CSE 265: System and Network Administration

- Debugging
  - Learn the customer's problem
  - Find the root cause and fix it
  - Have the right tools
- Fixing things once
  - Fix things once, rather than over and over
  - Avoid the temporary fix trap
  - Learning from carpenters

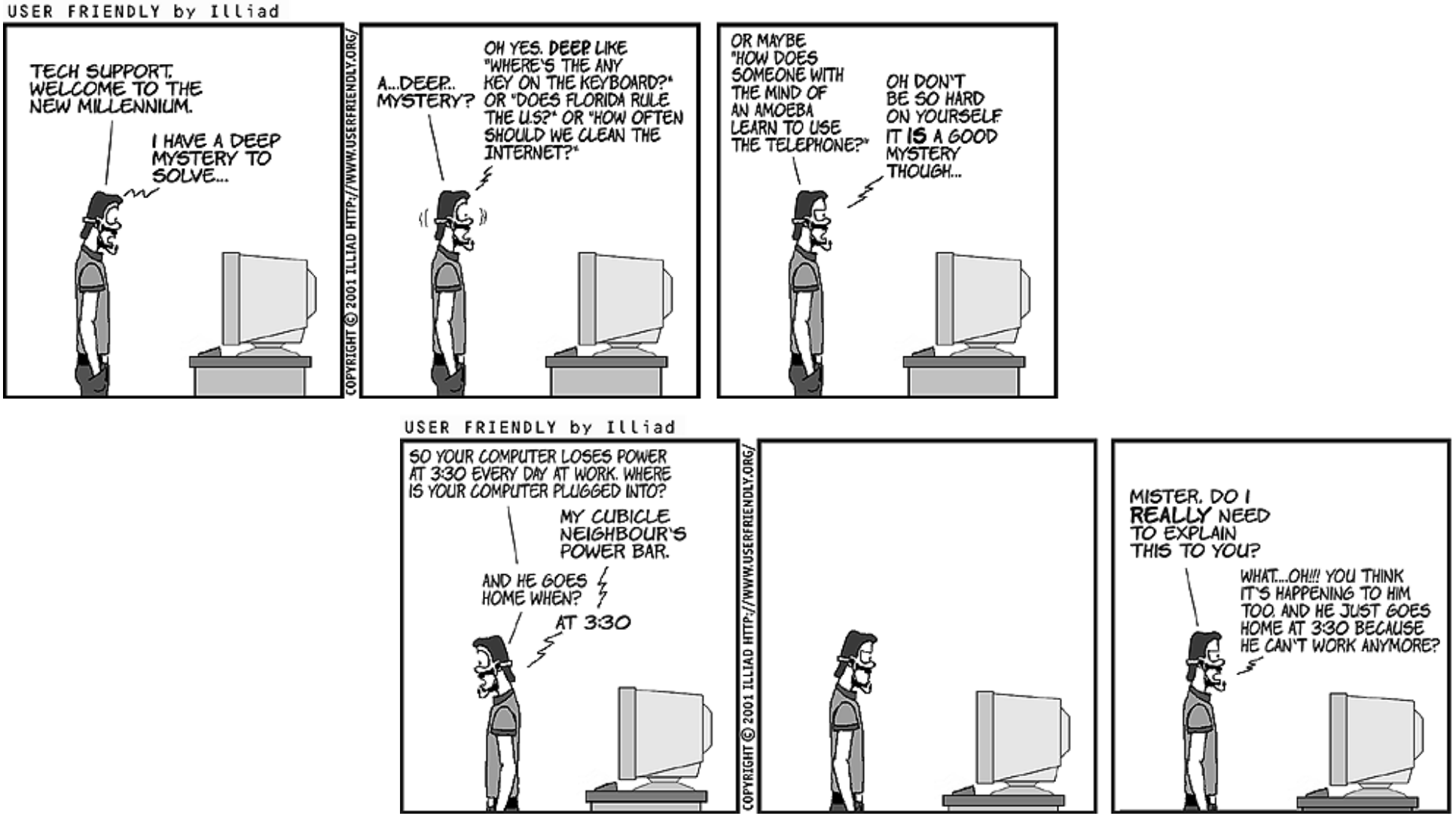
 ©2004-2016 Brian D. Davison

Learn the customer's problem

- Step one: understand (at a high level) what the user is trying to do, and what part is failing
- The customer expects a particular result from some action, but is getting something else
- Ex:
  - My mail program is broken
  - I can't reach the mail server
  - My mailbox disappeared!
- Any could be true, but the real problem could be DNS, a power failure, a network problem, etc.
- When resolved, make sure the customer agrees!

