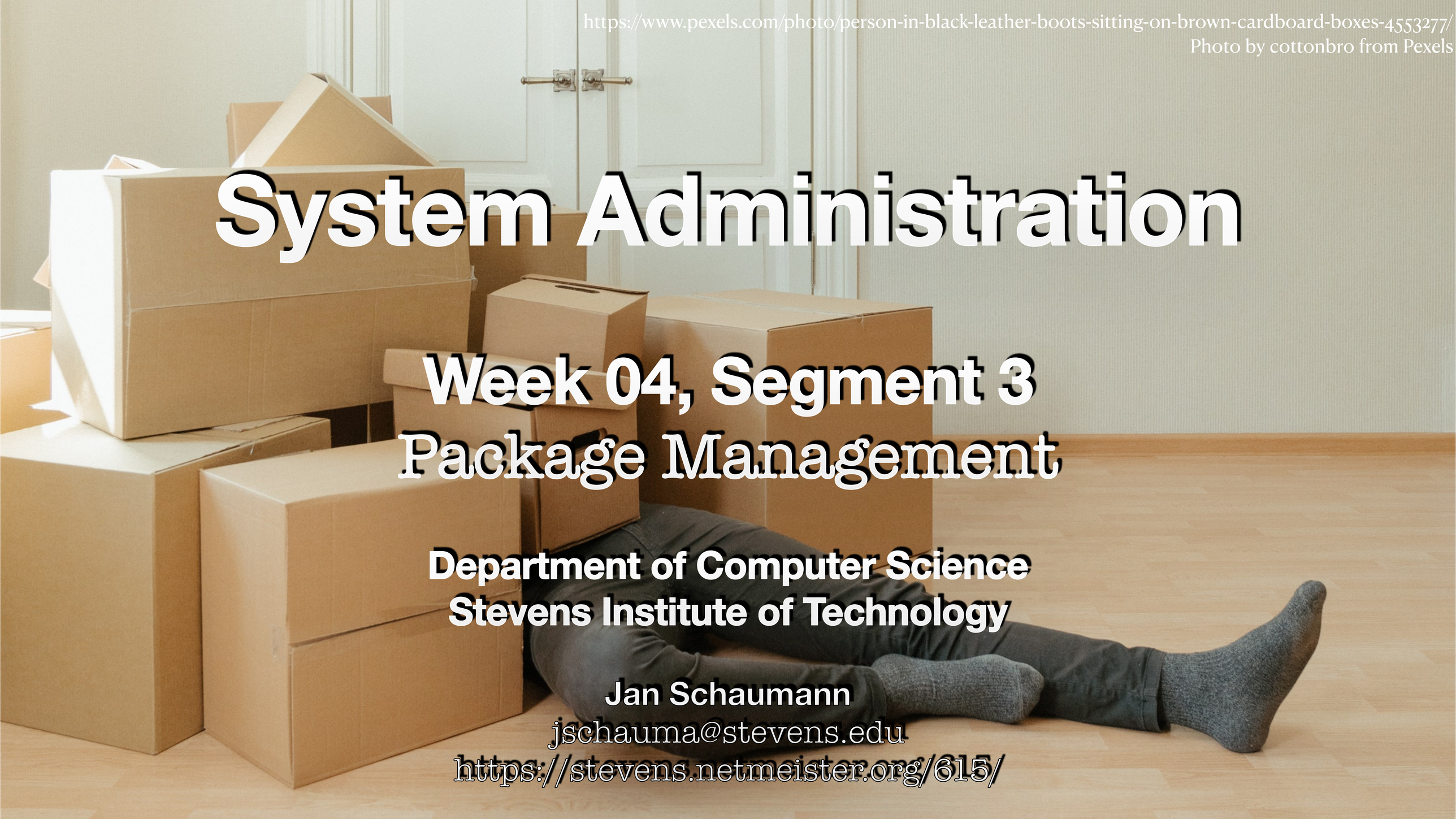


https://www.pexels.com/photo/person-in-black-leather-boots-sitting-on-brown-cardboard-boxes-4553277/
Photo by cottonbro from Pexels

System Administration

Week 04, Segment 3
Package Management

Department of Computer Science
Stevens Institute of Technology

Jan Schaumann
jschauma@stevens.edu
https://stevens.netmeister.org/615/

System Software vs. Third Party Software

Example	System / OS	3rd Party / Add-on
kernel		
drivers		
firmware		
libc		
shell		
`ssh(1)` / `sshd(8)`		
mail server		
http server		
database		
python		

System Software vs. Third Party Software

Example	System / OS	3rd Party / Add-on
kernel	✅	❌
drivers		
firmware		
libc		
shell		
`ssh(1)` / `sshd(8)`		
mail server		
http server		
database		
python		

System Software vs. Third Party Software

Example	System / OS	3rd Party / Add-on
kernel	✅	❌
drivers	✅	✅
firmware		
libc		
shell		
`ssh(1)` / `sshd(8)`		
mail server		
http server		
database		
python		

System Software vs. Third Party Software

Example	System / OS	3rd Party / Add-on
kernel	✅	❌
drivers	✅	✅
firmware	❌	✅
libc		
shell		
`ssh(1)` / `sshd(8)`		
mail server		
http server		
database		
python		

System Software vs. Third Party Software

Example	System / OS	3rd Party / Add-on
kernel	✅	❌
drivers	✅	✅
firmware	❌	✅
libc	✅	❌
shell		
`ssh(1)` / `sshd(8)`		
mail server		
http server		
database		
python		

