Using YUM to Increase Linux Security

Connie Sieh
csieh@fnal.gov

Troy Dawson
dawson@fnal.gov

Fermi National Accelerator Laboratory
May 25, 2004
HEPiX

History

- Originally released first Fermi Linux in August 1998
- Network based security bugs but no way to install errata easily
- Researched tools to automatically install errata RPMs
- Found and tested AUTORPM

History continued

- Worked well until RedHat started to release errata
  - circular dependencies
  - new dependencies
- Ftp server load was increasing rapidly because AUTORPM downloaded every rpm every time it checked for updates

Requirements

- Easy for end user to use
  - Large number of self administered desktops
- Low overhead on servers(autorpm was high)
- Continuing support
- Easy to administer
- Ability to install rpms with dependencies
- Fixes shortfalls of AUTORPM
  - New dependencies
  - Circular dependencies
  - Parses output of rpm, breaks when output changes

Choices

- Urpmi
- up2date/current
- Autoupdate
- Autorpm 2.0
- Apt-rpm
- YUM (YellowDog Updater Modified)

up2date/CURRENT

- RedHat releases source to up2date – client
- RedHat did not release source to server side
- CURRENT is a open source implementation of server side
- Did not scale as admitted by author at time of research

Urpmi

- Mandrake
- Did not work on RedHat based releases without much porting
- Gui available
- Handles cdrom RPMs well

Autoupdate

- Written in Perl
- Mature
- Handles new dependencies
- Parses output of rpm
- Command line only , no Gui

Autorpm 2.0

- Written in Perl
- Changed syntax of command line and config file from version 1
- Parses output of rpm
- Never got it to handle dependencies
- Downloads full rpm to get meta data each time

APT-RPM

- Based on APT from Debian
- Written and supported by Connectiva
- Written in C++
- At time of this research it parsed output of command line, current version calls rpmlib
- Gui available
  - synaptic
- RPM database needs to be **consistent**
  - Issue if RPM's were installed with --nodeps or --force

APT-RPM continued

- Handles dependencies well
- Command line options can be confusing
- Meta data prebuilt
- RPMS must reside in specific directories requiring extra steps to create meta data
- Contains embedded scripting language
  - Lua
  - Did not exist when original decision was made
- Handles cdrom RPMs well
- Expected to be winner, but at beginning of research yum did not exist

YUM (YellowDog Updater Modified)

- Written in python
- Very modified fork of YUP (Yellowdog UPdater)
- Uses RedHat code from Up2date and anaconda
- Active code development
  - was in beta when we first tested it
  - Developer made patches same day as bug reports
  - Developer is from Duke University Physics Dept

YUM -continued

- Very easy and powerful to use
  - Yum install <package name>
  - Yum groupinstall <groupname>
- Meta data prebuilt, only overhead is download
- Handles new dependencies
- Command line only , no Gui
- Meta data can reside anywhere, client configfile specifies location

YUM implementation

- Kernels have to be done manually with help from yum
- Ability to **force** new RPMs if needed
- Provide cron script
  - Random delays start to not overwhelm servers
  - Load balances between servers
- Provide **check-update** only cron script which only sends mail
- Automatically installed, user has to turn off if not wanted

YUM summary

- Ftp overhead is much less
- Not going to **jinx** security record by saying how good this has been
- Easy for end user to use
- Really helps users to upgrade new kernels