 ©2004-2016 Brian D. Davison

Learn the customer's problem



 ©2004-2016 Brian D. Davison

Example #1: tape failures

As every System Administrator knows, reliable backups are a must. Because of this my team became suitably concerned when the operators handling our central database servers started to report "tape failures." The failures soon became regular, and required regular manual intervention to keep operational.

In investigating the cause of this problem, corporate security and production floor rules forced us to depend on the operators for information. The operations staff placed the blame on the off-site tape-storage service's jostling tapes during transport, and requests for samples of failed tapes gave no indication as to the cause.

 ©2004-2016 Brian D. Davison

Example #1 continued

The root cause of the problem didn't become obvious until this had been going on for a couple of months. During a large system upgrade, my team was able to observe the operators at work.

The operations staff had been out-sourced to a low-cost contracting firm that apparently contained a large percentage of fans of the local professional hockey team. The operators were skidding the 8mm tapes across the computer-room floor like a hockey puck instead of carrying them across the floor. Adding a rule prohibiting throwing, skipping, and sliding of backup tapes quickly restored backups to a reliable state.

Tape Hockey, by Allen Peckham

 ©2004-2016 Brian D. Davison

Find the problem's cause and fix it

- Workarounds are good, but fixing the root cause is much better
  - Rebooting/restarting is a common workaround
- E.g., solution for full-disk problem is **not** to delete old log files
- Improving the speed of reboots is not really the solution either!

 ©2004-2016 Brian D. Davison

Example #2: mail problems

An ISP noticed that email service was particularly slow one day and they were getting complaints that it took up to 4 hours to deliver messages that were sent through the SMTP server.

The quick (and easy) solution would be to restart the server and flush out whatever was slowing it down. That would have masked the problem, however.

Instead, they monitored the service and noticed that they were getting repeated accesses from the same site. Hundreds, no thousands of emails flowing into us from one source. This indicated that someone was spamming through the ISP.

 ©2004-2016 Brian D. Davison

Example #2 continued

With the knowledge of their IP address, they were able to track down who they were and block them from the system thereby stopping them from spamming through it any more.

(Yes, they had spam blocking in place; this user was a customer and therefore was allowed to use the SMTP server. Their Acceptable Use Policy, however, forbade using it to send unsolicited commercial bulk email so they banned the user.)

Source: Lecture notes of Scott Heffner, Keene State College

 ©2004-2016 Brian D. Davison

Example #3: missing files

In the middle of the night, all the machines went down, with varying amounts of stuff missing.

Nobody knew what what was going on! The systems were restored from backup, and things seemed to be going OK, until the next night.

This time, Corporate Security was called in, and the admin group's supervisor was called back from his vacation (I think there's something in there about a helicopter picking the guy up from a rafting trip in the Grand Canyon).

 ©2004-2016 Brian D. Davison

Example #3 continued

By chance, somebody checked the cron scripts, and all was well for the next night...

Why? What happened:

> We have a home-grown admin system that controls accounts on all of our machines. It has a remove user operation that removes the user from all machines at the same time in the middle of the night.
>
> Well, one night, the thing goes off and tries to remove a user with the home directory '/'...

Organization: AT&T Bell Labs, Murray Hill, NJ, USA

 ©2004-2016 Brian D. Davison

How to find the cause

- Be systematic
  - Form hypotheses, test them, note the results, make changes based on those results
- Use
  - Process of elimination
  - Successive refinement
- Real problem is most often associated with the most recent change made to the host, network, or whatever is broken
  - From a lack of testing

 ©2004-2016 Brian D. Davison

Process of elimination

- Remove different parts of the system until the problem disappears
  - Problem was in last part removed
- Common technique for hardware problems
  - Swap or remove pieces until it works
- Also works for software

 ©2004-2016 Brian D. Davison

Successive refinement

- Add one new component at a time and verify that it works correctly
  - traceroute works this way
- May require examining intermediate stages of output
- For systems/processes with many components, the process of elimination and successive refinement may take a while.
  - Why? What is an alternative?

