help.
- *Heat*: The blazing summer heat or a poorly placed heater oven can cause systems to overheat and suddenly black out. One should not let the ambient temperature near a computer to rise much above about 25 degrees Centigrade. Clearly some equipment can tolerate heat, better than other equipment. Bear in mind that metals expand significantly, so moving parts like disks will be worst affected by heat. Increased temperature also increases noise levels that can reduce network capacities by a fraction of a percent. While this might not sound like much, a fraction of a percent of a Giga-bit cable is a lot of capacity. Heat can cause RAM to operate unpredictably and disks to misread/miswrite. Good ventilation is essential for computers and screens for avoiding electrical faults.
- *Cold*: Sudden changes from hot to cold are just as bad. They can cause unpredictable changes in electrical properties of chips and cause systems to crash. In the long term, these changes could lead to cracks in the circuit boards and irreparable chip damage.

2.4. HANDLING EQUIPMENT

- *Humidity:* In times of very cold weather and very dry heat, the humidity falls to very low levels. At these times, the amount of static electricity builds up to quite high levels, without dissipating. This can be a risk to electronic circuitry. Humans pick up charge just by walking around, that can destroy fragile circuitry. Paper sticks together causing paper crashes in laser printers. Too much humidity can lead to condensation and short circuits.

How could you go about verifying that conditions in the lab are appropriate?



Figure 2.1: Don't do this. Power connectors should not be hanging in mid air, as this means they are stretched and can come apart, causing an accident. The red network cables should not be tangled and hanging like this.

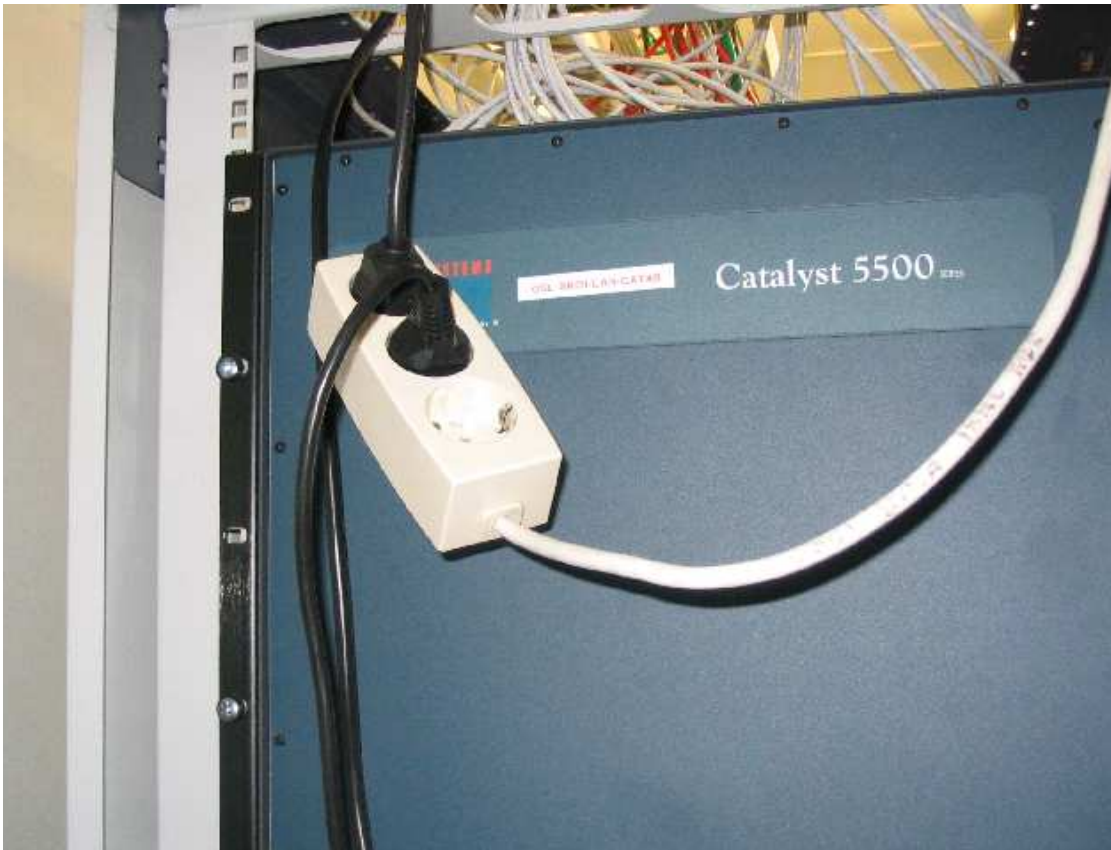


Figure 2.2: Dont do this. Again, a power connection is hanging in mid air. Cables are also hanging freely. Someone could walk into these and pull them apart.

Chapter 3

When entering and leaving the laboratory

3.1 Planning your time

Time in the lab is precious. Do not arrive without having planned what you are going to do. The lab environment is not conducive to thinking and planning—it is a stressful and noisy environment. You should do most of your thinking in advance. Other people could be using time that you waste.

3.2 Sharing the lab

In order to share the lab resources between many different groups, you should expect that:

- When entering the laboratory, none of the equipment will be in a pre-configured condition.
- After your time in the lab, all reasonable equipment should be returned to its original condition. Remove temporary cables and store them neatly. If you want to preserve any parts of your experiment, it is up to you to arrange it with the others on the course.

This presents you with a challenge. You should begin your lab term by devising a way of saving your configurations and setting up your experiments at short notice. *You must never save any configuration in the memory of the routers, or on the PC hard-disks.*

Solving this problem will give you a good working practice that will benefit you in a real working environment.

3.3 Resources

Cables need to last a long time. They can be damaged by running chairs over them, stepping on them or stretching them.

- Please keep cables neat and tidy. Coil them and sort them by their approximate length, so that others will be able to find them.
- Make sure that you know the difference between ordinary Ethernet cables and crossover (null-modem) cables. The latter are often colour-coded yellow. Are there any other colour codes or markings.
- Make sure that you know the difference between DCE and DTE serial cables.

Tools should never be removed from the lab – others will need them after you.

3.4 Security: the key

No one has general access to the lab. You have been given a key card to enter. Please do not lend this to others, under any circumstances.

Chapter 4

Documenting your lab-work

Different kinds of experiment warrant different kinds of report. You should aim to use your judgement in documenting work. It will take you some time to learn how to write good reports.

4.1 Keep a notebook

You should buy a (paper) notebook for recording a diary of your procedures, measurements and results from the lab. Keeping a tidy, legible notebook is important for your later understanding of what you did. Part of your grade will be based on how well you keep your notebook. If you later come to doubt your results, you will need to be able to read and understand the notes that you made, and perhaps reproduce the results again exactly.

4.2 Plan your work

The purpose of the lab experiments is two-fold:

- To develop your practical skills.
- To develop your analytical and reporting skills.

You should take your report writing seriously, since your grade will be based on this. Each experiment should be written up as a document, giving all details of your procedure and results. The document should have the form:

- **Describe the aim of your experiment:** write an introduction motivating the experiment, referring to any sources of information that you use explicitly or implicitly.
- **Describe the equipment and tools** to solve the experimental task. Explain why you make these choices and what the possible limitations are.
- **Describe the experimental setup:** How is the equipment set up and configured? Use diagrams and words to describe the motivate the configuration.
- **Describe the experimental procedure:** What is it that you actually do? How do you collect measurements?
- **Describe the results:** Use tables and graphs, along with qualitative discussions of what happens. Make sure that you collect enough data to support the conclusions that you will make.
- **Use of statistics to estimate uncertainty:** Statistics are a tool for quoting results with more certainty, and also estimating the level of uncertainty in the quoted results. Use statistics to present your results and analyze their reliability. Be careful and realistic about your figures, If

there is uncertainty in your measurements, describe it and estimate it. Never try to cover up uncertainty.

- **Interpret results and draw conclusions:** The final stage of an experiment is to interpret the data that you have collected and explain any anomalies in the results. What are the data telling you? How certain can you be of your conclusion?

4.3 The main elements

When reporting about an investigation, it is normal to use the following template:

- **Introduction:** An explanation of what the aim of the experiment is, in your own words and opinion. Since the experiments are very open-ended, you can choose to colour them with your own interests. Explain here what it was you were looking for and provide an overview to the reader.
- **Method:** In this part of the report you should describe exactly what you did. There should be enough detail to allow another person to be able to reproduce exactly what you did and get the same results.
- **Results:** This part of the report is where you document what actually happened. If you make measurements, you should make tables of them here and perform any statistical analysis here. Again, the purpose of documenting this is not to impress anyone, but to create a faithful record of what you have observed, together with the analysis you use to draw your conclusions. If you have written this up properly, then it should be easy to find any mistakes that might arise.
- **Conclusions and discussion:** You should be able to state what you have discovered in your experiment. In your conclusions you should address things like:
 - How effective was your method?
 - How reliable are your results?
 - What is the chain of cause and effect?
 - How certain can you be of your results.
 - What is the best hypothesis to explain what happened in the experiment.
 - Criticize your own work and say what you would have done differently under other circumstances.

Some notes on this:

- You do not have to take this four-fold list as a literal structure for your report. Sometimes it is useful to break up the report into subsections that are organized differently. Every part of your work should contain these four elements however.
- *The point of documentation is to show critical thinking. There is no right or wrong answer.*
- Using Latex for writing your report is recommended.
- Although some of you will be asked to produce corporate presentations in your future workplace, you should not write your reports here as corporate psychobabble. Here, we are interested in doing science, not in marketing gimmicks.

4.4 Analysing results

Analyzing results requires experience and a proper understanding of tools like mathematics and statistics. In industry, one can buy all kinds of programs that plot graphs and make nice pictures — but how many users of these programs actually understand what they are seeing?

The correct way to analyze results is to maintain a constant dialogue between *hypothesis* and *observation*, being critical at all times. Tools like statistics can be used to organize observations logically and test the reliability of our beliefs, but there is no substitute for scientific thinking.

- Tabulate raw data in appendices. Include your estimate of how much uncertainty there is in the values that you have observed.
- Are the data numerical or qualitative?
 - How should they be organized?
 - What information do they provide?
- What is the mean and standard deviation of the data?
- What is the distribution of data about the mean?

