A Control Point for Reducing Root Abuse of File-System Privileges

Glenn Wurster, Paul C. van Oorschot
School of Computer Science
Carleton University, Canada

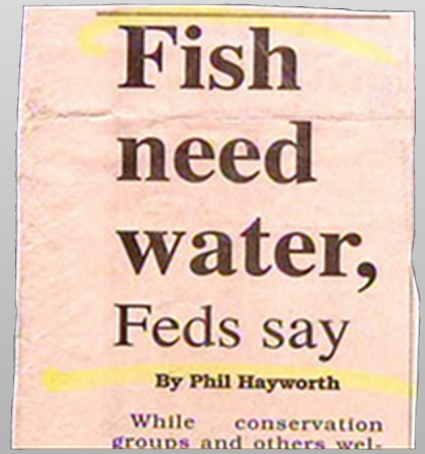
6 Oct 2010

Problem

Root privileged processes can arbitrarily modify the system

Solution

Don’t run as root


Fish need water, Feds say

By Phil Hayworth

While conservation groups and others wel-

Re-phrasing the requirements

On the desktop, we should treat two applications as mutually untrustworthy.

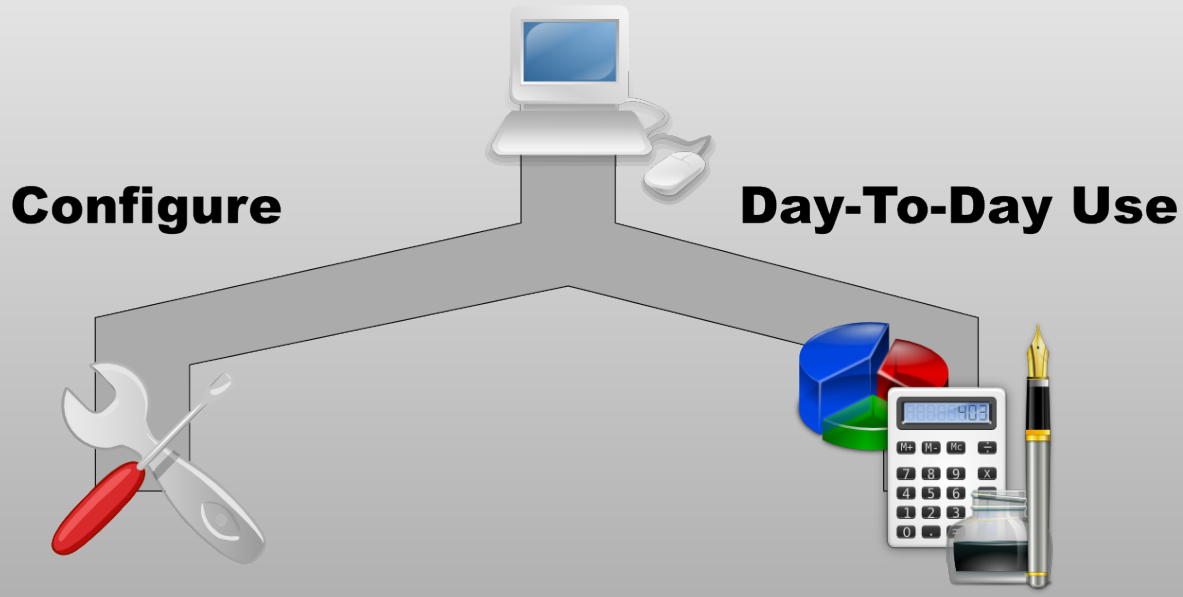
1. During install, upgrade, uninstall, and run-time.
2. The paper concentrates only on the file-system.
   - Allow file-system reads, but don't allow modifications.

