1. A DAY IN THE LIFE OF A SYSTEM ADMINISTRATOR

Before you go delving into the intricacies of system maintenance and administration, we'd thought that you would appreciate sharing the experience of an experienced system administrator. So, we installed a hidden camera in the office of one our Beta-test sites, and filmed the entire day's activities. The version presented here was edited, of course, to the bare essentials, and all embarassing scenes removed. We fabricated names to protect the innocent, and to make it impossible for the subject to file an invasion of privacy suit.

Our experienced administrator's name is Clair. The company he works for has been using a Micronix system for over a year, and Clair has been responsible for maintaining the system every day of that period, except for his 2 week vacation at Lake Tahoe. We chose Clair for this study because he has never called and asked us to replace a lost data file that wasn't backed up, (a hopeless task). We are sorry that we can't give Clair the recognition that he truly deserves.

1.1. Turning on the System

Our saga begins with Clair walking into his office and sitting down at his desk. He swivels his chair to face the Decision cabinet, and takes a key out of his pocket. Clair inserts the key in the Decision, and turns it all the way to the right, so the Decision is now on and reset. The light on the floppy disk drive starts to flash, and Clair realizes that he needs to find the Micronix Load diskette.

Clair is a meticulous person, and manages to keep everything in its place. It's just that he's a little groggy this morning. He scratches his head, and looks on the table where the Decision sits for the Load diskette. And sure enough, there's a diskette with the word "LOAD" printed on the label in shaky letters. Clair slips the diskette into the drive, and closes the drive door just as the light flashes off. The next time the light turns on, the solenoid in the drive clicks and the Loading message appears on the console.

Clair turns back to his desk and takes another sip of coffee from a foam cup. In another half hour, he seems to be thinking, the guys in engineering will have a pot of strong French roast coffee going. (Clair drinks about three cups of coffee before his eyes are fully open most mornings.)

The hard disk has quit humming and whirring, so Clair swivels to face the console again, and logs in as "root". Clair smiles to himself while entering the password, as if the root password, known only to himself and one other person, has some special meaning for him. Then, he enters the correct time and date. According to his 10 function LCD diver's watch, it is exactly 8:17:31 on July 6.

```
# date July 6 8:17:55
Wed Jul 6, 1983 8:17:56
# []
```

So, Clair enters a date several seconds in the future, and waits to hit the return at the end of the **date** command until the time syncronizes perfectly with his watch. This isn't really necessary, but Clair likes to be picky about little things.

Next, Clair starts up the file system check program. He does this by simply typing

```
# fsck /dev/root
Checking /dev/root
  ...
```

and turning back to his desk. Clair knows that **fsck** takes about 5 minutes to complete, so he can get on with the business of the day. He looks first at this appointment calendar, sighs, gets up and walks out of the room.

1.2. Going Multi-user

The clock built into our hidden camera indicates that about 14 minutes have elasped since Clair wandered out of his office. He appears to be slightly flustered; must be a new secretary in the office. Clair glances at the console. Fsck has finished, and, as usual, there weren't any "victims". Good. Clair types the word "exit" after the prompt, sending Micronix into the business of becoming a multi-user system. As the hard disk begins to hum and whizz, he faces back to his desk, and reaches for the top of his "in" stack. Several minutes later, the Micronix banner and the log-in prompt (Name:) appear on Clair's terminal, and on the other terminals that have been turned on around the office. Micronix has "gone multi-user" and is ready for the day's work.

1.3. Routine Physical Maintenance

Several hours have passed since Clair began his workday. At this moment, Clair is returning to his office after lunch (we can tell from the mustard smeared on his tie). Before settling down at his desk, Clair elects to perform the physical maintenance necessary to keep the Decision in top working condition. He reaches around to the back of the Decision cabinet, and grabs something that is located right in the middle of the back.

As his hand comes back into view, you can see that he is holding a black square of foam rubber, a little dusty on one side. This is the fan filter, which sits in a holder outside the cabinet. Clair walks out of the room to the washroom. The fan filter is washed in lukewarm, mild soapy water, blotted dry, and allowed sit until it is completely dry.

Fifteen minutes later, Clair re-enters his office with the

fan filter in hand and a smile on his face. He seems to appreciate the extra goof off time that being system administrator provides him with. He reaches around to the back of the Decision, folds the foam filter in half, and fits it back into its enclosure.

Monthly maintenance completed, Clair goes back to work. The only other physical maintenance that can be done is to wipe off the cabinet with a damp cloth.

1.4. End of the Day Backup

As the end of day approaches, our camera discloses a tired and wired looking Clair: he really should do something about his caffeine addiction. Maybe acupuncture, or a few sessions with a hypnotist. Nevertheless, dependable Clair prepares to perform the daily backup ritual. He reaches for a couple of formatted floppy diskettes, (complete with Micronix file systems), inserts the first one into the floppy disk drive, and closes the door.

Clair has been logged in as an ordinary user all day, so he first needs to become the superuser, root. He uses the **su** (switch-user) program to change his user identity.

```
% su
Password:
# []
```

Now, he has the power to access any file or directory on the system.

Clair has already checked to see that all the other persons using the system have logged off with the **who** command. If he were to backup a file while someone was writing to it, that file would not be correctly backed up. It isn't really necessary that everyone be logged off, just so no one change any file during the backup process. Then, he types

```
# td -uvi / /dev/mfa
Insert a diskette and press return ->
   ...
# []
```

Clair knows that those letters after the tree-dump command (**td**) aren't mandatory, that the program can be run interactively by just entering td. Putting the options on the command line does speed things up a little, though, and Clair is getting a little impatient to go home.

The backup process itself is automatic, except for inserting diskettes and pressing return. Clair appears to be grumbling about something, but he has conscientiously backed up the system for over a year now without complaining to anyone. Maybe if he were to share the root password with someone besides that secretary, he wouldn't need to leave so late every night.

When the backup program finishes, Clair carefully labels the diskette (it only required one today) with the day's date, puts it in the bottom drawer of his desk, and carefully locks his desk.

1.5. Turning the System Off

Well, that's it for today. Clair, still logged in as the superuser after backing up, types

```
# down

Micronix is down[]
```

to bring down the operating system gracefully. Then, he turns the key on the front of the Decision to the left to turn off the power, removes the key, and sticks it in his pocket. Another day, another dollar. Clair gets up, and moves out of view.

2. CHECKING THE FILE SYSTEM

By now, you are aware that the file system is both the organization of data and the data itself on a hard disk or diskette. Assuming 100% performance of hardware, software, and the personnel using them, your file system will last forever, and never have a problem. In the real world, the people, hardware, and software operate at perhaps 95% of perfection. This means that there will be occasions that create problems for your file systems.

2.1. Things NOT To Do To Your File System

These are the things that are most likely to give your file system a headache:

1. Resetting the system without first giving the **down** command
2. Interrupting power to the computer or the disk drive
3. Copying a file to a "/dev/" disk device file that contains a file system
4. Hardware problems within the disk drive

It is a good thing to be able to recognize a hardware problem when you see one, because running fsck on faulty hardware may do more harm than good. The symptoms produced by the first