4.5 Your grade

Your final grade in this course will be based on several things:

- The quality of your documentation.
- How well you show scientific thinking (an enquiring mind, formulating clear hypotheses) and present your results in a critical way.
- How well you develop sound routines for working in the lab, including safety, tidiness and methodology. (You will be asked to present your ‘best practice’ working procedures to the class about half-way through the course – so you should be thinking constantly about how you can improve).

Plotting graphs

There are many tools available for plotting graphs. An especially powerful tool is the free software program **xmgrace** (formerly **xmgr**). This tool parses files of numbers and plots them in two dimensions. It generates output in JPEG, PNG and EPS formats, for easy inclusion into web pages, Word or Latex documents.

5.1 Xmgrace

The program has a graphical user interface and a command-line batch mode. Both of these are quite useful.

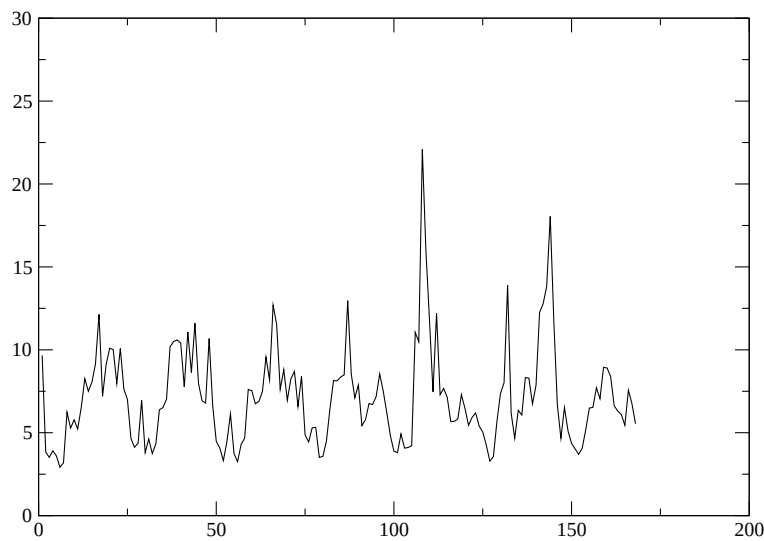


Figure 5.1: Figure without error bars.

For example, the following graphs in fig. 5.1 and fig 5.2 have been generated from a file of data that has this form:

1	9.669296	14.743795
2	3.836614	2.676340
3	3.513647	2.650355
4	3.907752	2.221279
5	3.606293	2.578484

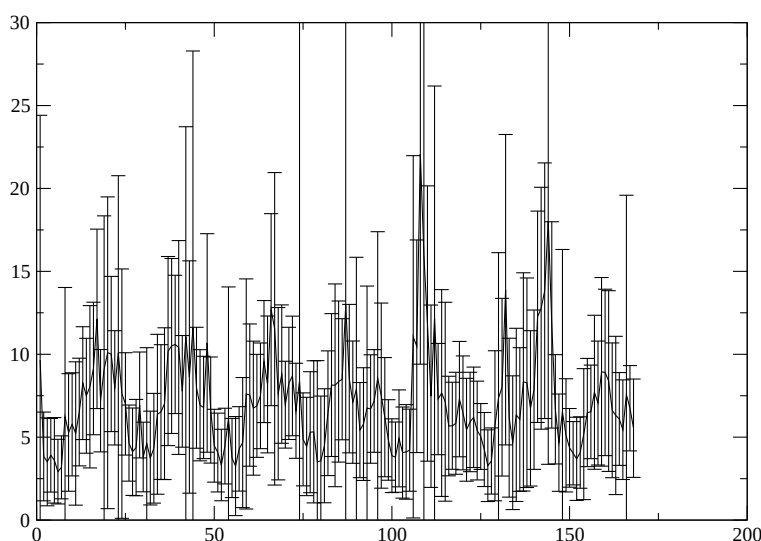


Figure 5.2: Figure with error bars.

```

6 2.923111 1.948955
7 3.180441 1.898489
8 6.231966 7.795706
9 5.284695 3.546003
10 5.774313 3.107895
11 5.222317 4.321972
12 6.515415 3.246366
13 8.258536 3.403739
14 7.499281 3.464971
15 8.041888 4.894231
...

```

The graphs were generated using the commands:

```

xmgrace -settype xy www-in.xy.cfenv -hardcopy -hdevice EPS
xmgrace -settype xydy www-in.xydy.cfenv -hardcopy -hdevice EPS

```

to generate Encapsulated Postscript files directly from the data, without using the graphical interface. They could also have been generated from the graphical interface by following the menus: Data, Import, ASCII, (SetType) etc.

Xmgrace reads several file formats.

```

X,Y
X,Y,DY
X,Y1,Y2,Y3...

```

i.e. a single line-graph with x, y points, a single line with error bars $x, y \pm dy$, or overlaid lines $x, y1, y2, \dots$

5.2 Adding labels and legends

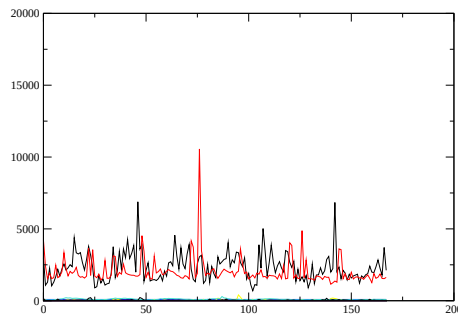
A graph is not complete without a caption to explain what it is showing, and labels on the axes. If your graph contains several lines, it should display a legend, which explains what the different lines are. The default is to plot lines in different colours; this can be changed for black and white printing.

For example, suppose we read in a multiline data-set:

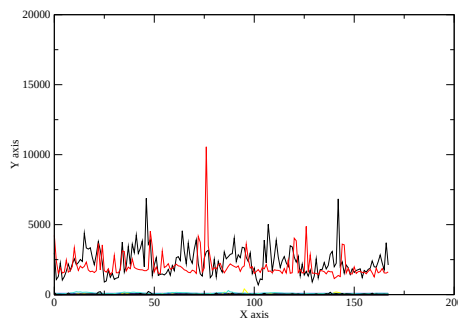
5.2. ADDING LABELS AND LEGENDS

```
0 11.520878 46.062277 63.216033 ...
1 12.733537 47.316163 70.035250 ...
etc
```

1. Start the program in graphical mode.
2. Select, Data, Import, ASCII.
3. In the dialogue box, remove the Filter “*.dat” to see all the files in a directory.
4. Select “Load as NXY”, select the filename and press OK. Press Cancel to get rid of the dialogue.
5. You should now see a graph like this:



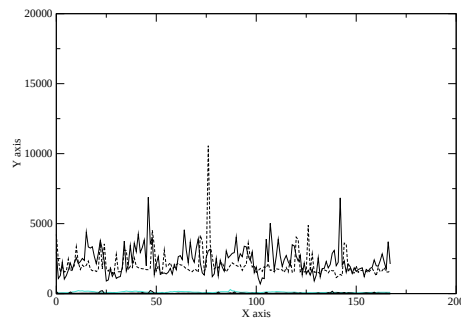
6. Select Plot, Axis properties and enter labels for the X and Y axes (press “Apply” for each change).
7. You should now see a graph like this:



8. To change from colour to black and white, select Plot, Set Appearance, and click on “Symbols”. Select a new colour and line style in “Line properties”. For each set in the “select set” dialogue, click and select a style. Press “Apply” for each change.
9. You should now see a graph like this:
10. To create a legend box, select Plot, Graph Appearance, Leg. box. (This does not seem to work in some versions.)

The difficult part about the legend box is finding it! It tends to fall off the end of the displayed area. You must specify coordinates where it will appear. Note that the viewport coordinate range is described on the “Main” tab of Graph Properties.

If you generate postscript file, you can edit the strings in the postscript for greater finesse.



5.3 Trouble-shooting

Problems: you might see the error message "Strings are not allowed". This means that a non-numerical string was found in the file, where a numerical value was expected. Check that no occurrences of "NaN" or "nan" have not come into the file, if you have generated the file from a program.

5.4 Command summary

```
host\$ xmgrace -nxy cfenv-averages
host\$ xmgrace rootprocs.cfenv
host\$ xmgrace -settype xydy rootprocs.cfenv
host\$ xmgrace -settype xydy rootprocs.cfenv -hardcopy -hdevice JPEG
```

Chapter 6

Troubleshooting

Trouble-shooting is a general skill. You can learn more about it in ref. [?]. In this course you will develop your trouble-shooting skills. A few hints can be borne in mind.

One of the important things to learn in the la is to not be afraid of handling the equipment. If something does not work, or if you make a mistake, you should be confident in your ability to fix the problem. If you are afraid of making changes, you will never master the equipment.

First of all, you should remember obvious things:

- Is it switched on?
- Am I logged in as supervisor (enable)?
- Is the cable crossed over or not?
- Is the cable damaged?

You can amplify on certain points yourself. For instance:

- Be aware that not all of the equipment components work perfectly. If something is not working, try changing the component for a different one.
- Are cables working or connected properly? Do you have the right kind of cable?
- Are port lights active? Is the port working?
- Are IP addresses set correctly?
- Have you allowed several minutes for switch ARP tables to be updated? (Only then will you be able to make a connection to a router.)
- Is there 'ping' connectivity?
- Use show/traceroute to see where problems arise.
- Verify that configuration commands actually succeed before assuming that they have.

See the example troubleshooting diagram below. You can formulate your own troubleshooting procedures.