Other Approaches to Divide Root

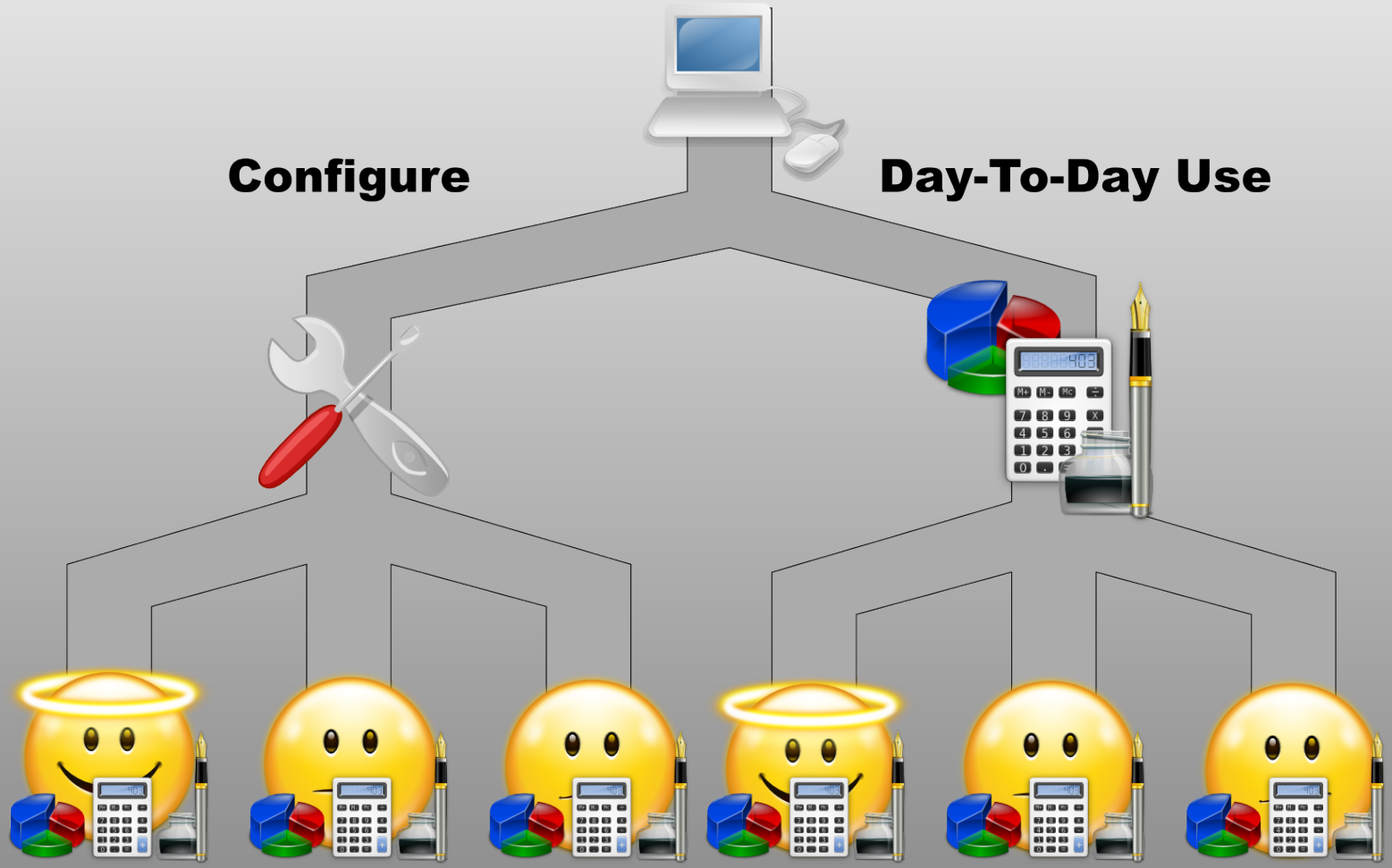
e.g., SELinux

- dpkg given almost total control over the file-system

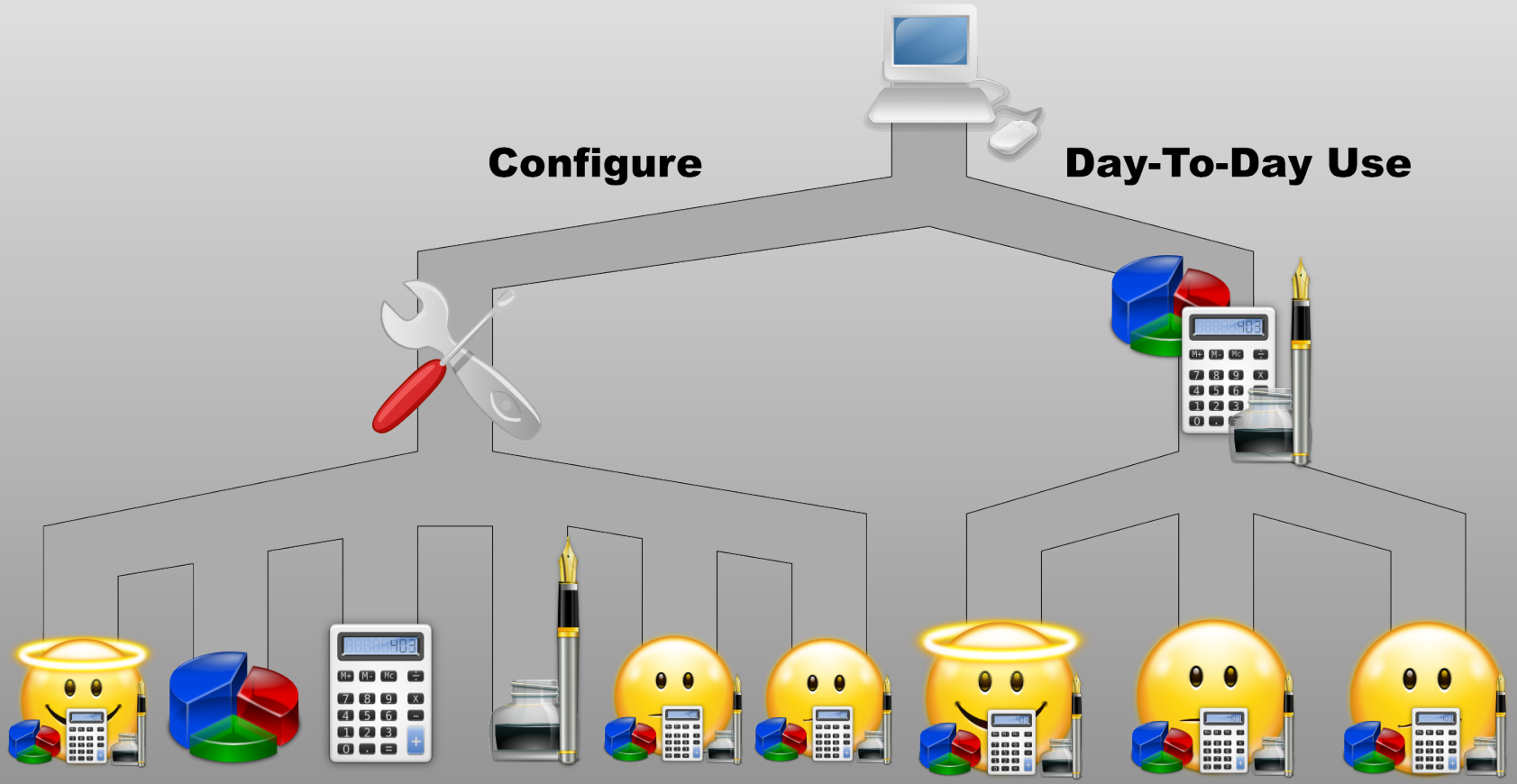
