Chapter 1

Introduction

1.1 What is network and system administration?

Network and system administration is a branch of *engineering* that concerns the operational management of human–computer systems. It is unusual as an engineering discipline in that it addresses both the technology of computer systems and the users of the technology on an equal basis. It is about putting together a network of computers (workstations, PCs and supercomputers), getting them running and then *keeping* them running in spite of the activities of *users* who tend to cause the systems to fail.

A system administrator works for users, so that they can use the system to produce work. However, a system administrator should not just cater for one or two selfish needs, but also work for the benefit of a whole community. Today, that community is a global community of machines and organizations, which spans every niche of human society and culture, thanks to the Internet. It is often a difficult balancing act to determine the best policy, which accounts for the different needs of everyone with a stake in a system. Once a computer is attached to the Internet, we have to consider the consequences of being directly connected to all the other computers in the world.

In the future, improvements in technology might render system administration a somewhat easier task – one of pure resource administration – but, today, system administration is not just an administrative job, it is an extremely demanding engineer's job. It's about hardware, software, user support, diagnosis, repair and prevention. System administrators need to know a bit of everything: the skills are technical, administrative and socio-psychological.

The terms *network administration* and *system administration* exist separately and are used both variously and inconsistently by industry and by academics. System administration is the term used traditionally by mainframe and Unix engineers to describe the management of computers whether they are coupled by a network or not. To this community, network administration means the management of network infrastructure devices (routers and switches). The world of personal computers (PCs) has no tradition of managing individual computers and their subsystems, and thus does not speak of system administration, *per se*.

To this community, network administration is the management of PCs in a network. In this book, we shall take the first view, since this is more precise.

Network and system administration are increasingly challenging. The complexity of computer systems is increasing all the time. Even a single PC today, running Windows NT, and attached to a network, approaches the level of complexity that mainframe computers had ten years ago. We are now forced to think *systems* not just computers.

1.2 Applying technology in an environment

A key task of network and system administration is to build hardware configurations, another is to configure software systems. Both of these tasks are performed for users. Each of these tasks presents its own challenges, but neither can be viewed in isolation.

Hardware has to conform to the constraints of the physical world; it requires power, a temperate (usually indoor) climate, and a conformance to basic standards in order to work systematically. The type of hardware limits the kind of software that can run on it. Software requires hardware, a basic operating system infrastructure and a conformance to certain standards, but is not necessarily limited by physical concerns as long as it has hardware to run on.

Modern software, in the context of a global network, needs to inter-operate and survive the possible hostilities of incompatible or inhospitable competitors. Today the complexity of multiple software systems sharing a common Internet space reaches almost the level of the biological. In older days, it was normal to find proprietary solutions, whose strategy was to lock users into one company's products. Today that strategy is less dominant, and even untenable, thanks to networking. Today, there is not only a physical environment but a technological one, with a diversity that is constantly changing. Part of the challenge is to knit apparently disparate pieces of this community into a harmonious whole.

We apply technology in such an environment for a purpose (running a business or other practice), and that purpose guides our actions and decisions, but it is usually insufficient to provide all the answers. Software creates abstractions that change the basic world view of administrators. The software domain `.com` does not have any fixed geographical location, but neither do the domains `.uk` or `.no`. Machines belonging to these software domains can be located anywhere in the world. It is not uncommon to find foreign embassies with domain names inside their country of origin, despite being located around the world. We are thus forced to think globally.

The global view, presented to us by information technology means that we have to think penetratingly about the systems that are deployed. The extensive filaments of our inter-networked systems are exposed to attack, both accidental and malicious in a competitive jungle. Ignore the environment and one exposes oneself to unnecessary risk.

1.3 The human role in systems

For humans, the task of system administration is a balancing act. It requires patience, understanding, knowledge and experience. It is like working in the casualty ward of a hospital. Administrators need to be the doctor, the psychologist, and – when instruments fail – the mechanic. We need to work with the limited resources we have, be inventive in a crisis, and know a lot of general facts and figures about the way computers work. We need to recognize that the answers are not always written down for us to copy, that machines do not always behave the way we think they should. We need to remain calm and attentive, and learn a dozen new things a year.

Computing systems require the very best of organizational skills and the most professional of attitudes. To start down the road of system administration, we need to know many *facts* and build confidence though experience – but we also need to know our limitations in order to avoid the careless mistakes which are all too easily provoked.