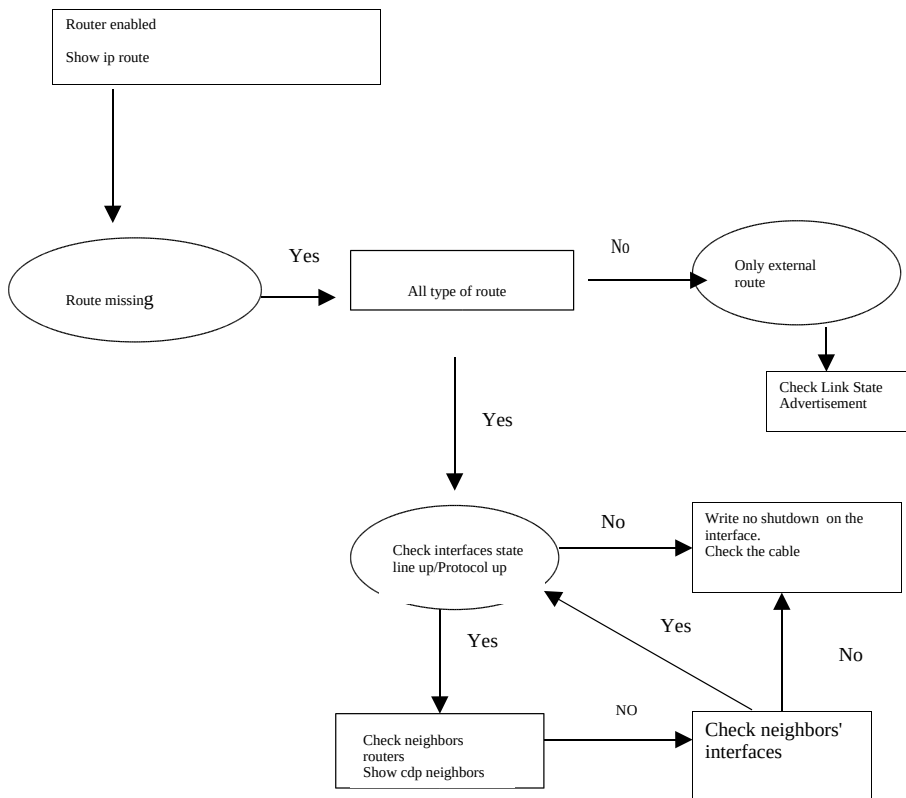
ROUTER TROUBLESHOOTING FLOWCHART

Figure 6.1: Example troubleshooting procedure for routers.

Part II

Mandatory Experiments

Chapter 7

Host configuration management and web server

Experiment 1. *The purpose of this experiment is to learn about setting up a Unix Apache web server, and running a farm of web servers using automatic configuration management cfengine.*

You will install apache first on a computer that has virtual machines with Linux pre-installed, and then you will use cfengine to set up and customize each web server automatically.

- **Estimated time:** *Two weeks*
- **Preparation:** *Install apache on GNU/Linux. Read about cfengine, apache, nano, sudo and Debian package management utility APT.*
- **Equipment:** *PCs and virtual machines.*
- **How to leave the lab after session:** *Virtual machines can continues to run. Physical machines will not be preserved, so make sure you complete Linux installation before leaving the lab.*

In this exercise you are asked to work step by step through some simple recipes for managing web farms. This is a skill that many employers need.

7.1 Setting up a virtual Unix server farm

7.1.1 The setup

In this experiment we will be using a Linux machine that hosts 4 virtual machines. We use virtual machines for practical reasons, but the setup as shown in figure 7.1 could also have been accomplished with physical hosts rather than virtual ones.

All the servers have cfengine installed on them. Server *one* will be our policy host and all the other servers including server *one* will be updating their configuration/policy through cfengine from server *one*. Apache will be installed on server *two*, *three* and *four*.

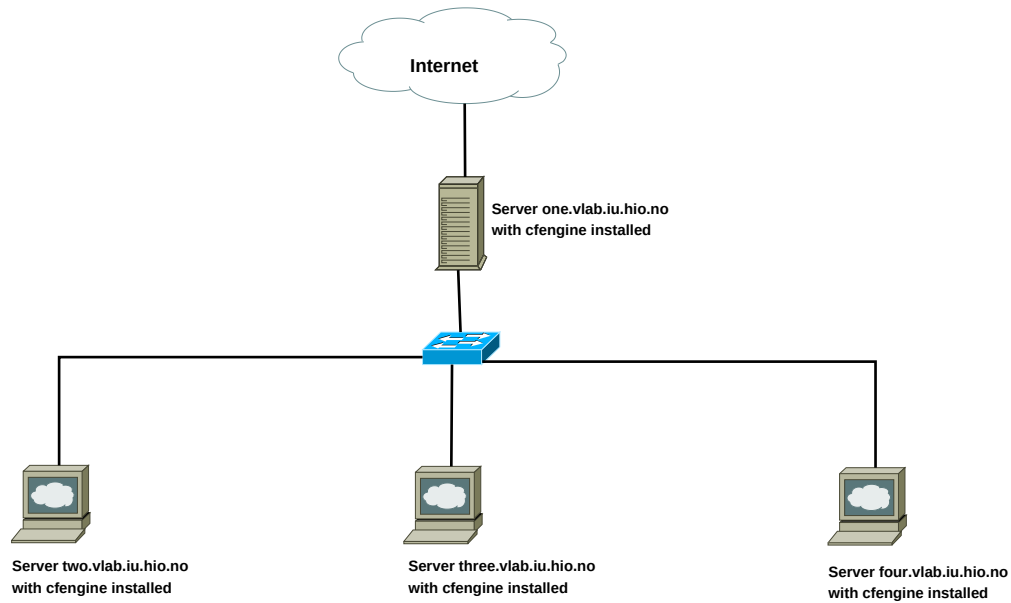


Figure 7.1: Web farm setup.

7.1.2 The login process

The first step is to login to the machines that will be hosting the web servers with the provided username and password. The machines are protected with a firewall that only allows access from our network. To login from another Linux host issue the command:

```
ssh username@host
```

(Where username is the one provided to you and host is the name or the IP address of the host). The machines are:

- two.vlab.iu.hio.no with the IP address 128.39.73.202
- three.vlab.iu.hio.no with the IP address 128.39.73.203
- four.vlab.iu.hio.no with the IP address 128.39.73.204

The first thing you will notice is that there are no graphics and you will have to do all your administration task on the command line. To edit files for example, you can use the editors nano and jed which are pre-installed on the hosts. If you prefer another editor then you will have to install it your selv. Read the next section **Installing Apache** on how to intstall software.

7.1.3 Installing Apache

The GNU/Linux distribution that is installed on these hosts is Ubuntu Dapper which is a Debian derived distribution. To install software on a Debian derived distribution, the package management utility APT(Advanced Packaging Tool) is used. To install Apache on the host you use the command:

```
sudo apt-get install apache2
```

After initiating the command you will be asked to accept the installation of apache2 and any other needed packages. Just answer yes and everything will be installed automatically.

```
Reading package lists... Done
Building dependency tree... Done
The following extra packages will be installed:
  apache2-common apache2-mpm-worker apache2-utils libapr0 libexpat1 libmagic1
  libpcre3 mime-support openssl ssl-cert
Suggested packages:
  apache2-doc lynx www-browser file ca-certificates
The following NEW packages will be installed:
  apache2 apache2-common apache2-mpm-worker apache2-utils libapr0 libexpat1
  libmagic1 libpcre3 mime-support openssl ssl-cert
0 upgraded, 11 newly installed, 0 to remove and 26 not upgraded.
Need to get 2761kB of archives.
After unpacking 9044kB of additional disk space will be used.
Do you want to continue [Y/n]? y
```

Thats it, the Apache server is now installed on your host. To test the installation, open a web browser on another machine and type:

- `http://two.vlab.iu.hio.no` or `http://128.39.73.202` (If `two.vlab.iu.hio.no` is where you installed the apache server, otherwise change two or the IP address with the appropriate name/IP)

If you see something similar to figure 7.2, then apache is installed and functioning. Install Apache on the remaining hosts. The next step is to configure Apache as needed.

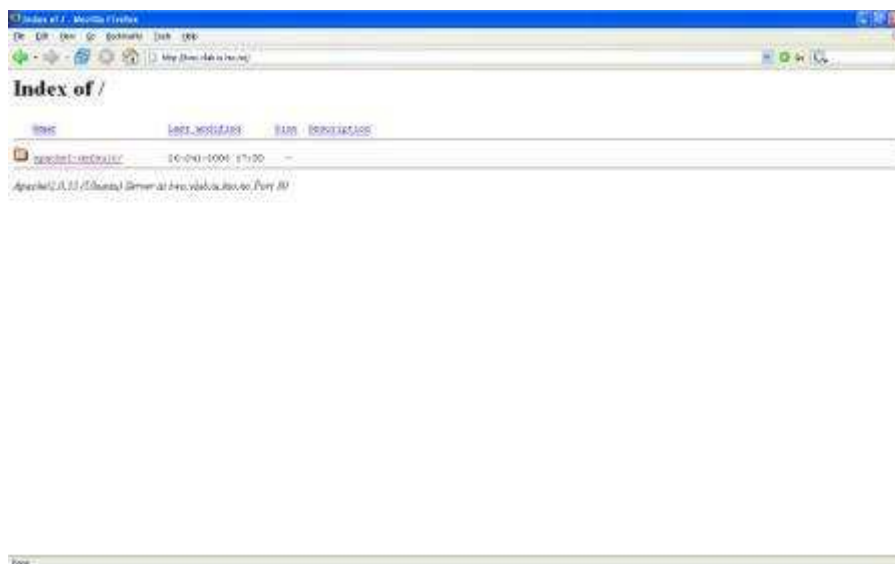


Figure 7.2: Apache running successfully.

7.1.4 Important configuration files

We will be configuring Apache through cfengine. The idea is to do all our configurations at one place by editing only one file and let cfengine do its magic of configuring our hosts.

Cfengine Files

Each host including server *one* has the following files:

```
/etc/cfengine/cfagent.conf (Main configuration file)
/etc/cfengine/update.conf (Simple file that let cfengine function incase of corrupted cfagent.conf file)
/etc/cfengine/cfservd.conf (Configuration file to run cfengine as a daemon)
```

What we want is a configuration/policy that will affect the whole site, therefore we will not edit these files, but rather edit similar files found in the directory:

```
/site/policy/
```

Apache Files

The apache main configuration file where most of our configuration will be done is located at:

```
/etc/apache2/apache2.conf
```

Apache also includes other configuration files which are found at:

```
/etc/apache2/ports.conf (For specifying which port Apache listens to)
/etc/apache2/sites-available/default (For configuring virtual hosts)
```

Default installation of Apache sets its DocumentRoot to the directory:

```
/var/www
```

This means that Apache will look for files and serve them only from that location. Apache can also serve files from a specified users directory (normally `www` or `public_html` under the users home directory), but this is disabled by default.

7.1.5 Configuring Apache

Server *one* is our policy host so login to server *one* with the provided username and password.