System Software vs. Third Party Software

Example	System / OS	3rd Party / Add-on
kernel	✅	❌
drivers	✅	✅
firmware	❌	✅
libc	✅	❌
shell	✅	✅
`ssh(1)` / `sshd(8)`		
mail server		
http server		
database		
python		

System Software vs. Third Party Software

Example	System / OS	3rd Party / Add-on
kernel	✅	❌
drivers	✅	✅
firmware	❌	✅
libc	✅	❌
shell	✅	✅
`ssh(1)` / `sshd(8)`	✅	✅
mail server		
http server		
database		
python		

System Software vs. Third Party Software

Example	System / OS	3rd Party / Add-on
kernel	✅	❌
drivers	✅	✅
firmware	❌	✅
libc	✅	❌
shell	✅	✅
`ssh(1)` / `sshd(8)`	✅	✅
mail server	✅	✅
http server		
database		
python		

System Software vs. Third Party Software

Example	System / OS	3rd Party / Add-on
kernel	✅	❌
drivers	✅	✅
firmware	❌	✅
libc	✅	❌
shell	✅	✅
`ssh(1)` / `sshd(8)`	✅	✅
mail server	✅	✅
http server	??	✅
database		
python		

System Software vs. Third Party Software

Example	System / OS	3rd Party / Add-on
kernel	✅	❌
drivers	✅	✅
firmware	❌	✅
libc	✅	❌
shell	✅	✅
`ssh(1)` / `sshd(8)`	✅	✅
mail server	✅	✅
http server	??	✅
database	❌	✅
python		

System Software vs. Third Party Software

Example	System / OS	3rd Party / Add-on
kernel	✅	❌
drivers	✅	✅
firmware	❌	✅
libc	✅	❌
shell	✅	✅
`ssh(1)` / `sshd(8)`	✅	✅
mail server	✅	✅
http server	??	✅
database	❌	✅
python	??	✅

System Software vs. Third Party Software

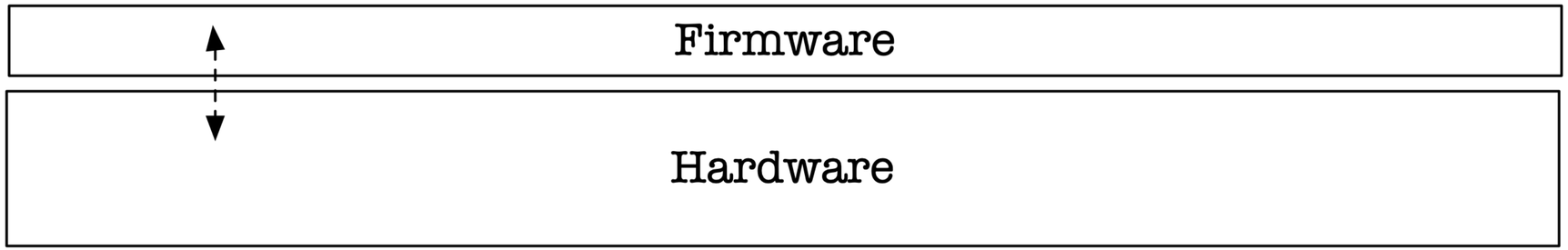
Example	System / OS	3rd Party / Add-on	Packaged?
kernel	✅	❌	
drivers	✅	✅	
firmware	❌	✅	
libc	✅	❌	
shell	✅	✅	
`ssh(1)` / `sshd(8)`	✅	✅	
mail server	✅	✅	
http server	??	✅	
database	❌	✅	
python	??	✅	

System Software vs. Third Party Software

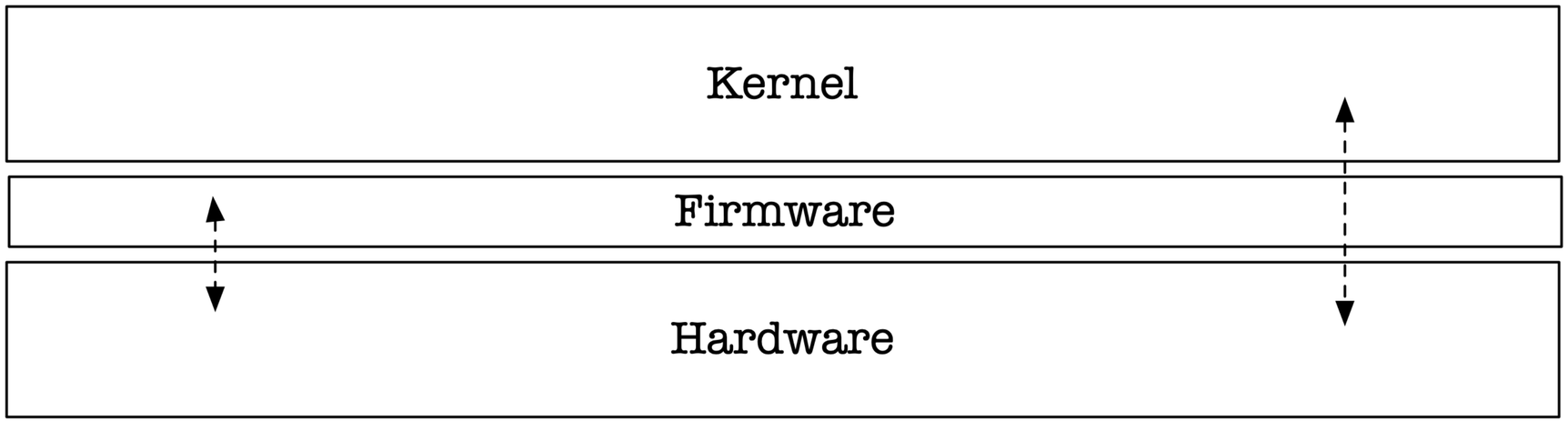
Example	System / OS	3rd Party / Add-on	Packaged?
kernel	✅	❌	??
drivers	✅	✅	??
firmware	❌	✅	??
libc	✅	❌	??
shell	✅	✅	??
`ssh(1)` / `sshd(8)`	✅	✅	??
mail server	✅	✅	??
http server	??	✅	??
database	❌	✅	??
python	??	✅	??