1.4 Ethical issues

Because computer systems are human–computer communities, there are ethical considerations involved in their administration. Even if certain decisions can be made objectively, e.g. for maximizing productivity or minimizing cost, one must have a policy for the use and management of computers and their users. Some decisions have to be made to protect the rights of individuals. A system administrator has many responsibilities and constraints to consider. Ethically, the first responsibility must be to the greater network community, and then to the users of our system. An administrator's job is to make users' lives bearable and to empower them in the production of real work.

1.5 Is system administration a discipline?

Is system administration a science? Is computer science a science? The same question has been asked of many disciplines. We can answer the question in like mind here. Unlike physics, chemistry or biology, system administration is lacking in a systematic body of experimental data which would give its rules and principles an empirical rigor. However, that is not to say that system administration cannot be made to follow this scientific form. Indeed, there is good reason to suppose that the task is easier in the administration of systems than in fields like software engineering, where one cannot easily separate human subjective concerns from an objective empiricism.

System administration practices, world-wide, vary from the haphazard to the state of the art. There is a variety of reasons for this. The global computer community has grown considerably, operating systems have become increasingly complex, but the number of system administrators has not grown in proportion. In the past, system administration has been a job which has not been carried out by dedicated professionals, but rather by interested computer users, as a necessary chore in getting their work done. The focus on making computers easy to use has distracted many vendors from the belief that their computers should also be easy to manage. It is only over the gradual course of time that this has changed, though even today, system administrators are a barely visible race, until something goes wrong.

The need for a formal discipline in system administration has been recognized for some time, though it has sometimes been met with trepidation by many corners of the Internet community, perhaps because the spirit of free cooperation which is enjoyed by system administrators could easily be shattered by too pompous an academic framework. Nonetheless, there are academics and software engineers working on system administration, and it is quite common for system administrators to spawn from a scientific education.

Academic concerns aside, the majority of computer systems lie in the private sector, and the Internet is only amplifying this tendency. In order to be good at system administration, a certain amount of dedication is required, with both theoretical and practical skills. For a serious professional, system administration is a career in engineering. There is now an appreciable market for consulting services in security and automation of system administrative tasks. Not only is system administration a fascinating and varied line of work, it can also be lucrative.

1.6 The challenges of system administration

System administration is not just about installing operating systems. It is about planning and designing an efficient *community* of computers so that real *users* will be able to get their jobs done. That means:

- Designing a network which is logical and efficient.
- Deploying large numbers of machines which can be easily upgraded later.
- Deciding what services are needed.
- Planning and implementing adequate security.
- Providing a comfortable environment for users.
- Developing ways of fixing errors and problems which occur.
- Keeping track of and understanding how to use the enormous amount of knowledge which increases every year.

Some system administrators are responsible for both the hardware of the network and the computers which it connects, i.e. the cables as well as the computers. Some are only responsible for the computers. Either way, an understanding of how data flow from machine to machine is essential as well as an understanding of how each machine affects every other.

In all countries outside the United States, there are issues of internationalization, or tailoring the input/output hardware and software to local language. Internationalization support in computing involves three issues:

- Choice of keyboard: e.g. British, German, Norwegian, Thai etc.
- Fonts: Roman, Cyrillic, Greek, Persian etc.
- Translation of program text messages.

Inexperienced computer users usually want to be able to use computers in their own language. Experienced computer users, particularly programmers, often prefer the American versions of keyboards and software in order to avoid the awkward placement of commonly used characters on non-US keyboards.

1.7 Common practice and good practice

In a rational world, every choice needs a reason, even if that reason is an arbitrary choice. That does not undermine the need for a book of this kind, but it cautions us about accepting advice on trust. This is just the scientific method at work: informed scepticism and constant reappraisal.

If this book does nothing else, it should encourage a critical approach to network and system engineering. One can spend a career hearing advice from many different sources and not all of it will be good advice. The best generic advice anyone can give in life is: think for yourself; pay attention to experts but don't automatically believe anyone.

In the system administration world, it is common to speak of 'best practice'. A scientific mind is immediately suspicious of such a claim. In what sense is a practice best? When and for whom? How should one evaluate such a claim. This is one of the things we wish to consider in this book.

Clearly, it is always a good idea to see what others have done in the past, but history has no automatic authority. There are three reasons why ideas catch on and 'everyone does it':

- Someone did it once, the idea was copied without thinking and no one has thought about it since. Now everyone does it because everyone else does it.

- Experts have thought a lot about it and it really is the best solution.

- An arbitrary choice had to be made and now it is a matter of convention.