Enabling the UserDir option in Apache by using the editfiles command in cfengine

Edit the file `/site/policy/cfagent.conf` with your favorite editor and add the following line(s):

```
control:
  actionsequence = ( processes editfiles )
  EditfileSize   = ( 0 )

editfiles:
  two::
    { /etc/apache2/apache2.conf
      AppendIfNoSuchLine "UserDir public_html"
      AppendIfNoSuchLine "UserDir disabled root"
    }
```

7.1. SETTING UP A VIRTUAL UNIX SERVER FARM

Your `/etc/cfengine/cfagent.conf` will now look like this:

```
# cfagent.conf, created by the MLN cfengine plugin
control:
  actionsequence = ( processes editfiles )
  EditfileSize   = ( 0 )
processes:
  one::
    "cfserverd" restart "/usr/sbin/cfserverd"
editfiles:
  two::
    { /etc/apache2/apache2.conf
      AppendIfNoSuchLine "UserDir public_html"
      AppendIfNoSuchLine "UserDir disabled root"
    }
```

This will append `UserDir public_html` and `UserDir disabled root` at the end of the file `/etc/apache2/apache2.conf`. Examine the file `/etc/apache2/apache2.conf` and verify the actual editing has been conducted by cfengine. Then make the directory `public_html` under `/home/username` directory(Where `username` is your username). In this directory create an `index.html` file that contains some html code. Use the following commands:

```
cd ~
mkdir public\_html
cd public\_html
echo "<html>Test</html>" >> index.html
```

To test whether Apache is serving users files or not, open a web browser on another machine and type:

- `http://two.vlab.iu.hio.no/~username` or `http://128.39.73.202/~username` (Where `two.vlab.iu.no` is the server and `username` is your username)

If you see a page that shows `Test`, then Apache is configured correctly to serve pages from users directories.

Making sure that the Apache process is started

There could be times when a servers process stops or dies, thus making the service unavailable. Cfengine can start a process if it finds that the process is not running. We will configure cfengine so that it restarts the apache process if it discovers that the service is not running.

Edit the file `/site/policy/cfagent.conf` with your favorite editor and add the following line(s) under the `(processes:)` action:

```
any::
  "apache2" restart "/etc/init.d/apache2 restart"
```

Your `/site/policy/cfagent.conf` should now look like this:

```
# cfagent.conf, created by the MLN cfengine plugin
control:
  actionsequence = ( processes editfiles )
  EditfileSize   = ( 0 )
processes:
  one::
    "cfsservd" restart "/usr/sbin/cfsservd"
  any::
    "apache2" restart "/etc/init.d/apache2 restart"
editfiles:
  two::
    { /etc/apache2/apache2.conf
      AppendIfNoSuchLine "UserDir public_html"
      AppendIfNoSuchLine "UserDir disabled root"
    }
```

These will make sure that any host that updates its policy from the policy host and has Apache server installed on it, will get Apache started if the service is not running. You can test this by stopping Apache with the following command on the host running Apache:

```
/etc/init.d/apache2 stop
```

To verify whether Apache is running or not issue the command:

```
ps aux |grep apache
```

If you don't see any `apache2` processes, then Apache is not running. To make the `cfengine` configuration take effect immediately on the server running Apache, issue the command (on the server running Apache):

```
cfagent -v -IKq
```

This will run `cfagent` on the host and update its policy from the policy host. Then check and see if Apache has been started by `cfengine` by issuing the command:

```
ps aux |grep apache
```

If you see something similar to the following output, then `cfengine` has successfully started Apache. Otherwise check the main configuration file on the policy host for errors:

```
ps aux |grep apache
root      3552  0.0  1.7   7648  2352 ?        Ss   18:21   0:00 /usr/sbin/apache2 -k start -DSSL
www-data  3553  0.0  1.2   7440  1672 ?        S    18:21   0:00 /usr/sbin/apache2 -k start -DSSL
www-data  3554  0.0  1.5 228980  2100 ?        Sl   18:21   0:00 /usr/sbin/apache2 -k start -DSSL
www-data  3556  0.0  1.6 228980  2104 ?        Sl   18:21   0:00 /usr/sbin/apache2 -k start -DSSL
root      3612  0.0  0.3   1628   492 tty1    S+   18:21   0:00 grep apache
```

`Cfengine` is a very powerful program that lets you manage the configuration of a large number of servers. In this experiment, we have demonstrated a small portion of what `cfengine` is capable of. We covered for the most part actions/commands such as `processes` and `editfiles`.

The idea and aim is to give you a taste of the power of `cfengine` and give you the opportunity to further explore its capabilities. Please look into `cfengine`'s other actions/commands such as `files`, `shellcommands`, `copy`, `links`, `disable`. Use them in one form or another before concluding this experiment.

Chapter 8

Network Basics

Experiment 2. *The purpose of this experiment is to become familiar with the lab equipment and to build a number of network models, using especially the switches that are general tools for connecting PCs into a local area network. You will have need for these tools when building Ethernet networks in later experiments.*

In each case you should perform some kind of performance test and evaluate the setup procedure you carry out so as to compare the different solutions.

You must think of your own criteria for comparing the parts of the experiment. Place yourself in the role of a system administrator looking for the best solution for a large important network. Your ability to evaluate the results of your experiment could be vital for saving time and money.

- **Estimated time:** *Two weeks*
- **Preparation:** *Read about VLAN.*
- **Equipment:** *PCs and Catalyst switches*
 1. *Catalyst or Nortel switch*
 2. *Routing module or Juniper M7*
- **How to leave the lab after session:** *No configurations must be stored.*

In this exercise you are asked to work step by step through some simple recipes for using the configurable LAN switches and then answer some basic questions about LAN connectivity.

8.1 Equipment

8.1.1 Router capabilities

Make a brief summary of the different interfaces and capabilities of the routers and switches in the network laboratory.

8.1.2 Cable types

Make a brief summary of the different cable types used in the network lab, and what they are used for.

8.1.3 Quick configuration

1. Create scripts that can be loaded via `minicom` to blank out the router configurations and to set up new configurations. You can use the exercises from MS005A as test examples, if you like.
2. Make brief notes on your solution. We shall discuss these solutions with all groups in plenum.

8.1.4 Catalyst switches

The 5509 catalyst switch contain two separate switching fabrics, called Channel 1 and Channel 2, a supervisor engine that controls the switching behaviour and a routing module that is used to map VLANs to subnets. The 5513 switch does not have a routing module and hence cannot be used to create connected VLAN subnets unless we couple a router to it.

Switching Supervisor Engine

The first step is to establish the Console Port Connection from a PC workstation to the switch supervisor engine.

1. Access the console port using the console terminal "minicom" and a special cable S3-console. Be careful – some of the cables are crossed over and others are not. Silver "extensions" box-connectors can be used to crossover the cable and join cables together.
2. Turn ON the power to the switch and the console terminal.

After turning on the power to the switch and console terminal on systems with Supervisor EngineIII, you see this initial boot-up display:

System Power On Diagnostics

```
NVRAM Size.....128KB
LED Test.....Done
ID Prom Test.....Passed
DPRAM Size.....16KB
DPRAM Data 0x55 Test.....Passed
DPRAM Data 0xaa Test.....Passed
DPRAM Address Test.....Passed
Clearing DPRAM.....Done
System DRAM Memory Size.....16MB
DRAM Data 0x55 Test.....Passed
DRAM Data 0xaa Test.....Passed
DRAM Address Test.....Passed
Clearing DRAM.....Done
etc..
```

3. At the Enter password: prompt, press Return.
4. Enter privileged mode.

```
enable
```

5. At the Enter password: prompt, press Return.

Configuring a catalyst switch

The catalyst switches are programmable network switches. They can be made to behave as though they were several independent switches. An independent switch is called a VLAN, or virtual local area network.

For example, the first row of ports could be made into one VLAN and the second row could be made into another. Alternatively, the first five ports of the first row could be a VLAN and the rest of the switch could be another. You can program (almost) as many VLANs as you like.

The PCs in one VLAN form a broadcast domain. Computers in VLAN1 are all connected together and can communicate by ethernet, as usual. Computers in VLAN1 and computers in VLAN2 cannot talk to each other by default. They are disconnected, as if they were separate subnets. To connect them together you must either attach a cable from a port in VLAN1 to a port in VLAN2¹, or you must connect them *logically* using a router.

One of the catalyst switches has a router module that allows you to create virtual router interfaces between the VLANs. The larger switch does not have a router module, so its VLANs cannot be connected without the help of an external router.

To create VLANs on the switch, you must be in supervisor mode (enable). You use the command:

```
set vlan <number> <port list>
```

The ports do not have to be next to one another on the switch, so we have great flexibility of configuration.

8.1.5 Nortel BayStack switch

The Nortel BayStack 5510 switch is a programmable layer 3 switch that can be used to create VLANs, like the Cisco catalyst switches. Unlike one of the Catalysts, it does not have its own routing module; if you want to route between different VLAN subnets, you have to use the InterSwitch Connection protocol 801.1q., otherwise known as ‘tagging’.

The Nortel switch has twenty-four copper gigabit Ethernet ports and one optical gigabit Ethernet port (which replaces port 23). This can be used with an optical cable to connect it to the optical port on the Juniper M7 router. Together the Baystack and the Juniper M7 can work as a VLAN.

The Nortel switch has a web server interface, a menu based console interface and a command line interface. Take your pick!

8.1.6 Juniper routers

The lab has two Juniper J-series 2300 routers, each with two fast ethernet interfaces, and a Juniper M7i router with a single optical Gigabit Ethernet switch. This optical interface can be used to connect to the Baystack switch. Together the Baystack and the Juniper M7 can work as a VLAN.

The Juniper routers are modified FreeBSD machines, with specialized hardware. They have a web interface and a command shell, much like Unix. Using the command shell is very easy and is recommended. You can type “?” to get help or completions at any stage.

Connecting to the console, you log in as root and can then type normal Unix commands:

```
root@amboeba\% ifconfig -a
root@amboeba\% ls
```

etc.

The Juniper machines have a specialized shell called the Command Line Interface:

```
root@amboeba\% cli
root@amboeba\% ping 10.10.10.10
```

Once inside this, you can use “?” to get help. (See the example configurations below.)

The Juniper OS uses a version control system like CVS for updating configurations. When in configure mode, making changes to the Juniper configuration, nothing is changed until you type “commit”.

