**HostManagement**

**Choices**

**We can make life easy or difficult for ourselves by the decisions we make at the outset of host installation. The first step in setting up hosts is to make some basic choices. Should we:**

**• Follow the OS designer's recommended setup? (Often not good enough.)**

**• Create our own setup?**

**• Make all machines alike?**

**• Make all machines different?**

**Booting and Shutting Down NT**

**The NT boot procedure on a PC begins with the BIOS, or PC hardware. This performs a memory check and looks for a bootable disk. A bootable disk is one which contains a Master Boot Record (MBR).**

**The boot block is located in the first sector of the bootable drive. It identifies which partition is to be used to continue with the boot procedure.**

**On each primary partition of a bootable disk, there is a boot program which 'knows' how to load the operating system it finds there. NT has a menu-driven boot manager program which makes it possible for several OSes to coexist on different partitions**

**Once the disk partition containing NT has been located, the program NTLDR is called to load the kernel. The file BOOT. INI configures the defaults for the boot manager**

In organizing disk space, we can make the best use of resources, and separate:

• Space for the operating system.

• Space which can be shared and made available for all hosts.

• Space which can be used to optimize local work, e.g. temporary scratch space, space which can be used to optimize local performance (avoid slow networking).

• Space can be used to make distributed backups, for multiple redundancy.

**Partitioning**

Disks can be divided up into partitions. Partitions physically divide the disk surface into separate areas which do not overlap.

The disk controller makes sure that partitions behave as independent, logical disks. The main difference between two partitions on one disk and two

separate disks is that partitions can only be accessed one at time, whereas multiple disks can be accessed in parallel.

**Formatting and Building File Systems**

Disk formatting is a way of organizing and finding a way around the surface of a disk

On a disk surface, it makes sense to divide up the available space into sectors or blocks.

Regrouping and labeling procedure is called *formatting in PC culture, and is called making a file system in Unix culture*

**Swap Space**

In Windows operating systems, virtual memory uses file system space for saving data to disk.

In Unix-like operating systems, a preferred method is to use a whole, unformatted partition for virtual memory storage

**File System Layout**

A working computer system has several facets:

• The operating system software distribution.

• Third party software.

• Users' files.

• Information databases.

• Temporary scratch space.

These are logically separate because

• They have different functions.

• They are maintained by different sources.

• They change at different rates.

• A different policy of backup is required for each.

**Principle (Separation I) *Data which are separate from the operating system should he*** *kept in a separate directory tree, preferably on a separate disk partition. If they are mixed with the operating system file-tree it makes re-installation or upgrade of the operating system unnecessarily difficult*

**Diskless Clients**

Diskless workstations are, as per the name, workstations which have no disk at all. Diskless workstations know absolutely nothing other than the MAC address of their network interface (Ethernet address).

**Dual Homed Host**

A host with two network interfaces, both of which is coupled to a network, is called a dualhomed host. Dual homed hosts are important in building *firewalls for network security*

**Cloning Systems**

We are almost never interested in installing every machine separately. A system administrator usually has to install ten, twenty or even a hundred machines at a time

• A few Unix-like operating systems provide a solution to this using package templates so that the installation procedure becomes standardized.

• The hard disks of one machine can be physically copied, and then the host name and IP address can be edited afterwards.

• All software can be placed on one host and shared using NFS, or another shared file system.

**User Management**

**User Registration**

One of the first issues on a new host is to issue accounts for users

The need for centralization is often in conflict with the need for delegation of

responsibility. It is convenient for autonomous departments to be able to register their own users, but it is also important for all users to be registered under the umbrella of the organization, to ensure unique identities for the users and flexibility of access to different parts of the organization

Registration

of single users under NT can be performed remotely from a workstation, using the

***net user username password /ADD /domain***

**Local and Network Accounts**

Most organizations need a system for centralizing passwords, so that each user will have the same password on each host on the network

With a local account, a user has permission to use only the local host.

With a network account, the user can use any host which belongs to a network *domain.*

In NT the Security Accounts Manager (SAM) is used to add local accounts to a given workstation.

For network accounts, Unix-like systems have widely adopted Sun Microsystems' **Network Information Service (NIS)**

An NT domain server involves not only shared databases, but also shared administrative policies and shared security models. A host can subscribe to one or more domains, and one domain can be associated with one another by a trust relationship

**Principle (Distributed accounts) *Users move around from host to host, share data and*** *collaborate. They need easy access to data and workstations all over an organization.*

**Suggestion 4 (Passwords) *Give users a common username on all hosts, of no more than*** *eight characters. Give them a common password on all hosts, unless there is a special reason not to do so. Some users never change their passwords unless forced to, and some users never even log in, so it is important to assign good passwords initially. Never assign a simple password and assume that it will be changed.*

**NT Accounts**

Single NT accounts are added with the command net user *username password /ADD /domain*

**Groups of Users**

Both Unix and NT allow users to belong to multiple groups. A group is an association of user names which can be referred to collectively by a single name. File and process permissions can be granted to a group of users.

Groups are created by command, rather than by file

editing, using

net group *groupname /ADD*

Users may then be added with the syntax

net group *groupname usernamel username2. . . /ADD*

Some standard groups are defined by the system, e.g.

Administrators

Users

Guest

The Administrators group has privileged access to the system.

**Account Policy**

Most organizations need a strict policy for assigning accounts and opening the system for users.

Closing Unix accounts can be achieved simply by changing their default shell in /etc/passwd

You account has been closed because your password was found to be vulnerable to attack. To reopen your account, visit the admin office , carrying some form of personal identification.

echo "$message"

sleep 10

exit 0

**Principle 19 (Environment) *It should always be clear to users which host they are using and*** *what operating system they are working with. Default environments should be kept simple both in appearance (prompts, etc.,) and in functionality (specially programmed keys, etc.). Simple environments are easy to understand.*

**Suggestion 5 (Clear prompts) *Try to give users a command prompt which includes the*** *name of the host they are working on. This is important, since different hosts might have different operating systems, or different files. Including the current directory in the prompt, like DOS, is not always a good idea.*

**The Superuser's Environment**

What kind of user environment should the superuser have? As we know, a privileged account has potentially dangerous consequences for the system. From this account, we have the power to destroy the system, or sabotage it. In short, the superuser's account should be configured to avoid as many casual mistakes as possible.

**Disk Space**

Disks fill up at an alarming rate. Users almost never throw away files unless they have to

**Suggestion 7 (Problem users) *Keep a separate partition for problem users' home directories,*** *so that they only cause trouble for one another, not for more considerate users.*

**Quotas and Limits**

• *Disk quotas: place fixed limits on the amount of disk space which can be used per user.*

• *CPU time limit: some faulty software packages leave processes running which consume* valuable CPU cycles to no use.

• *Policy decisions: users collect garbage. To limit the amount of it, one can specify a system* policy which includes items of the form: 'Users may not have mp3, wav, mpeg, etc., files on the system for more than one day

**Killing Old Processes**

Processes sometimes do not get terminated when they should.

One way to clean up processes in a work environment is to look for user processes which have run for more than a day.

Cfengine can also be used to clean up old processes. Cfengine's processes commands are used to match processes in the process table (which can be seen by running ps ax on Unix).

**Moving Users**

When disk partitions become full, it is necessary to move users from old partitions to new ones