Software Types

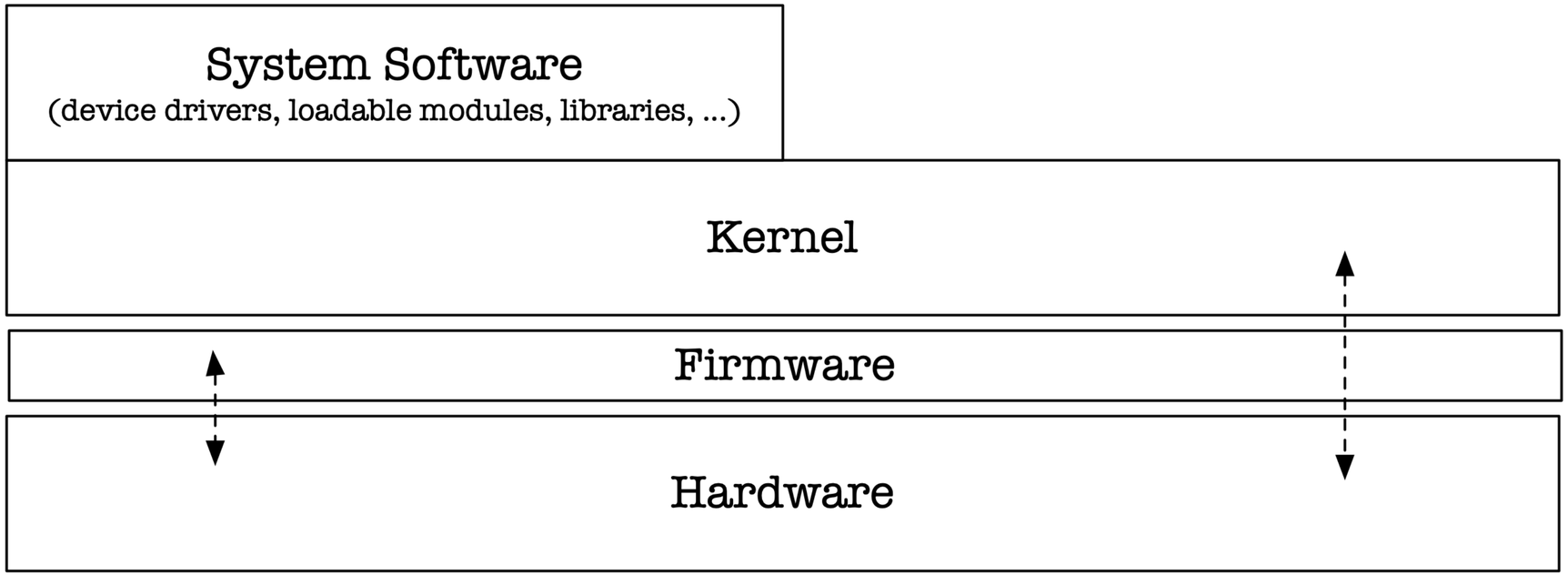
Software Types



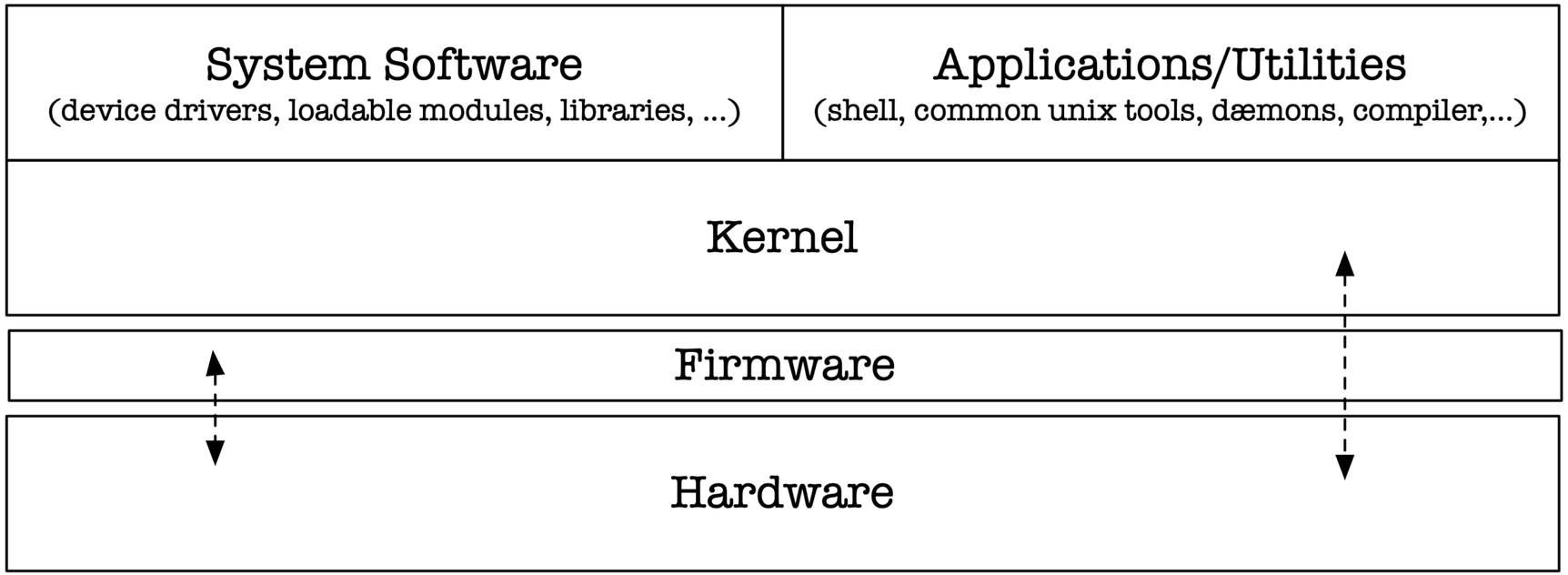