¹There is no point in doing this, since you could simply join the two VLANs together by reprogramming them into a single VLAN using the switch supervisor engine.

```
cli
configure
edit system root-authentication
set plain-text-password
set system host-name amoeba
commit
```

8.2 Connecting computers together

In this section we look at how computer are connected together by different kinds of hardware, starting with the simplest case of a single wire. We look at the most basic use of the switches: as a smart replacement for a multi-port repeater or hub. Each port receives only its own frames rather than seeing every Ethernet frame on the LAN.

8.2.1 Connect two PCs together using a direct Ethernet to Ethernet connection

Find a NULL modem (crossover) cable and connect two PCs directly by a cable. Set IP addresses and default routes to the machines using any IP address and network of your choice. Verify that the machines can ping one another.

8.2.2 Connect two PCs together by a switch

1. Log onto the switch in supervisor mode (enable)
2. Find two correct cables and connect the PCs to two of the switch ports (it can be any port)
3. Make sure that the lights on the interface are green
4. check the ports using the command

```
show port
show port x/y
show vlan
```

where x in the module and y is the port number.

5. Configure the PC interfaces so that they are on the same subnet
6. Remember to create a static route from one PC to the other

To configure a static route for each PC address in Linux, you might need to set a route table entry (depending on what version of linux you are using) use e.g.

```
route add 222.45.3.8 eth0
route add 222.45.3.9 eth0
```

and so on. Note that a default route is no good here, since we do not have a router. Modern operating systems should be able to handle this automatically.

7. Show that you can ping one PC from the other

8.2.3 Showing that broadcast traffic is unrestricted with a basic switch setup

1. Connect a third PC to the switch and make sure that you can ping it,
2. You might need to set a static route to each PC in order to be able to ping between arbitrary PCs.
3. Show that you can see the packets on all PCs if you ping to 222.45.3.255 (i.e. the broadcast address) (Note that older versions of tcpdump do not show ICMP traffic) All of these tasks so far are using the default: vlan 0. To make more virtual LANs we must create some logical networks in the router module.

To log onto the router from the switch supervisor engine, type

```
> session 7
```

since the routing module is on slot 7. We then look to see what virtual interfaces are defined on the routing module. On a regular router, there would be an Ethernet-IP mapping for each interface. In the routing module, we have such a pair for each *virtual interface*.

```
Router>
Router>
Router>en
Router#show ip arp
```

Protocol	Address	Age (min)	Hardware Addr	Type	Interface
Internet	10.10.43.1	-	00d0.d3bf.cc00	ARPA	Vlan43
Internet	10.10.44.1	-	00d0.d3bf.cc00	ARPA	Vlan44
Internet	127.0.0.2	-	00d0.97e9.93ff	ARPA	Vlan0
Internet	192.16.20.60	-	00d0.d3bf.cc00	ARPA	Vlan1

Now get rid of vlan1

```
Router#config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#no interface vlan1
Router(config)#no interface vlan43
Router(config)#no interface vlan44
CTRL-Z
```

Now we have only the default vlan0

```
Router#show ip arp
```

Protocol	Address	Age (min)	Hardware Addr	Type	Interface
Internet	127.0.0.2	-	00d0.97e9.93ff	ARPA	Vlan0

We cannot remove this vlan - it is like having no VLANs at all. i.e. we are now using the switch as a "smart hub" with all ports as a single LAN.

NOTE: when creating VLANs in the Catalyst switch, you need to define the VLAN first in the switch before you can use them as virtual interfaces in the router module.

8.2.4 Splitting our 3 PCs into 2 connected VLANs with Cisco

We will now place 2 of the three PCs on one vlan and the third on the other. Once this is done we will have separated the broadcast domains so that a broadcast on one VLAN does not reach the PC on the other Vlan.

A router is needed to connect together the VLANs. We configure one virtual interface on the router for each separate vlan. These have device names vlan1 vlan2 ... vlan1023. We must also configure an IP address for the router on each virtual interface.

Note: the following does not work:

```
Router(config-if)#ip address 222.45.3.1
% Incomplete command.
Router(config-if)#ip address 222.45.3.1 255.255.255.0
Router(config-if)#interface vlan2
Router(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan2, changed state to d
Router(config-if)#ip address 222.45.3.2 255.255.255.0
222.45.3.0 overlaps with Vlan1
Router(config-if)#
```

0

This is not allowed because the VLANs have to be on different subnets. This is because the router administers the separation between the VLANs, and its software is designed to separate traffic on subnet boundaries.

Question 1. *If a router can only make VLANs out of different subnets, what is the point of VLANs? i.e. if the different VLANs are really different subnets, what is the point of having a new concept for VLANs? Hint: how many switches and router interfaces do you need to make separate subnets without vlan technology? Is there new functionality? Is there a cost saving?*

Here is an example where we create two non-overlapping subnets on VLAN channels 1 and 3.²

```
Router>en
Router#conf t
Enter configuration commands, one per line. End with CNTL/Z.

Router(config)#interface vlan3
Router(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan3, changed state to down
Router(config-if)#
Router(config-if)#ip address 120.45.3.129 255.255.255.128
120.45.3.128 overlaps with Vlan2
Router(config-if)#exit

Router(config)#no interface vlan2
Router(config)#ip address 120.45.3.129 255.255.255.128
%LINK-5-CHANGED: Interface Vlan2,
Router(config)#

Router(config)#interface vlan3
Router(config-if)#ip address 120.45.3.129 255.255.255.128
Router(config-if)#exit
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by vty0 (127.0.0.2)
Router#sh ip arp
Protocol Address Age (min) Hardware Addr Type Interface
```

²Notice that we change addresses to a class B base network (120.45.x.x). The router software did not allow us to split a class C network into two subnets!

8.3. SECURITY FEATURES

```
Internet 120.45.3.1      - 00d0.d3bf.cc00  ARPA  Vlan1
Internet 127.0.0.2      - 00d0.97e9.93ff  ARPA  Vlan0
Internet 120.45.3.129   - 00d0.d3bf.cc00  ARPA  Vlan3
```

Remember to make sure that the new virtual interfaces are administratively "up", by typing "no shutdown" in the router configuration. The command "show ip interface brief" is also useful for a quick summary.

Now connect the PCs to the switch ports, and configure them. On the lower subnet:

```
ifconfig eth0 120.45.3.8 netmask 255.255.255.128 broadcast 120.45.3.127
```

and on the higher subnet:

```
ifconfig eth0 120.45.3.130 netmask 255.255.255.128 broadcast 120.45.3.255
```

After it works, you should see something like this:

```
router#sh ip arp
Protocol Address      Age (min)  Hardware Addr  Type   Interface
Internet 120.45.3.1          - 00d0.d3bf.cc00  ARPA   Vlan1
Internet 120.45.3.8          0 0060.085b.65bd  ARPA   Vlan1
Internet 120.45.3.9          2 000b.db4b.3e08  ARPA   Vlan1
Internet 127.0.0.2          - 00d0.97e9.93ff  ARPA   Vlan0
Internet 120.45.3.129     - 00d0.d3bf.cc00  ARPA   Vlan3
router# ping 120.45.3.8
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 120.45.3.8, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms

```
router# ping 120.45.3.9
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 120.45.3.9, timeout is 2 seconds:

!!!!

8.3 Security features

Find out about how to restrict MAC addresses that can connect to a port.

8.4 Remote connectivity

To configure the in-band Interface (Telnet Connectivity)

1. If you are using a local network connection to the console port, set the logical port sc0.

Assign the Catalyst 5000 switch IP address to a VLAN The switch is assigned to the same VLAN as the one containing its IP address

```
set interface sc0 up
set interface sc0 [vlan] [ip_addr [netmask [broadcast]]]
```

2. Configure static routes.

```
set ip route destination gateway [metric] [primary]
```

3. Configure a default route, if desired.

```
set ip route default gateway [metric] [primary]
```

4. Check the configuration status of the switch.

```
show interface
```

5. Display the route table entries of the configuration.

```
show ip route
```

```
Fragmentation   Redirect   Unreachable
-----
enabled         enabled    enabled
```

The primary gateway: 192.16.20.1

Destination	Gateway	RouteMask	Flags	Use	Interface
default	192.16.20.1	0x0	UG	2	sc0
192.16.20.0	192.16.20.50	0xffffffff00	U	89751	sc0

```
Console> (enable) show interface
Console> (enable)
```

After entering the set interface sc0 up command and the set ip route command, you see this display:

```
Console> (enable) set interface sc0 up
Interface sc0 administratively up.
Console> (enable) set interface sc0 192.16.20.50 255.255.255.0
Interface sc0 IP address and netmask set.
```

```
Console> (enable) show interface
Console> (enable) show ip route
D-Checking Port Capabilities
```

The show port capabilities command allows you to determine the capabilities of the modules and ports in a switch. This example shows you how to show the port capabilities for module 6, port 1:

```
Console> show port capabilities 6/1
Model          WS-X5225R
Port           6/1
Type           10/100BaseTX
Speed          auto,10,100
Duplex         half,full
Trunk encap type 802.1Q,ISL
Trunk mode     on,off,desirable,auto,nonegotiate
Channel        6/1-2,6/1-4
```

8.5 Creating VLANs with the Nortel-Juniper collaboration.

To make a VLAN with an external router, we create a series of VLAN broadcast domains on the Nortel switch, which overlap at a single port (here, the optical port 23). We enable 801.1q tagging on that port and set a VLAN id. All the traffic is then labelled or tagged by the switch and is sent via port 23 to the router. The router removes the tags and sends the relevant traffic to the relevant logical interface.

To create vlans on the Juniper M7 router, we create virtual interfaces with Juniper's logical interface "units", and then tag them with an 801.1Q vlan id. First we prepare the interface for 801.1q tagging:

```
root@amoeba# cli
root@amoeba# configure
root@amoeba# set interfaces ge-1/3/0 vlan-tagging
```

Then we define a virtual interface for each VLAN, using the Juniper virtual "units".

```
# set interfaces ge-1/3/0 unit 0 vlan-id 0
# set interfaces ge-1/3/0 unit 1 vlan-id 100
# set interfaces ge-1/3/0 unit 1 family inet address 128.39.65.3/24
# set interfaces ge-1/3/0 unit 2 vlan-id 200
# set interfaces ge-1/3/0 unit 2 family inet address 10.10.10.10/24
# commit
# quit
```

The unit 0 interface seems to be reserved, though we have not found this in any documentation. Once you enable VLAN-tagging, all of the sub-interfaces must have a VLAN-id, even unit 0.

```
# show interfaces

interfaces {
    ge-1/3/0 {
        vlan-tagging;
        unit 0 {
            vlan-id 0;
        }
        unit 1 {
            vlan-id 100;
            family inet {
                address 128.39.65.3/24;
            }
        }
        unit 2 {
            vlan-id 200;
            family inet {
                address 10.10.10.10/24;
            }
        }
    }
}
```

The router should now be ready.