Two States



Two States, Many Users



Two States, Many Applications

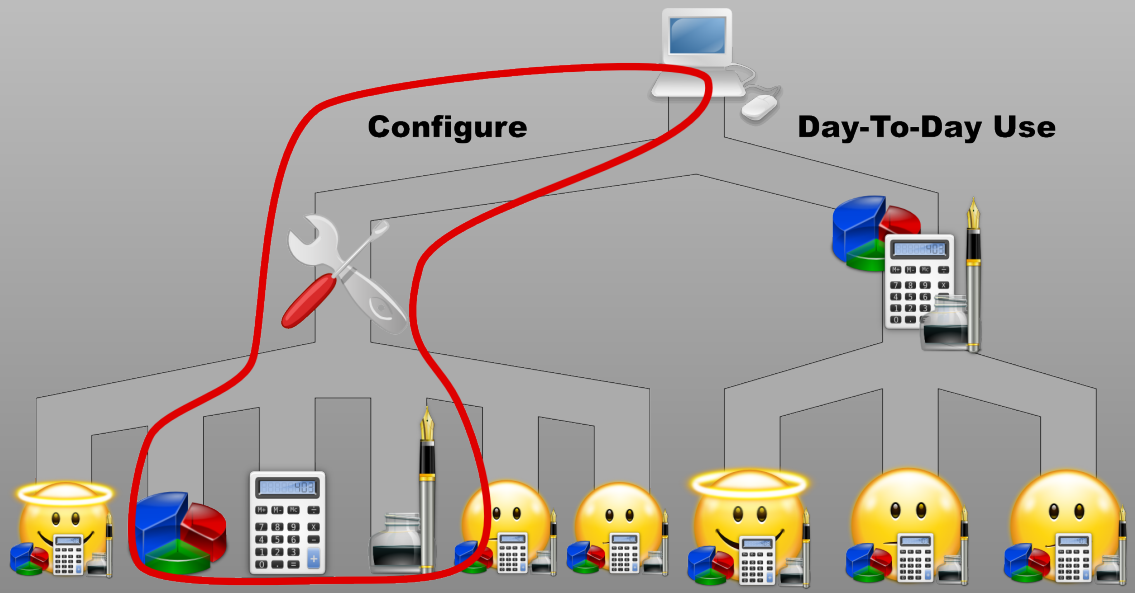


Our focus

Configuration related files:

1. Modified during configuration, not during day-to-day use
2. We focus on *system-wide* configuration files

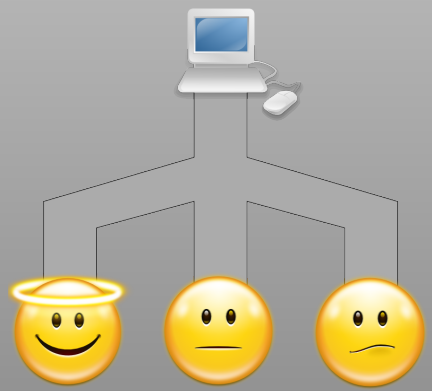
Files most commonly modified through install, upgrade, and uninstall

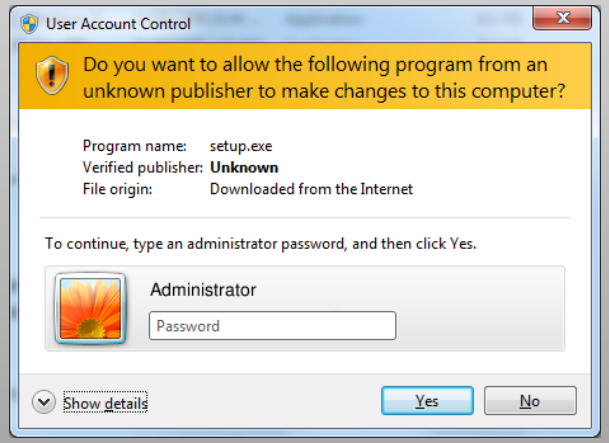


Application Installers

Run a Script or Binary

1. Provided by application author
2. Usually run as Administrator
3. e.g., `make install`, self-extracting ZIP

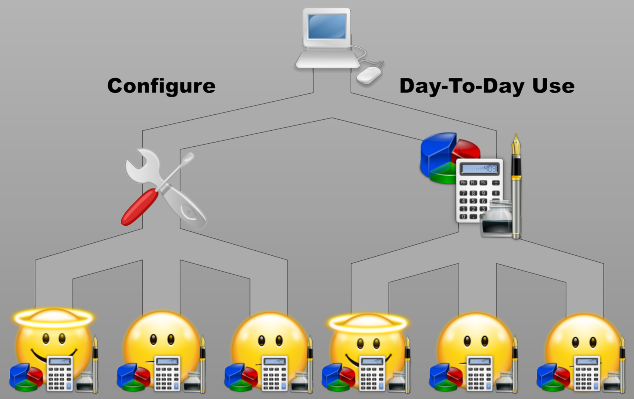


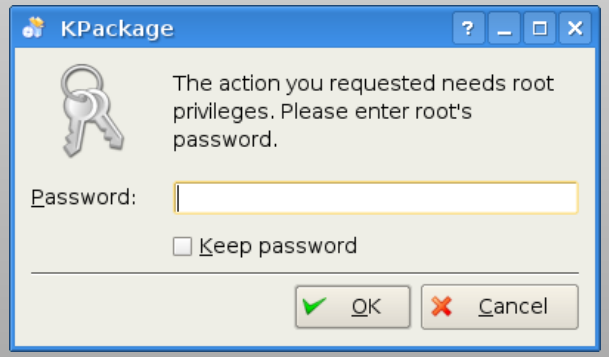


Application Packages

```
sudo apt-get install <package>
```

1. Typically, become root and run package manager
2. Package manager runs scripts in package

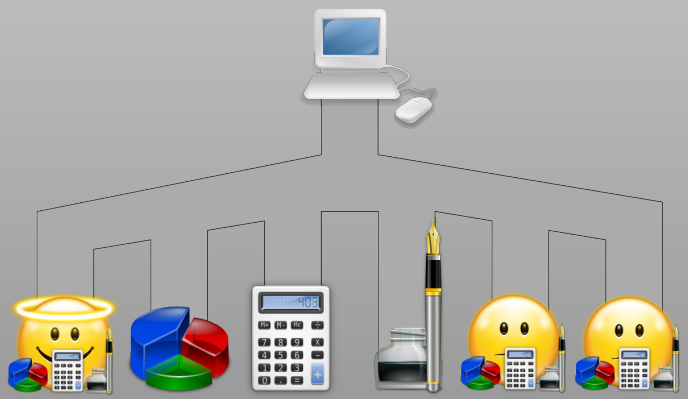




Application Bundles

Drag and Drop

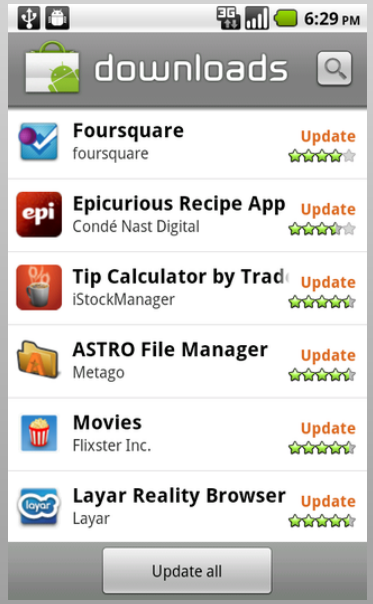
1. Drag to the destination folder