Software Types



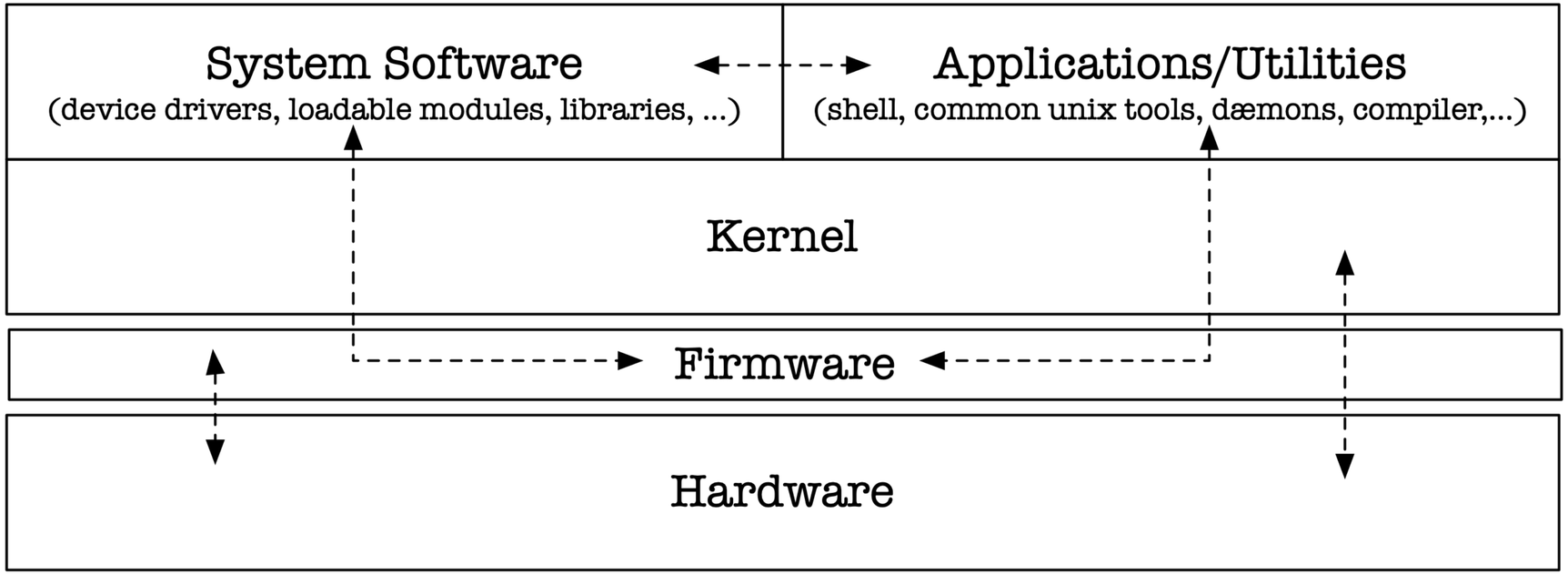
Software Types



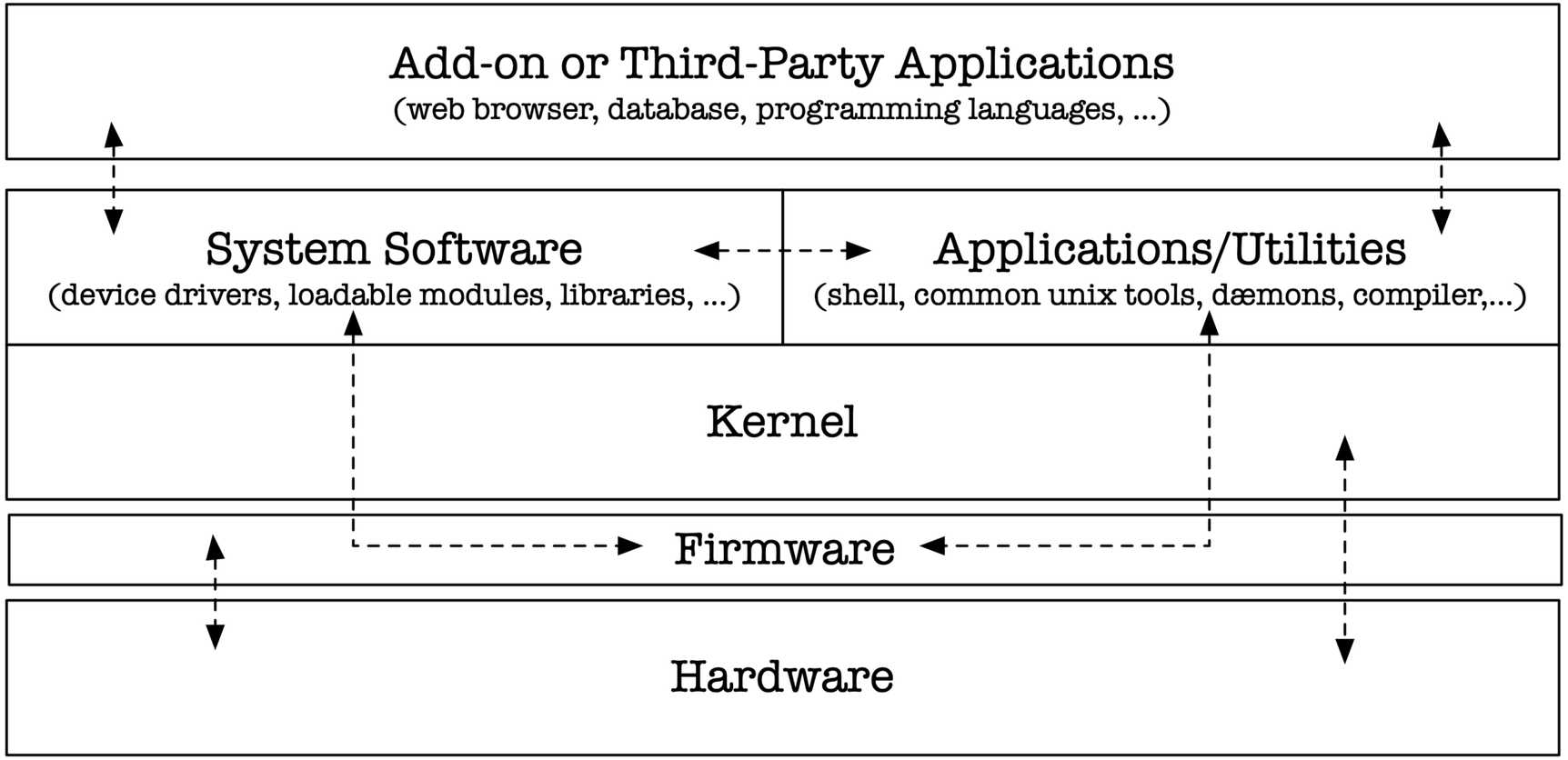
Software Types