On the Nortel Baystack, login on the console (wait for the rolling banner to stop!) and use the menus to configure the ports. In the menu system the space-bar changes a value and ENTER selects it.

On the main menu, choose “Switch Configuration”, then “VLAN configuration” then “VLAN Port Configuration”. Under this section you will set two things:

- AutoPVID (all ports) should be enabled.
- Tagging should be switched off on all ports except the trunking port (23).

VLAN Port Configuration

```
Port: [ 1 ]
Filter Untagged Frames: [ No ]
Filter Unregistered Frames: [ No ]
Port Name: [ Port 1 ]
PVID: [ 100 ]
Port Priority: [ 0 ]
Tagging: [ Untag All ]

AutoPVID (all ports): [ Enabled ]
```

VLAN Port Configuration

```
Port: [ 23 ]
Filter Untagged Frames: [ No ]
Filter Unregistered Frames: [ No ]
Port Name: [ Port 23 ]
PVID: [ 1 ]
Port Priority: [ 0 ]
Tagging: [ Tag All ]

AutoPVID (all ports): [ Enabled ]
```

Turning on auto PVID on all ports maps the VLAN id-number to the port series number automatically. Now go back to the menu above and choose “VLAN configuration”.

- Create VLAN 100, and enable it.
- Create VLAN 200. and enable it.

Use the port membership interface to set the ports you want to belong to the current VLAN:

VLAN Configuration

```
Create VLAN: [ 100 ]          VLAN Type: [ Port-Based ]
Delete VLAN: [ ]             Protocol Id (PID): [ None ]
VLAN Name: [ VLAN #100 ]     User-Defined PID: [ 0x0000 ]
Management VLAN: [ No ] Now: 1  VLAN State: [ Active ]
```

Port Membership

```
1-6   7-12 13-18 19-24
-----
```

```
Unit #1  ++++++ ----- +----- +-----
```

KEY: + = A Member of This VLAN, - = Not a Member of This VLAN

Use space bar to display choices, press <Return> or <Enter> to select choice.

Press Ctrl-R to return to previous menu. Press Ctrl-C to return to Main Menu.

8.5. CREATING VLANS WITH THE NORTEL-JUNIPER COLLABORATION.

Note that you have to make sure the VLANs do not overlap on the ports yourself. Except on the trunk port (23), where they must overlap.

Note that the Nortel switch is somewhat sensitive to the order in which you do things. If you find that a VLAN is not working, try undefining or deleting the VLAN and re-defining it. Remember to turn on AutoPVID first. If you get stuck, you can always choose to reset the switch to factory defaults and start again.

Place one PC on VLAN 100 and another on VLAN 200, and make sure you can ping your way around the network. Remember to set the default routes on the PCs to point to the router addresses. Since all interfaces are local, you should not need to run a routing protocol.

```
pc-100# ifconfig eth0 128.39.65.77 netmask 255.255.255.0 broadcast 128.39.65.255
pc-100# route add default gw 128.39.65.3
```

```
pc-200# ifconfig eth0 10.10.10.2 netmask 255.255.255.0 broadcast 10.10.10.255
pc-200# route add default gw 10.10.10.10
```

```
pc100# ping 128.38.65.3
pc100# ping 10.10.10.10
pc100# ping 10.10.10.2
```


Part III

Suggested In-depth Experiments

Chapter 9

Advanced routing

Experiment 3. *The purpose of this experiment is to develop your experience with the simplest distance vector routing protocols.*

- **Estimated time:** *Three weeks*
- **Preparation:** *Read about*
- **Equipment:** *PCs routers and switches.*
- **How to leave the lab after session:** *No configurations must be stored.*

This experiment starts with the scenario described in the preliminary course MS003A between routers in Oslo and Paris. You should not configure OSPF in this exercise however.

9.1 RIP V1/V2

The first part of the experiment is about the Routing Internet Protocol. In the course MS003A, only the basic version 1 protocol was discussed, but several improvements have been made to RIP that make it a workable solution for small network clusters.

You can use the text book *Cisco Cookbook*, chapter 6 to help you with this exercise.

1. Configure RIP on Paris, Oslo and Backbone and verify that RIP is up and running

```
128.39.0.0/16 is variably subnetted, 2 subnets, 2 masks
R      128.39.0.0/16 [120/1] via 10.10.43.3, 00:00:03, Ethernet0
      10.0.0.0/24 is subnetted, 2 subnets

backbone#sh ip route rip
      128.39.0.0/16 is variably subnetted, 2 subnets, 2 masks
R      128.39.0.0/16 [120/1] via 10.10.43.3, 00:00:18, FastEthernet0/0
```

This shows which route has been learned by the router.

RIP is a distance vector protocol which broadcasts its full routing table to all routers in the network. The advertised routes are received by all the other routers. These routers regularly update their forwarding tables.

```
23:19:28: RIP: sending v1 update to 255.255.255.255 via Ethernet0 (10.10.43.2)
23:19:28:      network 222.45.3.0, metric 1
23:19:28: RIP: sending v1 update to 255.255.255.255 via FastEthernet0 (222.45.3)
23:19:28:      default, metric 1
```

```
23:19:28:      network 128.39.0.0, metric 2
23:19:28:      network 10.0.0.0, metric 1
23:19:34: RIP:  received v1 update from 10.10.43.3 on Ethernet0
23:19:34:      128.39.0.0 in 1 hops
23:19:56: RIP:  sending v1 update to 255.255.255.255 via Ethernet0 (10.10.43.2)
23:19:56:      network 222.45.3.0, metric 1
23:19:56: RIP:  sending v1 update to 255.255.255.255 via FastEthernet0 (222.45.3)
23:19:56:      default, metric 1
23:19:56:      network 128.39.0.0, metric 2
23:19:56:      network 10.0.0.0, metric 1
23:20:04: RIP:  received v1 update from 10.10.43.3 on Ethernet0
23:20:04:      128.39.0.0 in 1 hops
23:20:24: RIP:  sending v1 update to 255.255.255.255 via Ethernet0 (10.10.43.2)
23:20:24:      network 222.45.3.0, metric 1
23:20:24: RIP:  sending v1 update to 255.255.255.255 via FastEthernet0 (222.45.3)
23:20:24:      default, metric 1
23:20:24:      network 128.39.0.0, metric 2
23:20:24:      network 10.0.0.0, metric 1
23:24:57: RIP:  received v1 update from 10.10.43.4 on Ethernet0
23:24:57:      0.0.0.0 in 1 hops
23:24:57:      198.18.1.0 in 1 hops
23:24:57:      199.19.1.0 in 1 hops
```

2. There's a number of techniques that are used to reduce the complexity and the issue of large periodic broadcasting. By reading in the course texts, find out what these are and use them to:
 - Speed up the convergence time.
 - Reduce the bandwidth utilization on the network.
 - Manage the network in general: i.e. specify routes, filter routes and summarize routes.
3. Add a loopback interface on the network with the same network as the Oslo network but a different subnet Show the routing summary output from RIP and explain the result you get. What is happening?
4. Try to prevent the updating on an interface by using a `passive-interface` command.

9.2 EIGRP - an improved RIP?

You can use the text book *Cisco Cookbook*, chapter 7 to help you with this exercise.

1. Configure basic EIGRP on the network now and compare it with RIP Can you use the same techniques as with RIP? What is the advantage of using EIGRP instead of RIP? Which one would you recommend to your customer (explain your answer)?
2. Load balancing is the capability of a router to distribute traffic over all its network ports that are the same distance from the destination address. Load balancing increases the utilization of network segments. This means that it increases the network capacity. There are two types of load balancing: *equal cost path* and *unequal cost path*.

Do some reading so that you will understand how unequal cost path load balancing works in the Enhanced Interior Gateway Routing Protocol (EIGRP). Test out the flow of traffic on balanced routes, explaining what tools you use to make measurements.

9.3 Routing loops

Recall the problem of routing loops and counting to infinity. Several methods and techniques exist for preventing these problems:

- Split horizon: never advertise a route out of the interface through which you learned it.
- Poison reverse: once you learn of a route through an interface, advertise it as unreachable back through that same interface.

If you have time, learn about these methods and test them out.

Chapter 10

BGP Policies and Virtual Tunnels

Experiment 4. *The purpose of this experiment is to become more familiar with the BGP routing system and with concepts such as peering.*

- **Estimated time:** *Three weeks*
- **Preparation:** *Read about BGP and Virtual Tunnels.*
- **Equipment:** *PCs routers and switches.*
- **How to leave the lab after session:** *No configurations must be stored.*

In this experiment you must set up a network consisting of three autonomous systems. The underlying routing can be configured by any suitable method.

10.1 Autonomous systems

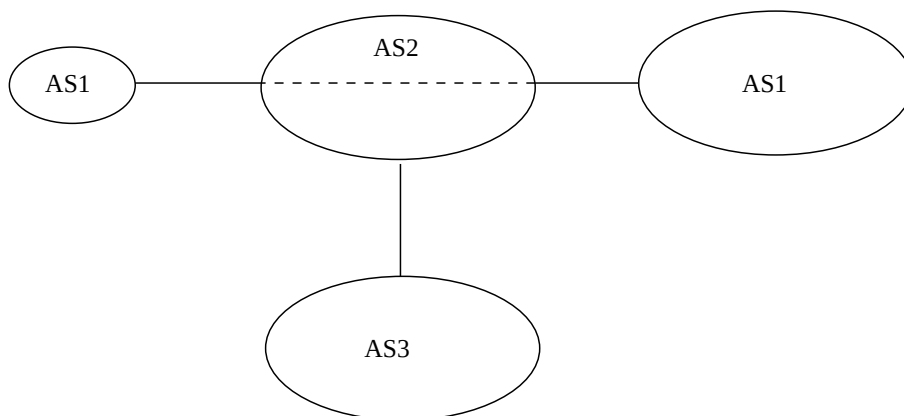


Figure 10.1: A fragment of AS1 tunnels through AS2 in order to be contiguous with its larger parent network region.