2. No scripts run during install



Google Android

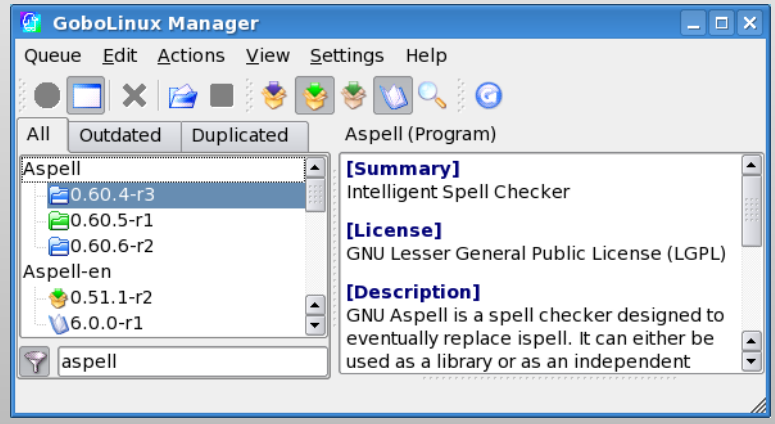
Self Signing

1. Isolate update to just the package
2. No scripts run during install



GoboLinux

1. Don’t modify files during upgrade
2. Redesign the file-system hierarchy

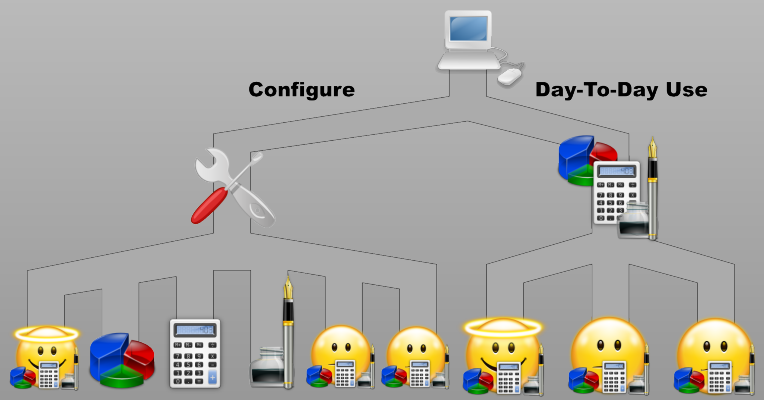


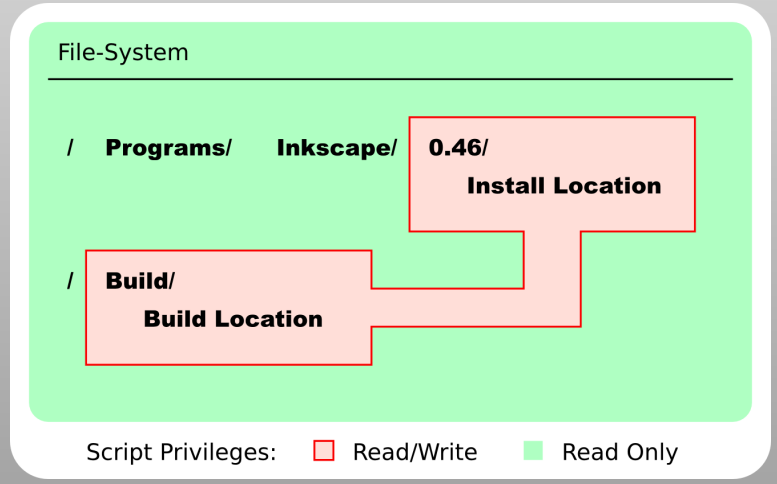

```
/Programs] ls -l OpenOffice
total 8
drwxr-xr-x  9 root root 4096 2005-09-22 01:07 1.1.4
drwxr-xr-x  3 root root 4096 2005-09-23 04:36 2.0
lrwxrwxrwx  1 root root    5 2005-09-23 04:36 Current -> 2.0

/Programs] ls -l GTK+
total 12
drwxr-xr-x  10 root root 4096 2005-10-02 01:39 1.2.10
drwxr-xr-x   9 root root 4096 2005-08-21 05:48 2.6.7
lrwxrwxrwx   1 root root    6 2005-10-02 01:39 Current -> 2.6.7
drwxr-xr-x   4 root root 4096 2005-10-02 01:39 Settings
```