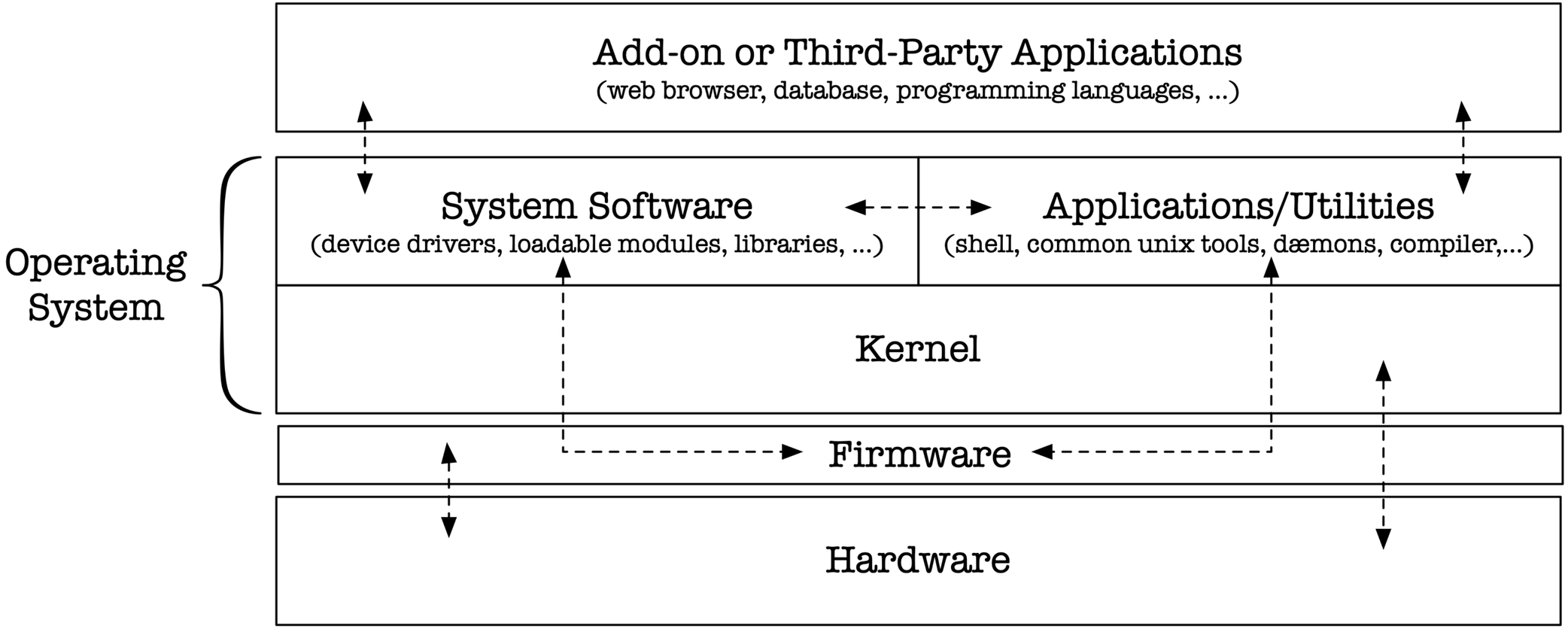
Software Types



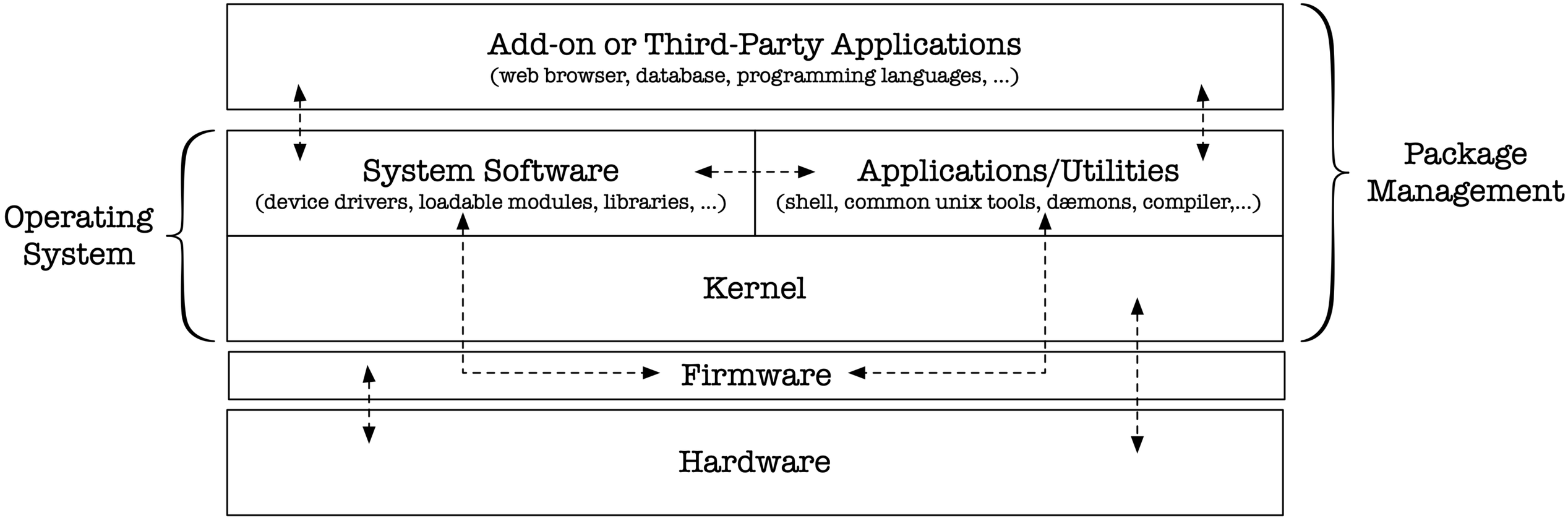
Software Types