For example, in the British Isles it is a good idea to drive on the left-hand side of the road. That's because someone started doing so and now everyone does it – but it's not just a fad: lives actually depend on this. The choice has its roots in history and in the dominance of right-handed sword-wielding carriage drivers and highwaymen but, for whatever reason, the opposite convention now dominates in other parts of the world and, in Britain, the convention is now mainly preserved by the difficulty of changing. This is not ideal, but it is reasonable.

Some common practices, however, are bizarre but adequate. For instance, in parts of Europe the emergency services Fire, Police and Ambulance have three different numbers (110, 112 and 113) instead of one simple number like 911 (America) or, even simpler, 999 (UK). The numbers are very difficult to remember; they are not even a sequence. Change would be preferable.

Other practices are simply a result of blind obedience to poorly formulated rules. In public buildings there is a rule that doors should always open outwards from a room. The idea is that in the case of fire, when people panic, doors should 'go with the flow'. This makes eminent sense where large numbers of people are involved. Unfortunately the building designers of my College have taken this literally and done the same thing with every door, even office doors in narrow corridors. When there is a fire (actually all the time), we open our doors into the faces of passers-by (the fleeing masses), injuring them and breaking their noses. The rule could perhaps be reviewed.

In operating systems, many conventions have arisen, e.g. the conventions for naming the 'correct' directory for installing system executables, like daemons, the permissions required for particular files and programs and even the use of particular software; e.g. originally Unix programs were thrown casually in `usr/bin` or `etc`; nowadays `sbin` or `libexec` are used by different schools of thought, all of which can be discussed.

As a system administrator one has the power to make radical decisions about systems. Readers are encouraged to make logical choices rather than obedient ones.

1.8 Bugs and emergent phenomena

Operating systems and programs are full of bugs and emergent features that were not planned or designed for. Learning to tolerate bugs is a matter of survival for system administrators; one has to be creative and work around these bugs. They may come from:

- Poor quality control in software or procedures.
- Problems in operating systems and their subsystems.
- Unfortunate clashes between incompatible software, i.e. one software package interferes with the operation of another.
- Inexplicable phenomena, cosmic rays, viruses and other attacks.

A system administrator must be prepared to live with and work around these uncertainties, no matter what the reason for their existence. Not all problems can be fixed at source, much as one would prefer this to be the case.

1.9 The meta principles of system administration

Many of the principles in this book derive from a single overriding issue: they address the *predictability* of a system. The term system clearly implies an operation that is *systematic*, or predictable – but, unlike simple mechanical systems, like say a clock, computers interact with humans in a complex cycle of feedback, where uncertainty can enter at many levels. That makes human–computer systems difficult to predict, unless we somehow fix the boundaries of what is allowed, as a matter of policy.

Principle 1 (Policy is the foundation). *System administration begins with a policy – a decision about what we want and what should be, in relation to what we can afford.*

Policy speaks of what we wish to accomplish with the system, and what we are willing to tolerate of behavior within it. It must refer to both the component parts and to the environment with which the system interacts. If we cannot secure predictability, then we cannot expect long-term conformance with a policy.

> **Principle 2 (Predictability).** *The highest level aim in system administration is to work towards a predictable system. Predictability has limits. It is the basis of reliability, hence trust and therefore security.*

Policy and predictability are intertwined. What makes system administration difficult is that it involves a kind of 'search' problem. It is the hunt for a stable region in the landscape of all policies, i.e. those policies that can lead to stable and predictable behavior. In choosing policy, one might easily promote a regime of cascading failure, of increasing unpredictability, that degenerates into chaos. Avoiding these regimes is what makes system administration difficult.

As networks of computers and people grow, their interactions become increasingly complex and they become *non-deterministic*, i.e. not predictable in terms of any manageable number of variables. We therefore face another challenge that is posed by inevitable growth:

> **Principle 3 (Scalability).** *Scalable systems are those that grow in accordance with policy; i.e. they continue to function predictably, even as they increase in size.*

These meta-themes will recur throughout this book. The important point to understand about predictability is that it has limits. Human–computer systems are too complex and have too many interactions and dependencies to be deterministic. When we speak of predictability, it must always be within a margin of error. If this were not the case, system administration would not be difficult.

1.10 Knowledge is a jigsaw puzzle

Factual knowledge, in the world of the system administrator, is almost a disposable commodity – we use it and we throw it away, as it goes out of date. Then we need to find newer, more up-to-date knowledge to replace it. This is a continual process; the turn-around time is short, the loop endless, the mental agility required demanding. Such a process could easily splay into chaos or lapse into apathy. A robust discipline is required to maintain an island of logic, order and stability in a sea of turbulent change.