GoboLinux - Restricting Scripts

Restricting Scripts

1. Script has write access to build source and install destination



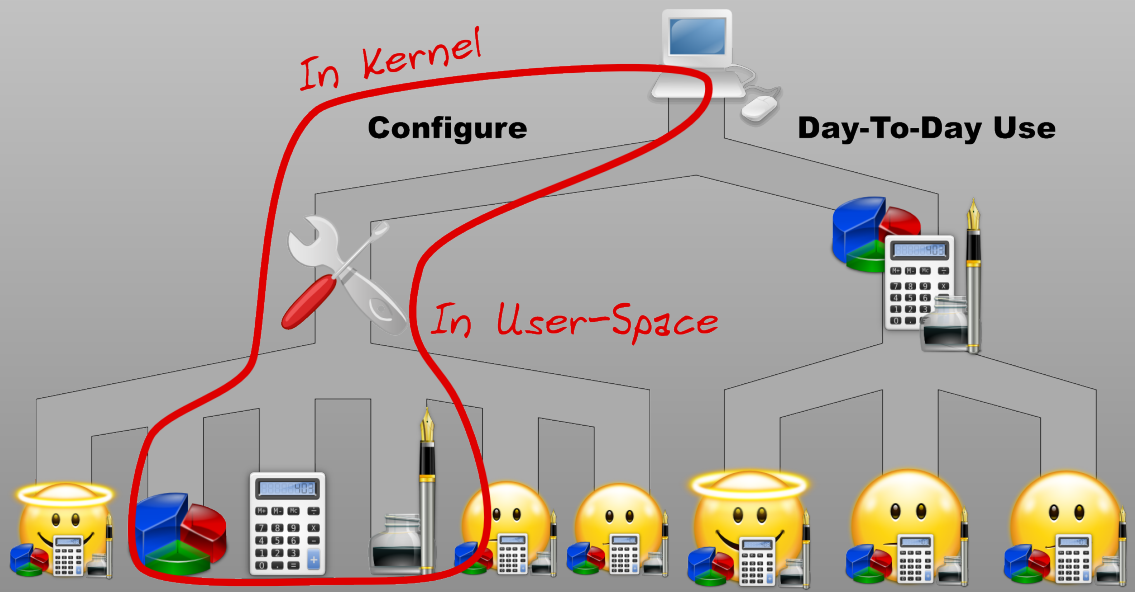


Application Installer Goal

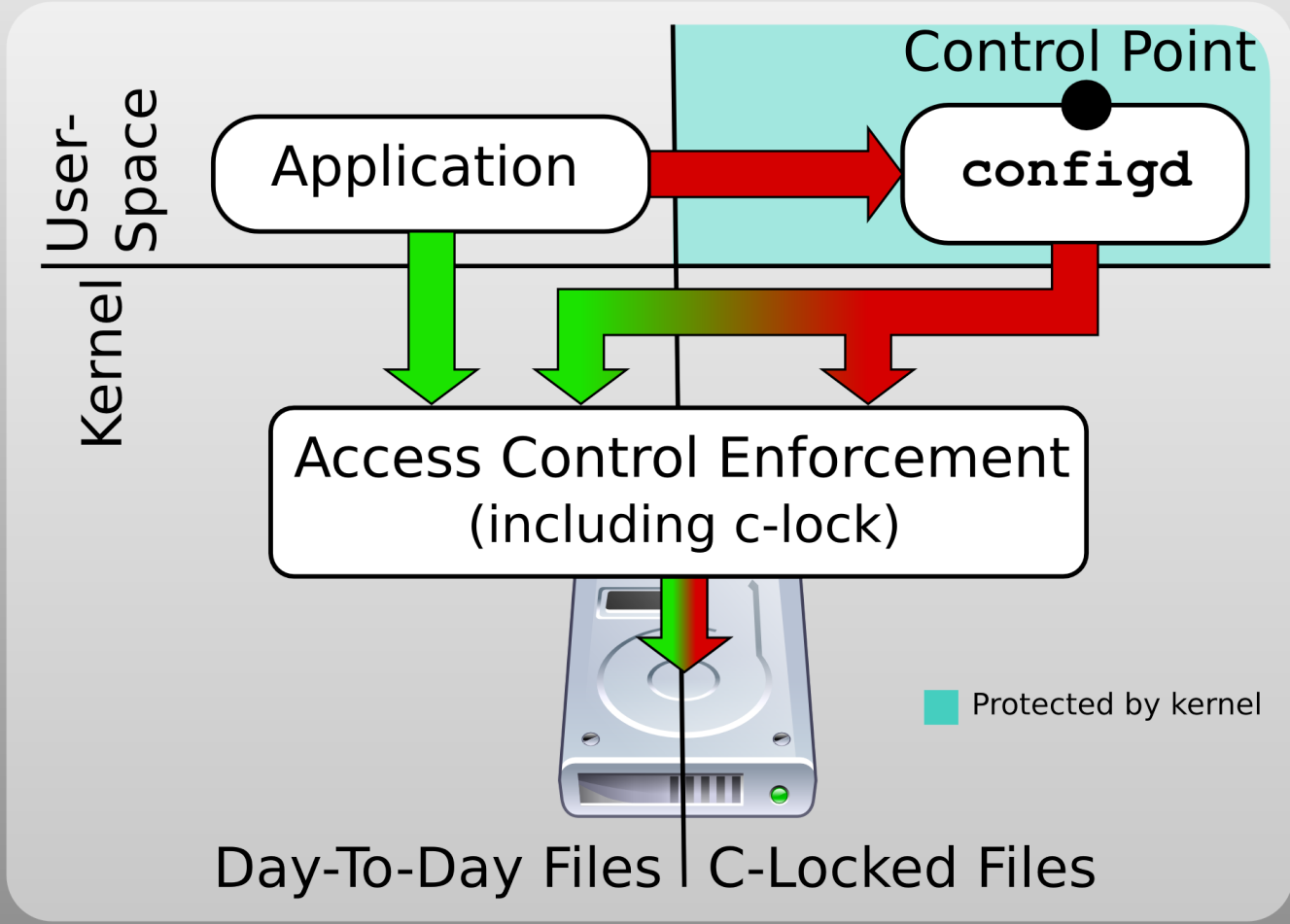
Method	Upgrade	Scripts	FS Hierarchy Agnostic	Encapsulates	Config Separation	User Friendly
Installer	✓	✓	✓			✓
Package	✓	✓	✓		✓	✓