Use 2 routers in AS2 and 1 router in all the other domain fragments above. The two routers in AS2 should communicate with iBGP and there should be eBGP communication between the ASs.

The question in this exercise is: what happens to the smaller fragment of AS1 on the left hand side of the figure? Suppose that this is a branch office of a larger company that is geographically separate from the main centre. Formally, one would like the entire company to be identified with AS1, but the BGP believes that the routing domain is a convex region with a definite boundary.

The way one normally handles this is to create a point to point link that makes the two sub-regions into a convex one. This can be done by one of two methods:

- A physical point-to-point connection from the sub-region to the main region.
- A virtual “tunnel” from the sub-region to the main region.

Virtual tunnels are a part of the IPSec functionality, for instance.

Create a secure tunnel from the fragment of AS1 to the main region and tie them together with the underlying routing protocol.

10.2 Peering

Add a direct connection between AS1 and AS3 and create a peer group between AS1 and AS2 so that they share policy. Describe the filtering policies that you implement.

10.3 Configuration errors

One of the problems with policy based management is that there is no one policing the consistency of the rules at different parts of the network. What happens if two ASs advertise the same routes as being local to themselves?

Use the routers to create a “black hole” in which one of the ASs advertises a network that it does not have. Show how this causes a problem for the inter-AS routing.

10.4 Equilibration

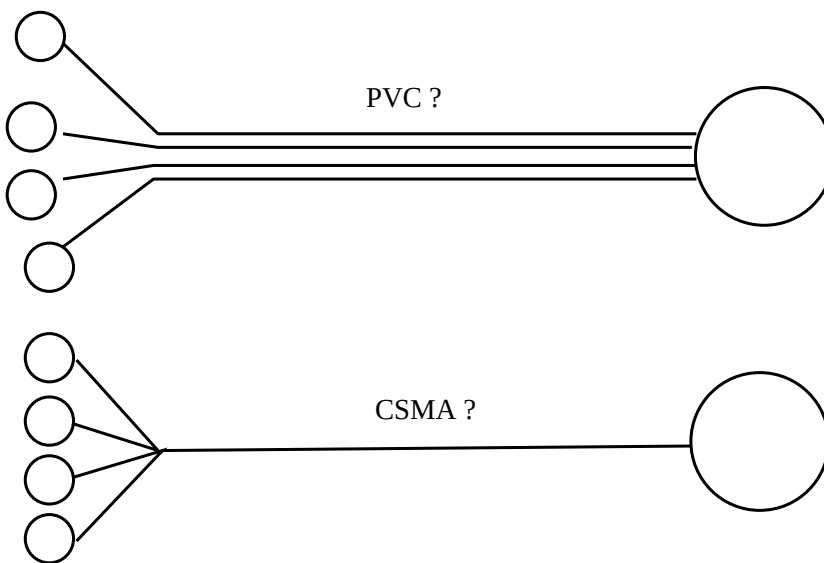
Measure how long it takes for a consistent configuration to emerge in the BGP network above when a change is made.

Chapter 11

Traffic Shaping and Quality of Service

Experiment 5. *The purpose of this experiment is to perform a scientific study of network performance under varying load. You will need to study methods of measurement and inquiry and plan how to proceed. You must present your report detailing the method, results and conclusions of your study.*

- **Estimated time:** *Four to five weeks*
- **Preparation:** *Read about measurement and scientific method. Recall exercises on frame relay. Read about traffic shaping and Quality of Service.*
- **Equipment:** *PCs, routers and switches.*
- **How to leave the lab after session:** *No configurations must be stored.*



You can use the course book *Cisco Cookbook*, chapter 10.1 and 10.4 etc. to help you here.

11.1 Scenario 1

Imagine a small group of organizations that share a connection to the Internet by a leased line. In this experiment you will measure the performance (under varying loads) of a router with multiple hosts. Set up a scenario for sharing the bandwidth of a trunk line that will be shared equally between several end hosts. Various functionality exists in IOS:

- Bandwidth management through rate limiting - Allows you to control the maximum rate of traffic sent or received on an interface.
- Traffic policing is often configured on interfaces at the edge of a network to limit traffic into or out of the network.
- Traffic that falls within the rate parameters is sent, whereas traffic that exceeds the parameters is dropped, or sent with a different priority.
- Packet marking through IP precedence, QoS group and so on.

The idea of this experiment is to pick a protocol with Quality of Service (QoS)/traffic shaping support and compare this to a competitively shared medium like Ethernet. Determine how these different strategies compare in performance when traffic is low/high and when the number of users sharing the link is low/high?

Referring to the figure above, you might choose to create 4 virtual sub-interfaces on the left hand side, with four different virtual circuits between end nodes and the right hand side. You would then compare this to four Ethernet nodes connected directly by a hub or LAN switch on the left.

When planning this experiment, think of the following points:

1. You should think about the problem above and interpret it in terms of the equipment you have available.
2. What protocols and data rates will you use? (10MB Ethernet, 100MB Ethernet? 56kB Frame Relay?) You will want to address questions like:
 - How much of the prioritized bandwidth is idle under different loads?
 - At which points in the network will you monitor the flows of traffic?
 - What will you use to monitor the flows? (ttcpd? show interface etc.)
 - What will you use to generate the flows? (ttcpd, private program etc)
3. You are asked, below, to find the change in behaviour of the system when you vary just one of many parameters that could be changed. Explain how you will isolate these independent channels of cause-effect.

You can not simulate the huge amounts of data that would be used in a real organization, so you will have to scale your experiment down to simulate a realistic case in the lab. Be sure to explain carefully how you do this.

In order to compare the performance of different sharing policies, you will have to do some thinking. We do not have an Ethernet and a Frame Relay connection with the same channel capacity, so you will have to measure the efficiencies and compare them by scaling or dividing by the maximum for that medium.

11.2 Measurements

Once you have designed your experiment and the system is working:

1. Define a measure of performance for each protocol: how much of the potential channel capacity is being used, e.g.
If the total channel capacity is C and we split it into 3 shares, then it should be possible to insert $C/3$ from each end host without problems. How much of this traffic actually arrives however? What happens if you try to send more than $C/3$?
2. How does the relative performance of the transmission vary between Ethernet and Frame Relay with the amount of traffic? i.e. as the maximum channel capacity is approached, how quickly are data sent? To be thorough, you can try both Frame Relay with and without traffic shaping limits.
3. How does the performance of total transmission vary with the number of users?
4. How does the performance of each customer's transmission vary with the number of users, assuming an equal share to all?

What data will you have to record to find answers to the questions above? How many times must you repeat the experiments to obtain a sensible answer? Use elementary statistics to show how reliable your estimates are.

11.3 Scenario 2

Frame-relay traffic shaping can also be applied to multi-service links. Find out, for instance, how to do Voice over IP by connecting an analogue telephone to the FX ports (on those routers that have them).

1. Start with configuring a frame-relay network with point to point sub-interfaces
 2. After checking that the frame-relay is active use QoS to enable traffic shaping on the network
- Here's a sample that show a traffic shaping configuration

```

policy-map mypolicy
  class voice
    priority 16
  class priority-data
    bandwidth 16
  class other-data
    priority 5

!--- Create a policy-map and apply the bandwidth
!--- command to a class.

!
int s0/1
  encapsulation frame-relay IETF
  load-interval 30
  frame-relay traffic-shaping

!--- Enable Frame Relay TS.

!
interface Serial0/0.1 point-to-point
  frame-relay interface-dlci 160
  class frclass

!--- Apply the map-class to the Frame Relay PVC.
```

```
!  
map-class frame-relay frclass  
  service-policy output mypolicy  
  frame-relay cir 64000  
  frame-relay bc 640  
  
!--- Apply the service-policy inside the map-class
```

Chapter 12

Wireless Networking

Experiment 6. *The purpose of this experiment is to perform a scientific study of wireless network performance.*

- **Estimated time:** *Two weeks*
- **Preparation:** *Read about wireless transmission and security.*
- **Equipment:** *PCs, routers and switches. Wireless sender.*
- **How to leave the lab after session:** *No configurations must be stored.*

Imagine creating a wireless network for a small business or home. Set up a wireless transmitter/receiver, connected to a LAN switch and use a PC connected to the LAN to send and receive traffic to a mobile laptop PC with a network card.

In this exercise you will be asked to test out the physical properties of wireless communication. You will need to find a way to measure data rates and signal strength and estimate the uncertainty in your measurements.

- Measure how far away from the source you can be.
- What materials shield the signal from the sender?
- Integrate the wireless network with the VLANs.

You can investigate anything that you like, but you should at least investigate the following and collect data for analysis:

1. Test the wireless network by line of sight (with no walls etc in between).
 - (a) Plot the signal strength S against the inverse distance $1/d$ and its square $1/d^2$.
 - (b) Plot the maximum data rate against the inverse distance $1/d$ and its square $1/d^2$.

Comment on the graphs that you have obtained.

2. Test two or three substances (walls, metals, clothing etc) for their signal shielding properties. Find a substance that will block transmission. Does it matter whether there are holes in the material? What thickness of your chosen materials will stop the signal completely (so that it is indistinguishable from noise)?

For example, try to get hold of some ‘chicken wire’ or garden fencing and see whether it shields the signal from the wireless transmitter.

Chapter 13

VPN and IPSec

Experiment 7. *The purpose of this experiment is to explore possibilities inherent in IPSec and virtual tunnelling technologies.*

- **Estimated time:** *Four weeks*
- **Preparation:** *Read about security.*
- **Equipment:** *PCs, routers with sufficient memory and CPU.*
- **How to leave the lab after session:** *No configurations must be stored.*

- Software encryption:
 - Cisco IOS Release 12.1(3)XI or later.
- Hardware encryption. For the 3600 series routers, you will need
 - Data encryption Advanced Integration Module (AIM)
 - Network Module (NM) provide hardware-based encryption.

The data encryption AIMs and NM are hardware Layer3 (IPSec) encryption modules and provide DES and Triple DES IPSec encryption for multiple E1s of bandwidth. They also have hardware support for Diffie-Hellman, RSA, and DSA key generation.