Software Types



Software Types



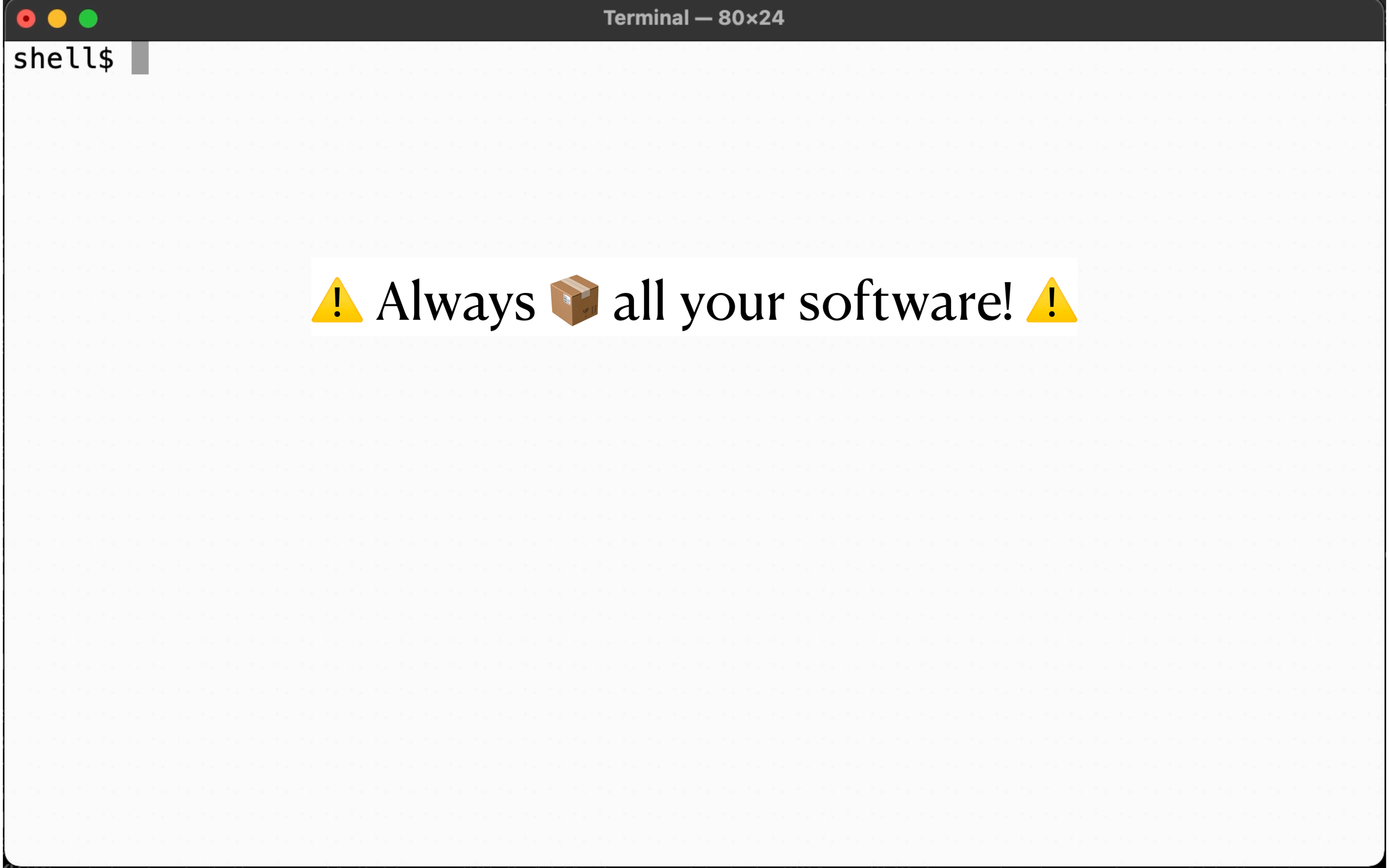
System Software vs. Third Party Software

