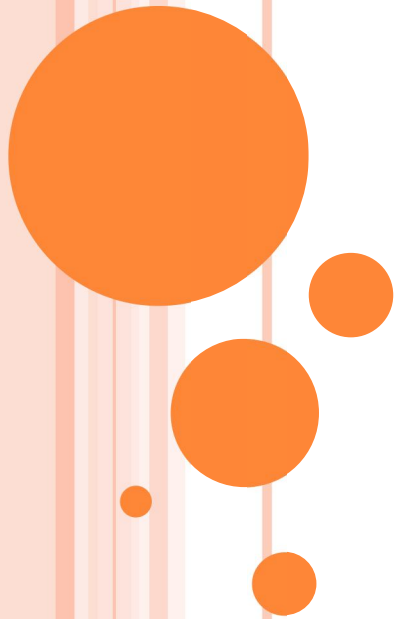


CONFIGURATION AND MAINTENANCE



INTRODUCTION

- Configuration management is the administration of state in hosts or network hardware.
- Maintenance is simply configuration in the face of creeping decay. All systems tend to decay into chaos with time.
- **Principle 37 (Disorder).** Systems tend to a state of disorder unless a disciplined policy is maintained, because they are exposed to random noise through contact with users.
- **Principle 38 (Equilibrium).** Deviation from a system's ideal state can be smoothed out by a counteractive response. If these two effects are in balance, the system will stay in equilibrium.



SYSTEM CONFIGURATION POLICY

- A system policy also has the role of summarizing the attitudes of an organization to its members and its surroundings and often embodies security issues.
- **Principle 39 (Policy).** *A clear expression of goals and responses prepares a site for future trouble and documents intent and procedure. Policy should be a protocol for achieving system predictability*
- policy documents acceptable behavior, but it should also document what response is appropriate in a crisis
- *System Policy includes*
 - Organization:
 - Users
 - Network
 - Mail
 - WWW
 - Printing
 - Security
 - Privacy



METHODS: CONTROLLING CAUSES AND SYMPTOMS

- **Principle 40 (Standardized methods offer predictability).** *Replacing direct low-level control with configurable high-level interfaces increases standardization and thus predictability. If the methods are implemented correctly, this improves quality control; if they are flawed, it becomes a systematic error, but with only a single point of failure and repair.*
- **Principle 41 (Symptoms and cause).** *Inadequate control over infrastructure demands a strategy of short-term symptom relief in lieu of a more permanent reparation at source.*



CHANGE MANAGEMENT

- Configuration management deals with the management of significant changes, e.g. upgrades, redesign and replacement.
- Planning changes of infrastructure can be dealt with using two general strategies:
 - Deconstruction followed by reconstruction.
 - Change of policy description followed by convergence to a new state.



CLOCK SYNCHRONIZATION

- One of the fundamental ways to ensure reliability in the system is to keep all the clocks synchronized.
- One option for most Unix-like systems is the `rdate` command, which sets the local clock according to the clock of another host.



○ #!/bin/sh

#

Fake rdate script for linux - requires ssh access on
server

#

echo Trying time server

DATE='/bin/su -c '/usr/bin/ssh time-server date' remote-
user'

echo Setting date string...

/bin/date --set="\$DATE"



CLOCK SYNCHRONIZATION

- A more reliable way of keeping clocks synchronized, which works both for Unix and for Windows, is to use the NTP protocol, or network time protocol.
- Two configuration files are needed to set up this service on a Unix-like host: **/etc/ntp.conf** and **/etc/ntp.drift**. **/etc/ntp.conf** looks something like this, where the IP address is that of the master time-server, whose clock we trust:

driftfile /etc/ntp.drift

authdelay 0.000047

server 128.39.89.10



AUTOMATION OF HOST CONFIGURATION

TOOLS FOR AUTOMATION

system administration tools developed are based either on the idea of

- *control interfaces (interaction between administrator and machine to make manual changes)*
- on the cloning of existing reference systems
- provide ways of cloning machines by distributing files and binaries from a central repository.



TOOLS FOR AUTOMATION

- **Tivoli** is probably the most advanced and wide-ranging product available.
- It is a Local Area Network (LAN) management tool based on CORBA and X/Open standards.
- It is a commercial product, advertised as a complete management system to aid in both the logistics of network management and an array of configuration issues.
- As with most commercial system administration tools, it addresses the problems of system administration from the viewpoint of the business community, rather than the engineering or scientific community.



TOOLS FOR AUTOMATION

- Tivoli admits bidirectional communication between the various elements of a management system.

DRAWBACK

- It focuses on application-level software rather than core system integrity.
- Also it lacks abstraction methods for coping with real-world variation in system setup.



MONITORING TOOLS

- Unfortunately there is currently no way of capturing the details of every action performed by the local host.
- The best one can do currently is to watch system logs for conspicuous error messages.
- Programs like **SWATCH** perform this task.
- Another approach is the analysis of system logs at a statistical level.
- Rather than looking for individual occurrences of log messages, one looks for



SCRIPTING LANGUAGE

- Perl was the first of a group of scripting languages including **python**, **tcl** and **scheme**, to gain acceptance in the Unix world. It has since been ported to Windows operating systems also
- The Perl language is a curious hybrid of C, Bourne shell and C-shell, together with a number of extra features which make it ideal for dealing with **text files** and **databases**.
- Perl is semi-compiled at runtime, rather than interpreted line-by-line like the shell, so it gains some of the advantages of compiled languages, such as syntax check before execution and so on. This makes it a safer and more robust language.



POLICY DECISIONS

It includes :-

- *Determine the system policy*
- *Sysadmin team agreement*
- *Expect the worst*
- *Educate users in good and bad practice*
- *Special users-like do some users need extra attention , extra resources etc.*



GENERAL PROVISION

- Educate users by posting information in a clear and friendly way.
- Make rules and structure as simple as possible, but no simpler.
- Keep valuable information about configuration securely, but readily, available.
- Document all changes and make sure that co-workers know about them
- Be careful while making changes.
- Be aware of system limitations, hardware and software capacity.



CFENGINE

Cfengine is about

- (i) defining the way you want all hosts on your network to be set up (configured)
- (ii) writing this in a single 'program' which is read by every host on the network
- (iii) running this program on every host in order to check and possibly fix the setup of the host.



CFENGINE

- A cfengine program is a free format text file, usually called cfagent.conf and consisting of declarations of the form

action-type

classes::

list of actions

- The action type tells cfengine what the commands which follow do. The action type can be from the following list.

**binservers, broadcast, control, copy, defaultroute,
directories, disable, editfiles, files, groups,
homeservers,
ignore, import, links, mailserver, miscmounts,
mountables,
processes, required, resolve, shellcommands, tidy,
unmount**

