Open Source Security Tools in a High Performance Environment

Liam Forbes

May 24, 2002

CUG Summit 2002
Manchester, England

Topics

- Why Open Source Tools?
- Third Party Software Installation Procedure
- Issues Porting Open Source to HPC
- Open Source Security Tools In Use
- Open Source Security Tools Under Evaluation
- Conclusion

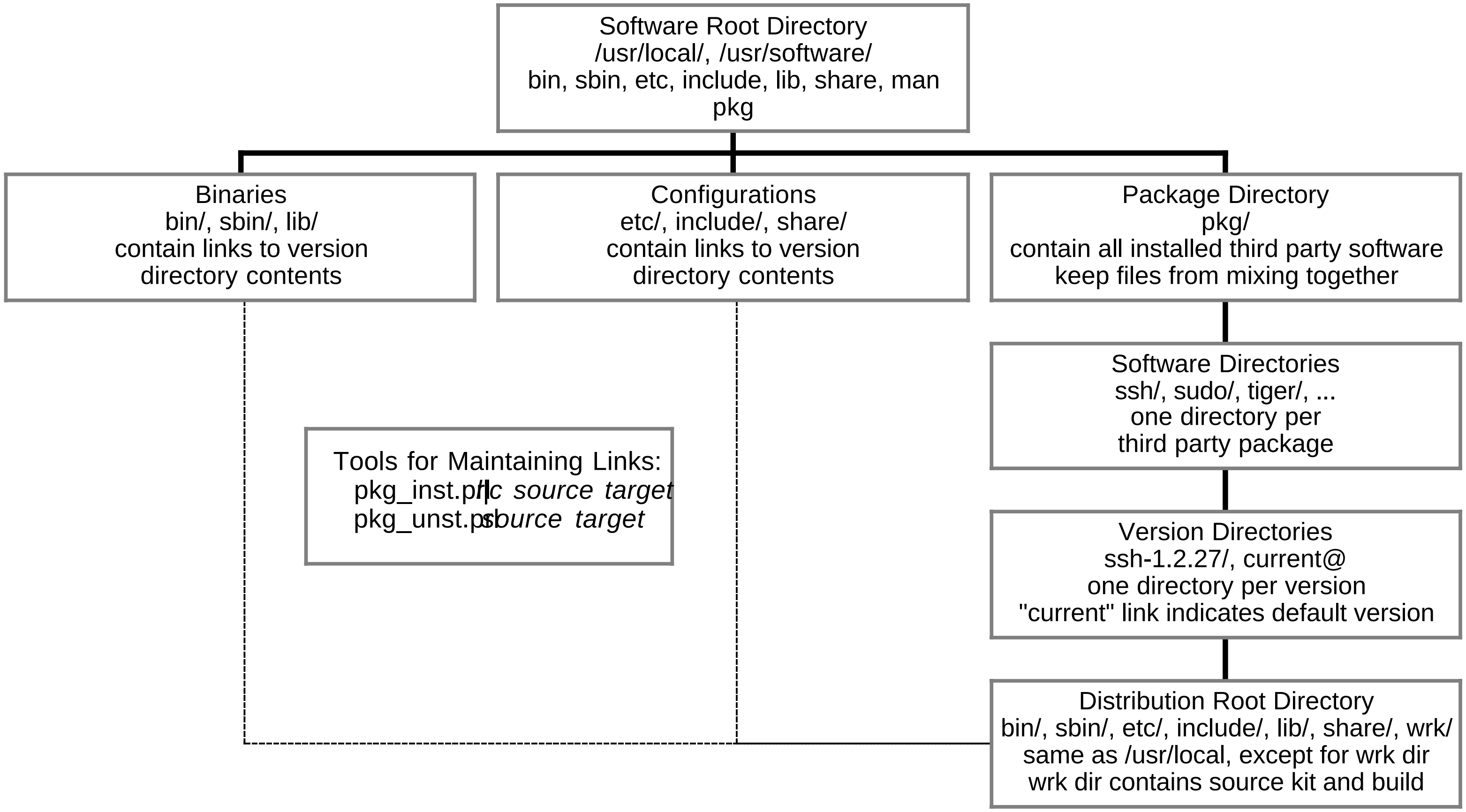
Open Source Benefits

- Provide Missing Security Features
- Provide Similar Features & Interfaces Across Platforms
- Unique Support Mechanism
- Good Reliability
- Great Flexibility

Open Source Drawbacks

- Porting Effort
- Unique Support Mechanism

Third Party Software Installs



Software Root Directory
/usr/local/, /usr/software/
bin, sbin, etc, include, lib, share, man
pkg

Binaries
bin/, sbin/, lib/
contain links to version
directory contents

Configurations
etc/, include/, share/
contain links to version
directory contents

Package Directory
pkg/
contain all installed third party software
keep files from mixing together

Software Directories
ssh/, sudo/, tiger/, ...
one directory per
third party package

Version Directories
ssh-1.2.27/, current@
one directory per version
"current" link indicates default version

Distribution Root Directory
bin/, sbin/, etc/, include/, lib/, share/, wrk/
same as /usr/local, except for wrk dir
wrk dir contains source kit and build

Tools for Maintaining Links:
pkg_inst.pl *-l|c source target*
pkg_unst.pl *source target*

SSH/OpenSSH for Unicos & Unicos/mk [1]

- Add headers and libraries for UDB and user environment (ia.h, tmpdir.h)
- Modify memory allocation based on data type sizes - some data types didn’t exist
- Modify session startup to correctly handle MLS privileges
- Modify authentication failure routines to properly update UDB
- Modify session shutdown to handle end of job signal(s) (WJSIGNAL)

SSH/OpenSSH for Unicos & Unicos/mk [2]

- Manage configuration files
  - Choose proper defaults
  - Setup proper ACLs
  - Compare and sync with other platforms
  - Document - especially differences
- Distribute host keys securely

Open Source Security Tools Installed at ARSC

- Kerberos5 1.1 (w/ localmods; latest 1.2)
- LPRng 3.5.3 (latest 3.8.9)
- SSH 1.2.27 (w/ localmods; latest 3.1)
- OpenSSH 3.0, 3.1 (w/ localmods; latest 3.1)
- Sudo 1.6.6 (latest 1.6.6)
- Swatch 2.2 (w/ localmods; latest 3.0.4)
- TCPWrappers 7.6 (latest 7.6)
- Tiger 2.2.4 (w/ localmods; latest 2.2.4)

Open Source Security Tools Under Evaluation at ARSC

- Tripwire - file integrity checker (localmods)
- Nessus - vulnerability assessment (localmods)
- SyslogNG - replace syslog (localmods)
- TARA - to replace Tiger (localmods)

Conclusion [1]

- Open Source Works, Even in a High Performance Environment with Unique Platforms.
- Each Tool Needs to:
  - Fulfill a need.
  - Improve the administrator’s life.
  - Improve users’ lives, or at least minimally interfere.

Conclusion [2]

- If using open source tools;
- If modify code to port to a new platform, fix a bug, add a feature;

- Submit Modifications Back to the Original Developer or
- Make the Patch(es) Available.

Random Thought

- What if IRIX or Unicos were Open Source?
  - Unicos on the Palm Pilot!
  - Supply fix when open case/SPR