IPSEC commands on a 3660 router:

```
crypto isakmp policy 10
  hash md5
  authentication pre-share
crypto isakmp key cisco123 address X.X.X.X
!
crypto ipsec transform-set myset esp-des esp-md5-hmac
!
crypto map mymap local-address FastEthernet0/0
crypto map mymap 10 ipsec-isakmp
  set peer X.X.X.X
  set transform-set myset
  match address 101
!
interface Tunnel0
  ip address X.X.X.X 255.X.X.X
  no ip directed-broadcast
  ipx network BB
```

```
tunnel source FastEthernet0/1
tunnel destination X.X.X.X
crypto map mymap
```

Chapter 14

Virtual machines and large installation configuration

Experiment 8. *The purpose of this experiment is to investigate the problem of managing a large number of computers that are networked together. Since we do not have a large number of computers, you will simulate them using virtual machines.*

- **Estimated time:** *Three weeks*
- **Preparation:** *Read about User-Mode Linux and test it out. Read about getting started with cfengine.*
- **Equipment:** *hosts netlab1-netlab4.*
- **How to leave the lab after session:** *simulations can run if there are sufficient resources.*

14.1 Virtual networks

You should spend a week learning how to configure User Mode Linux and designing a network structure, with two subnets.

14.2 Managing the network

Once you are familiar with how to create a virtual network simulation:

1. Using User-Mode Linux, design and set up a network consisting of two different subnets and about fifty machines. Give your subnets different subnet masks.
2. Set up E-mail services, several web servers and use cfengine to manage the hosts.
3. Simulate the activity of users by writing scripts that create files and use up CPU and memory, and using programs like **wget** to simulate web downloads.
4. You should set up cfengine to report to you as system administrator by E-mail. Investigate how well the cfengine management scales with numbers of machines, especially when files are being distributed from a central origin.
5. Can you make your network run on IPv6?

Chapter 15

Sensor monitoring

Experiment 9. *The purpose of this experiment is to build a custom temperature-monitoring system and gather temperature information from the lab using Smartec temperature sensors. This project includes two parts:*

- **Estimated time:** *Two weeks*
- **Preparation:** *Construction, soldering, SNMP.*
- **Equipment:** *hosts netlab1-netlab4.*
- **How to leave the lab after session:** *simulations can run if there are sufficient resources.*

This project requires some basic knowledge about electrical circuits, and it is an advantage with some soldering experience.

1. Assemble the temperature kit by soldering a semiconductor temperature sensor and the temperature acquisition board together. You can attach up to 4 sensor to one temperature acquisition board.
2. Collect the temperature data from the temperature acquisition board, as a time-series, and find a way to store long-term temperature data. Visualize the temperature development using e.g. RRDTool, xmgrace or gnuplot (RRDTools is recommended due to its round robin design!).

What is considered acceptable temperatures for the different equipment? How is the temperature in the racks affected by optimizing the air flow? What are the differences in temperature between the intake and the outtake?

If you have more time you can also gather all the temperature information on a single web page so that it can be included in the NOC*.

http://www.smartec.nl/temperature_kit.htm

http://www.smartec.nl/temperature_sensor.htm

Chapter 16

XEN Virtual machines

Experiment 10. *The purpose of this experiment is to explore the XEN virtualization technology and measure the performance overhead and scalability of XEN virtual machines.*

- **Estimated time:** *Three weeks*
- **Preparation:** *Read about XEN and set up some virtual machines.*
- **Equipment:** *Available PCs.*
- **How to leave the lab after session:**

The authors of Xen claim that running a machine as a Xen domain instantiation only adds a worst-case overhead of 8% and an average overhead of 2%.

Experiment with creating, destroying, and migrating Xen domains from one PC to another. See how the performance worsens as the number of concurrent machines is increased.

<http://www.cl.cam.ac.uk/Research/SRG/netos/xen/>

<http://www.cl.cam.ac.uk/Research/SRG/netos/xen/documentation.html>

<http://www.fedoraproject.org/wiki/FedoraXenQuickstart>

Chapter 17

XORP router

Experiment 11. *The purpose of this experiment is to build a XORP router and measure the performance by comparing it to Cisco and Juniper routers.*

- **Estimated time:** *Four weeks*
- **Preparation:** *Read about XORP and collect the necessary equipment.*
- **Equipment:** *Available PCs.*
- **How to leave the lab after session:**

1. Build a software router using XORP.
2. Perform a qualitative comparison of the functionality of the router with the commercial routing equipment in the lab.
3. Measure its performance in terms of routing table convergence and packet transmission rate, etc, for at least two different protocols.
4. How do old and new routers compare to the XORP router?
5. Are there any advantages to the XORP approach over a dedicated commercial router?

Chapter 18

Linux Terminal Server Project

Experiment 12. *The purpose of this experiment is to measure the response and scalability of Linux running over a virtual terminal server. You should formulate some questions of your own.*

- **Estimated time:** *Three weeks*
- **Preparation:** *Read about LTSP and set up some instances.*
- **Equipment:** *Available PCs.*
- **How to leave the lab after session:**

1. Build and configure a LTSP network using either LTSP or Skolelinux, with and at least four thin clients.
2. How many clients can connected at the same time?
3. How much load can a single server handle before the system becomes unusable (in terms of delay and "sluggishness").
4. What are the bottlenecks?
5. What is more important; the number of clients, or the power (RAM/CPU) of the server?
6. What about scalability? How does this solution compare to other comparable technologies such as RDP (Windows) and Nomachine?

Chapter 19

High availability web services

Experiment 13. *The purpose of this experiment is to investigate strategies for redundant, high-volume network services.*

- **Estimated time:** *Three weeks*
- **Preparation:** *Read about User-Mode Linux and test it out. Read about getting started with cfengine.*
- **Equipment:** *hosts netlab1-netlab4.*
- **How to leave the lab after session:** *simulations can run if there are sufficient resources.*

1. Set up a GNU/Linux Apache web server on a relatively old (slow) PC and test its ability to serve files.
2. Investigate how the performance of the server varies with the number of pre-forked processes and the configuration parameter that governs the maximum number of servers. Measure the latency (average time to wait) and the average data rate, using tools of your own choice.
3. Create a load balancer that allows you to connect two PCs in parallel. How does the performance of this parallel configuration compare to the performance of a single machine as the same parameters are varied as above.
4. Install a Microsoft Windows 2000 Internet Information Server (IIS) and compare its performance with a single Apache server on Linux, on the same hardware. Compare the performance of the servers, the latency and the scalability of these two solutions. Can you say whether one or the other is “better”?
5. Suggest a design for a high volume server.

Chapter 20

Disk and RAID performance

Experiment 14. *The purpose of this experiment is to investigate the performance of ATA (IDE) and SCSI disks under different loads and to test the performance penalty for using RAID solutions. You should design an experiment to stress test these technologies under single and multi-user access.*

- **Estimated time:** *Two to three weeks*
- **Preparation:** *Read about ATA (IDE) and SCSI disks, their strengths and weaknesses.*
- **Equipment:** *Anything you can get your hands on!*
- **How to leave the lab after session:** *Equipment should be tidied but may be labelled or reserved for the group.*

Design your own experiment to measure the performance of a disk.

- Consider what it is that you will measure.
- What tools will you use?
- What hypotheses will you form and test?
- How will you organize and analyze your data?
- How will you arrange to test the disks to their limits?
- How does performance change as load increases? Using current prices, work out the performance of each disk per price/MB.
- Using any tools that you can find, format a new disk (filesystem) and write some recognizable data to it. Then delete all of the files. Using tools of your choice, see how much of the data you are able to recover from the disk.

Chapter 21

SNMP and Network Operations Centre

The Simple Network Management Protocol is an important technology for network devices. This experiment asks you to learn about its use.

Experiment 15. *The purpose of this experiment is to investigate some of the characteristics of the Simple Network Management Protocol.*

- **Estimated time:** *One to two weeks*
- **Preparation:** *Read about SNMP and monitoring on the web.*
- **Equipment:** *Any.*
- **How to leave the lab after session:** *software etc. must be removed once the experiment is over.*

When using the SNMP tools below, use a program like `tcpdump` to show the traffic that passes between your console and the devices. In your final report, you should comment on the overhead incurred by SNMP.

1. Find out whether you need to enable SNMP services on routers and switches.
2. Write a PHP or Perl script that queries SNMP data from a printer.
3. Locate, download and compile the `scli` tool.
4. Locate MRTG noting that this might be already installed on the machine you are using. Install it and get it running. Show how it generates plots of traffic activity.
5. Investigate using the `scli` tool to configure routers and switches.

http://www.emeapress.att.com/library/images/att_noc_01.jpg

<http://people.ee.ethz.ch/~oetiker/webtools/rrdtool/>

<http://php.net>

Chapter 22

Multiseat system Under Linux

Multiseat systems provide multiple graphical interface hardware sets (monitor/keyboard/mouse/optionally sound) so that several people can use one computer at a time.

Experiment 16. *The purpose of this experiment is to investigate The power and advantages of using a Multiseat system.*

- **Estimated time:** *One to three weeks*
- **Preparation:** *Read about Multiseat systems on the web.*
- **Equipment:** *Pc's, video adapter, keyboard, and mouse for each seat.*
- **How to leave the lab after session:** *software etc. must be removed once the experiment is over.*

1. Find out about the type of hardware that is required to build Multiseat systems.
2. Install the required software to make Linux Multiseat capable.
3. What are the advantages of a Multiseat systems and where is it more likely to be used.
4. How is a Multiseat system compared to solutions such as Linux Terminal Server Project (LTSP).

<http://www.linuxtoys.org/multiseat/multiseat.html>

<http://www.onlamp.com/pub/wlg/8660>

<http://www.ltsp.org/>

Chapter 23

Miscellaneous

You are free to suggest your own experiments for the laboratory. You will gain course credit as long as the course instructors have approved the plan. An experiment should start by asking a question and should contain a degree of analysis.

- What is your hypothesis?
- How will you test it critically? (You should not set out to prove a hypothesis, but rather to disprove it!)

Suggested topics:

- Novell/Windows integration.
- Cengine on windows.
- Compare OSPF routing with IS-IS routing.
- Dialup services.
- Compare different operating systems.
- IP Telephony.
- Dummynet and traffic shaping with FreeBSD.