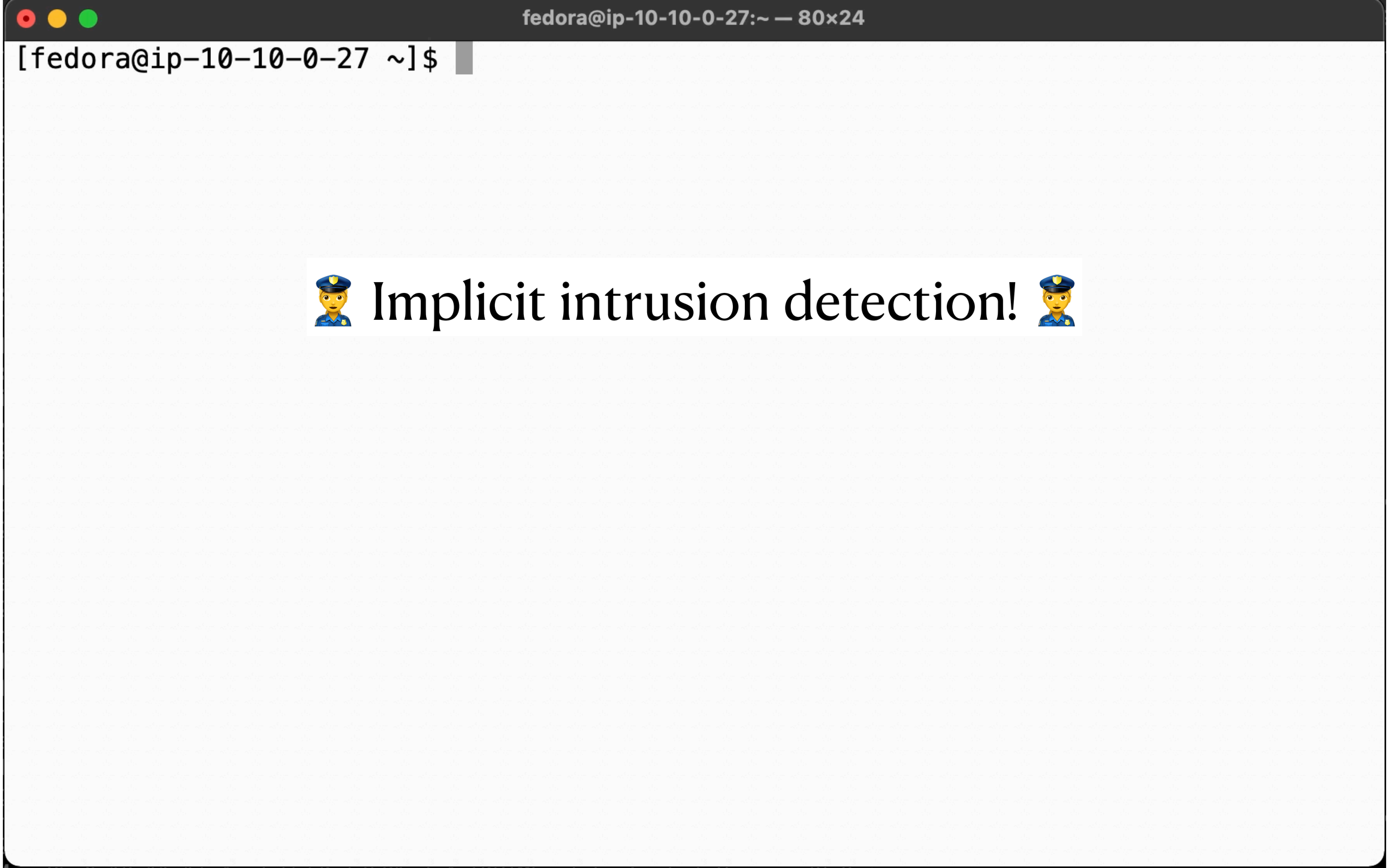
Consider:

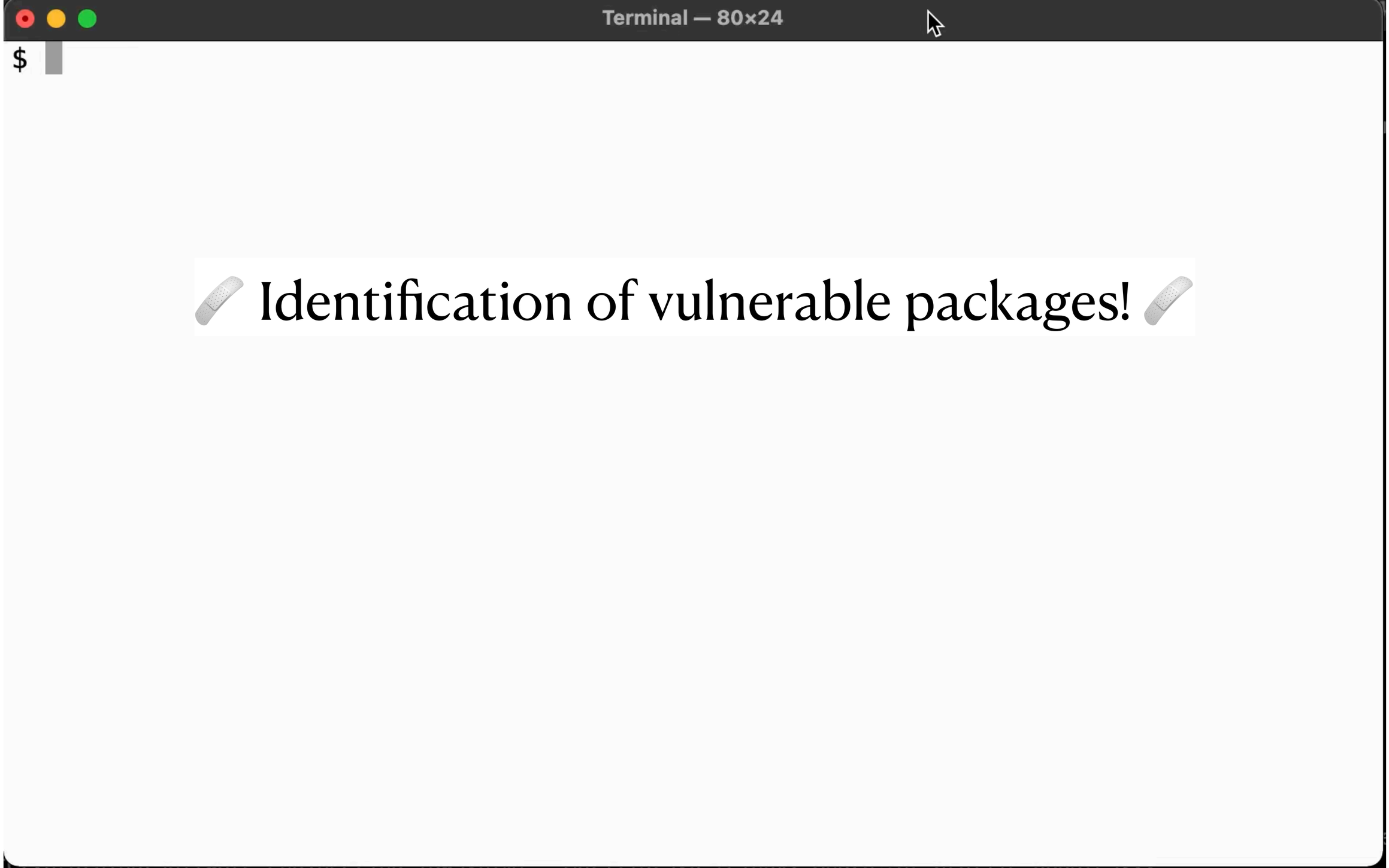
- OS upgrades vs. software upgrades
- location of configuration files
- duplicates or conflicting versions in the base system vs. the add-ons
- startup scripts, dæmons
- location of third party software
- dependencies
- installation by hand and/or installation using a package manager
- proprietary third party software

You don’t get to choose.

You routinely have to build from source *and* (re-)package your software.


Terminal — 80×24
shell$
⚠️ Always 📦 all your software! ⚠️


fedora@ip-10-10-0-27:~ — 80×24
[fedora@ip-10-10-0-27 ~]$
👮 Implicit intrusion detection! 👮


Terminal — 80×24
$
🩹 Identification of vulnerable packages! 🩹

Summary and Exercises

- What comprises an OS, what is “System” vs. Add-on is not an obvious distinction.
  - Some dependencies are more tightly coupled (e.g., kernel + libc).
  - Some required components, alternative options, or completely optional, yet convenient add-ons may be grouped into the OS “distribution”.
- Package Managers Features:
  - easy and scalable installation of software
  - automatic resolution of software dependencies
  - package and file inventory
  - package and file integrity checks
  - vulnerability checks
  - integration with the OS

Summary and Exercises

- Identify a piece of software you use, but that's not packaged for a given package manager. Create a package for it, then contribute upstream.
- Create a cheat sheet for 4 different package managers, listing the most important equivalent commands, such as "install a package", "update a package", "remove a package", "list contents of a package", and "list the package owning the given file".
- How does your preferred OS update firmware?
- How does the concept of *reproducible builds* relate to what we discussed here?
- What is the overlap with *system configuration*? Can a package manager *assert state*?
- Compare binary package management to building and installing an application from source: `https://stevens.netmeister.org/615/package-exercise.html`

Links

Software Installation and Package Management:

https://www.netmeister.org/book/05-software-installation-and-package-management.pdf

pkgsrc (NetBSD and others): https://www.netbsd.org/docs/pkgsrc/

FreeBSD Ports: https://www.freebsd.org/ports/

RPM: https://rpm.org/

Debian Package Management: https://wiki.debian.org/PackageManagement

OmniOS Package Management:

https://github.com/omniosorg/omnios-wiki/blob/master/GeneralAdministration.md#user-content-package-management