 ©2004-2016 Brian D. Davison

Have the right tools

- Diagnostic tools let you see into devices or systems to see inner workings
- Still need to interpret what you see
- Packet sniffers are easy to use
  - Understanding the protocols captured requires knowledge and training (e.g., networking courses)
- Understand how the tool works
  - It may draw the wrong conclusion
- Simple tools are often best
  - ping, traceroute, telnet

 ©2004-2016 Brian D. Davison

Take a scientific approach

- Given an unusual, recurring behavior or problem:
  - Collect data
  - Visualize [optional, but often helpful]
  - Discern patterns
  - Hypothesize source of patterns
  - Test for such sources
  - Apply solution
  - Test solution

 ©2004-2016 Brian D. Davison

End-to-end understanding helps

A customer reports that some of his files were disappearing – he had about 100MB in his home directory, and all but 2MB had disappeared.

He restored his files. A couple of days later it happened again.

This had been happening for a few weeks, but was embarrassed to tell the system admins.

Theory 1: Virus scans revealed nothing.

Theory 2: Prank, or bad cron job.

- Was given pager numbers, told to call next time
- Network sniffers were put into place

 ©2004-2016 Brian D. Davison

Example #4 continued

Happens again. Was asked what he last did? Used a lab machine to surf the Web.

Extra knowledge helps – a sysadmin remembered that Web browsers kept a cache and pruned it to stay under a certain limit (such as 2MB).

Lab workstation was misconfigured; browser found an invalid/missing cache directory and used the user's home directory instead.

 ©2004-2016 Brian D. Davison

Fix things once, rather than over and over again

- When something seems trivial or temporary, it is easy to ignore it, or use a quick fix
- A little effort will often pay for itself
- Rule: Fix it once
  - Corollary A: Fix the problem permanently
  - Corollary B: Leverage what others have done – don't reinvent the wheel
  - Corollary C: Fix a problem for all hosts at the same time

 ©2004-2016 Brian D. Davison

Avoid the temporary fix trap

- Sometimes a complete fix is impossible in that situation
- It's important that a temporary fix be followed by a permanent one
  - Record the actions taken for a temporary problem!
  - Put the creation of a full solution on a trouble ticket!
- Fixing the same small things is habit-forming – we get good at the keystrokes needed!
- We get used to the quick fix, and don't realize how much time we have lost as a result.

 ©2004-2016 Brian D. Davison

Mailing list example

- Running a mailing list seems easy.
  - E.g., automated subscribe and unsubscribe
- Book author ran many mailing lists, and had to deal with bounced messages.
  - Dealing with bounces takes time. Wrote scripts to help manage it – collect bounces, figure out who was bouncing, delete subscriber if error persisted. Still took ~1 hour a day!
- Better solution was other software that handled bounces or made list owners deal with them
  - He ignored bounces for a week; stayed late to install new software without interruption. Cost: 5 hours; savings: 4 hours per week.

 ©2004-2016 Brian D. Davison

Learning from carpenters

- Carpenters copy a length by re-using the original piece over and over again
  - Re-use working scripts rather than re-writing them
  - Use command-line shell shortcuts rather than re-typing
  - A little extra care is a small price compared to the potential damage of a mistake.

 ©2004-2016 Brian D. Davison

Example #5: rm folly

My mistake on SunOS (with OpenWindows) was to try and clean up all the '.*' directories in /tmp. Obviously "rm -rf /tmp/*" missed these, so I was very careful and made sure I was in /tmp and then executed

"rm -rf ./.*"

I will never do this again. If I am in any doubt as to how a wildcard will expand I will echo it first.

Organization: DataCAD Ltd, Hamilton, Scotland

 ©2004-2016 Brian D. Davison

Summary

- Understand the problem
- Fixes should be permanent
- Leverage others' fixes
- Fixes should be global
- Test your solution!

 ©2004-2016 Brian D. Davison