Bundle	✓			✓		✓
Android	✓			✓	✓	✓
GoboLinux		✓		✓	✓	✓
Goal	✓	✓	✓	✓	✓	✓

Breakdown of Separation

1. Configuration related files:
   - Identified as *c-locked*, protected by kernel
2. Encapsulating configuration of applications:
   - Delegated to a user-space app called `configd`



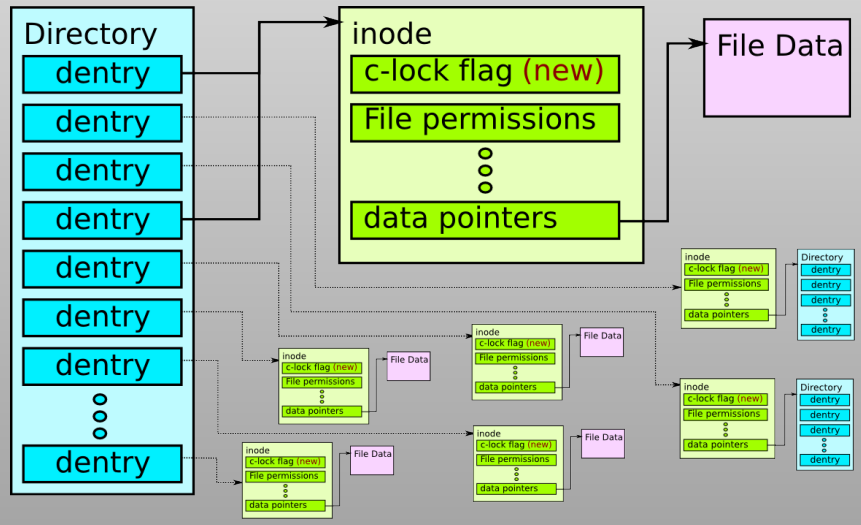
The Control Point



What can be c-locked?

