CS 447/647

Software Management

Overview

What is required for a network-based installation?

What is a high-level package manager? (Know Debian + RHEL)

What is a low-level package manager? (Know Debian + RHEL)

What is Debian?

How is a .deb package structured?

Installation

- Many different Methods
- Media - Easiest but doesn’t scale
  - USB
  - CDROM
  - DVD
- Network - Scales but requires development
  - PXE, iPXE more likely
  - TFTP
  - DHCP
  - PXE Kernel
  - PXE Menu
- chroot - Specialized
  - Virtual Machines
  - Containers

Installation - Media

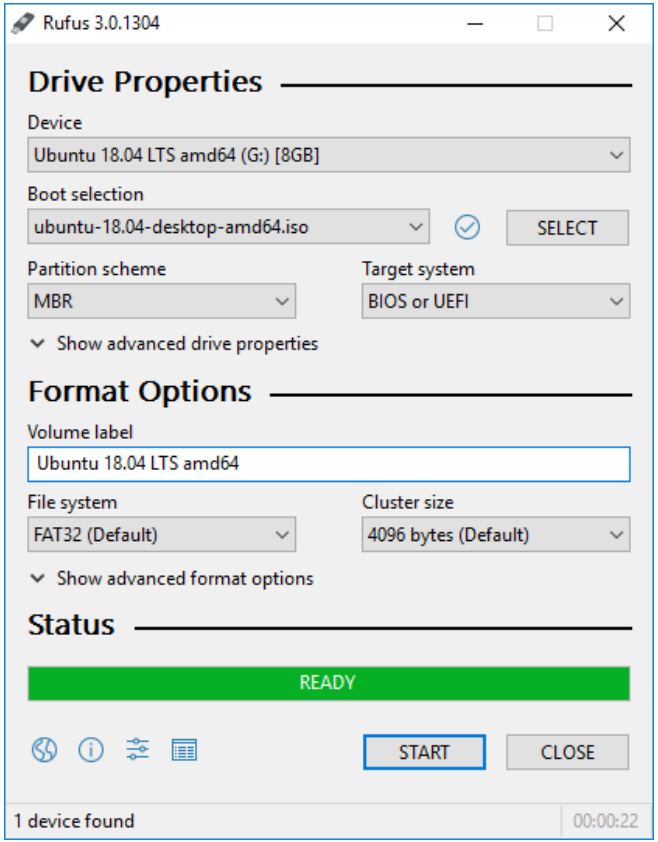
```
wget \
https://mirrors.ocf.berkeley.edu/ubuntu-releases/24.04.2/ubuntu-24.04.2-desktop-amd64.iso

dd if=ubuntu-24.04.2-desktop-amd64.iso of=/dev/sd[a-z] bs=1M


# or use a GUI
UNetbootin, Rufus
```