This book is about the aims and principles involved in maintaining that process – i.e. it is about the core of knowledge and ideas that remain constant throughout the turnover. It is supplemented with some practical, example recipes and advice. When you master this book you will come to understand why no single book will ever cover every aspect of the problem – you need a dozen others as well.[1] True knowledge begins with understanding, and understanding is a jigsaw puzzle

[1] Later you might want to look at some of the better how-to books such as those in the recommended reading list, refs. [223, 123, 122, 211].

you will be solving for the rest of your life. The first pieces are always the hardest to lay correctly.

1.11 To the student

To study this subject, we need to cultivate a way of thinking which embodies a basic scientific humility and some core principles:

- Independence, or self-sufficiency in learning. We cannot always ask someone for the right answer to every question.
- Systematic and tidy work practices.
- An altruistic view of the system. Users come first: collectively and only then individually.[2]
- Balancing a fatalistic view (the inevitability of errors) with a determination to gain firmer control of the system.

Some counter-productive practices could be avoided:

- The belief that there exists a right answer to every problem.
- Getting fraught and upset when things do not work the way we expect.
- Expecting that every problem has a beginning, a middle and an end (some problems are chronic and cannot be solved without impractical restructuring).

We can begin with a checklist:

- Look for answers in manuals and newsgroups.
- Use controlled trial and error to locate problems.
- Consider all the information; listen to people who tell you that there is a problem. It might be true, even if you can't see it yourself.
- Write down experiences in an A–Z so that you learn how to solve the same problem again in the future.
- Take responsibility for your actions. Be prepared for accidents. They are going to happen and they will be your fault. You will have to fix them.
- Remember tedious jobs like vacuum cleaning the hardware once a year.
- After learning about something new, always pose the question: *how does this apply to me?*

[2]The needs of the many outweigh the needs of the few (or the one)...

American English is the language of the net. System administrators need it to be able to read documentation, to be able to communicate with others and to ask questions on the Internet. Some sites have even written software tools for training novice administrators. See for instance, ref. [278]. Information can be found from many sources:

- Printed manuals
- Unix manual pages (`man` and `apropos` and `info` commands)
- The World Wide Web
- RFCs (Requests for comment), available on the web
- Newsgroups and discussions
- Papers from the SAGE/Usenix LISA conferences [22]
- More specialized books

A supplement to this book, with a collection of useful recipes and facts, is provided as a resource for system administrators at http://www.iu.hio.no/SystemAdmin. More detailed online course materials relating to the Oslo University Colleges Masters Degree are available at http://www.iu.hio.no/teaching/materials.

1.12 Some road-maps

This book contains many overlapping themes. If you are browsing through the book with a specific aim, the following road-maps might help you to shorten your journey.

1. Resource management: Chapters 2, 4, 5, 6, 7, 8, 9
2. Human management: Chapters 3, 5, 8, 11
3. IP networking: Chapters 2, 3, 6, 8, 10, 11, 12
4. System analysis: Chapters 3, 6, 8, 13
5. Security: Chapters 3, 5, 6, 7, 8, 11, 12

Much of the thinking behind the security policy recommendations in ISO 17799 permeate the book.

Exercises

Self-test objectives

1. What kinds of issues does system administration cover?
2. Is system administration management or engineering?

3. Why does the physical environment play a role in system administration?

4. Describe why ethics and human values are important.

5. Is system administration a science? Why/why not?

6. State the top-most principles that guide network and system administrators.

Problems

As a practical, hands-on subject, network and system administration exercises are heavily dependent on what equipment is available to students. Course instructors should therefore use the exercises in this book as templates for customizing their own exercises rather than viewing them as literal instructions.

1. Browse through this whole book from start to finish. Browsing information is a skill you will use a lot as a system administrator. Try to get an overall impression of what the book contains and how it is laid out.

2. List what you think are the important tasks and responsibilities of a system administrator. You will have the opportunity to compare this with your impressions once we reach the end of the book.

3. Locate other books and information sources which will help you. These might take the form of books (such as the recommended reading list at the end of this book) or newsgroups, or web sites.

4. Buy an A–Z notebook for systemizing the facts and knowledge that you pick up along the way.

5. What is an RFC? Locate a list of RFCs on a WWW or FTP server.