Store the *c-locking* flag in the inode, protecting:

1. Files
2. Symbolic Links
3. Hard Links
4. Directories



The Prototype: GoboLinux + Debian Linux

1. Files in the package are segregated by Debian's dpkg
2. Scripts are restrained using an approach similar to GoboLinux
3. File-system hierarchy is same as standard Debian

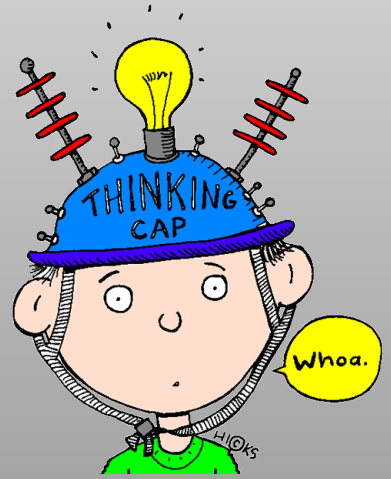
The Prototype: Restricting Applications

Restricting Installers

1. We likely don't have a custom security policy for the program being installed
2. We're not working with security experts

Enforcement

1. Continue enforcement past install
2. Any application gaining root should not be able to modify the system



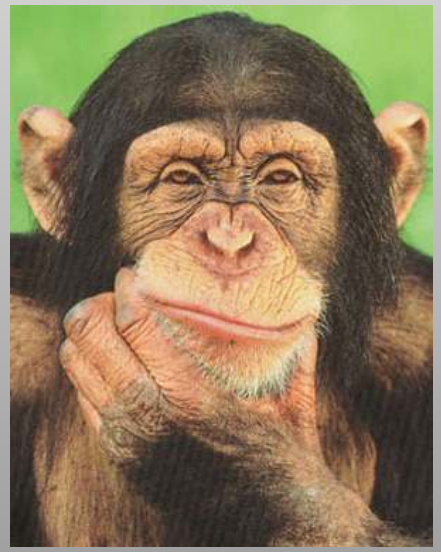
Philosophizing

Two options for restricting installers

1. Don’t run installers as root; or
2. Don’t give root all the privileges it currently gets

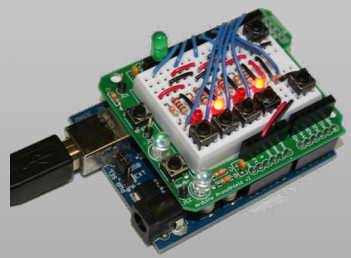
Shifting to not run installers as root

1. Users automatically become root to install
2. Applications still sometimes get root privileges
3. ’Root’ does not distinguish between configuration and day-to-day use



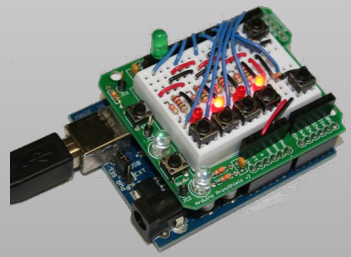
Prototype Implementation

1. Extended the Linux kernel to enforce *c-locked* flag
   - Used extended attribute functionality
   - Any file in a package is marked as *c-locked*
2. Extended dpkg to work with `configd`
3. Ran install scripts with a restricted UID



Prototype Evaluation

1. Performance overhead $\leq 4.8\%$
2. Malware prevented from modifying core *c-locked* system binaries
3. Satisfied design goals



A Control Point for Reducing Root Abuse of File-System Privileges

Glenn Wurster, Paul C. van Oorschot
http://ccsl.carleton.ca

Slide References

Projects:

- http://www.debian.org/
- http://www.gobolinux.org/

Images:

- http://www.mypointless.com/2009/05/more-headlines-of-obvious.html
- http://en.wikivisual.com/index.php/Key_(lock)
- http://zarious.deviantart.com/art/Spy-vs-Spy-WallPaper-2560X1024-115603200
- http://websvn.kde.org/trunk/kdesupport/oxygen-icons/scalable/apps/
- http://arstechnica.com/open-source/reviews/2010/07/android-22-froyo.ars/
- http://www.android.com/media/
- http://www.directindustry.com/prod/norma-group/exhaust-pipe-clamp-15287-33925.html
- http://www.codeproject.com/kb/WPF/TheWpfThoughtProcess.aspx
- http://www.multiplaying.net/2009/07/29/slurms-pondering-of-the-day4/
- http://www.cs.gettysburg.edu/~tneller/mazes/oskar4bit/arduino.html