Installation - Media

Installation - Media

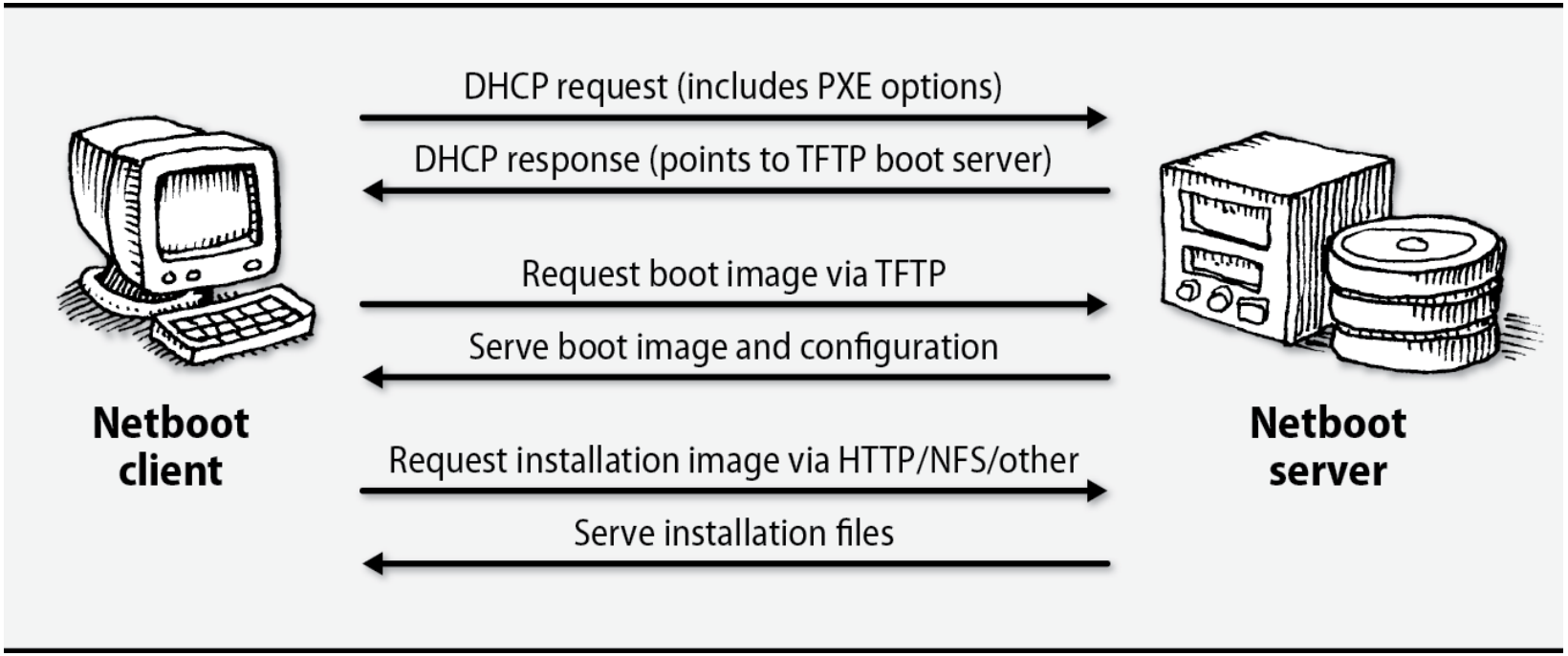


Network

Network

- Much more advanced than a media based installation
- Requires
  - DHCP
  - iPXE - https://ipxe.org/
  - TFTP - Trivial File Transfer Protocol
  - Networking knowledge
  - Scripting and menu creation
  - HTTP (optional but likely)
- Some open-source solutions
  - Cobbler
  - FOG Project (https://fogproject.org/)

Network - PXE


DHCP request (includes PXE options)
DHCP response (points to TFTP boot server)
Request boot image via TFTP
Serve boot image and configuration
Request installation image via HTTP/NFS/other
Serve installation files
Netboot client
Netboot server

Network - DHCP

- Dynamic Host Control Protocol
- RFC2131 - 1997
- Provides a framework for passing configuration information to hosts on a TCP\IP network
- UDP - User Datagram Packet Based
- “DHCP in its current form is quite insecure.”

https://tools.ietf.org/html/rfc2131

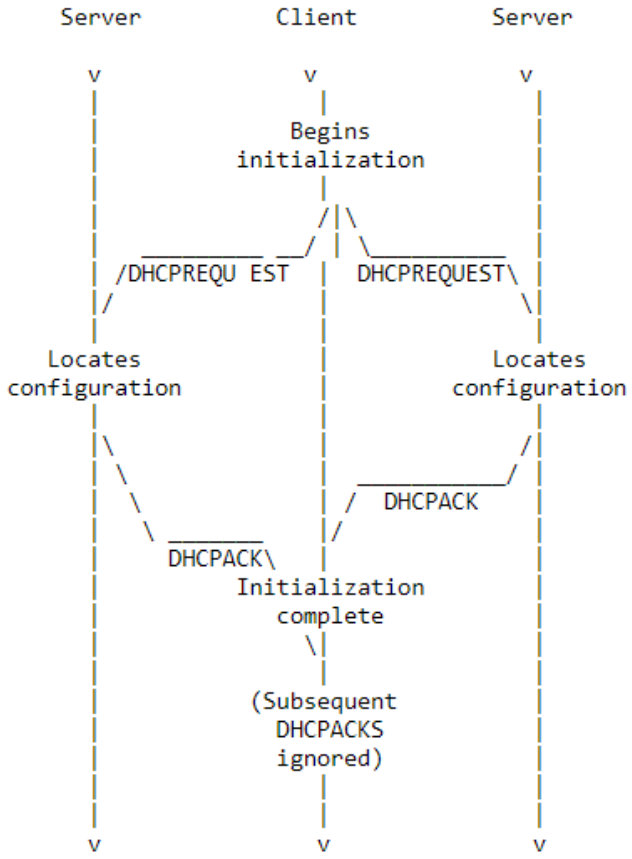


Figure 4: Timeline diagram of messages exchanged between DHCP client and servers when reusing a previously allocated network address

Network - DHCP Extensions

“Configuration parameters and other control information are carried in tagged data items that are stored in the 'options' field of the DHCP message.”

“Security issues are not discussed in this memo.”

RFC2132 - https://tools.ietf.org/html/rfc2132

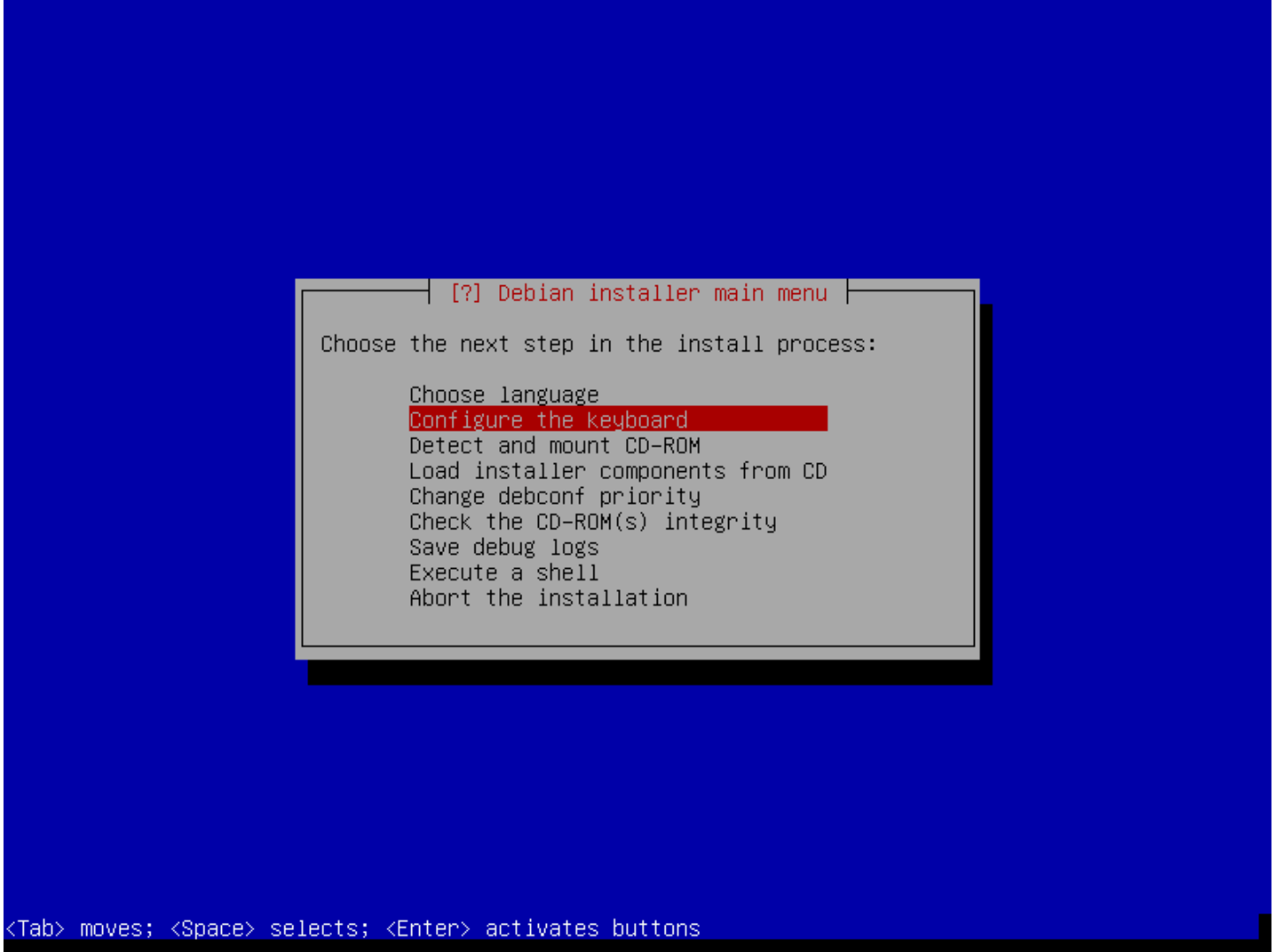
Network - DHCP Extensions

3: Router
5: Name Server
6: DNS Server
9: LPR - Line Printer Remote protocol
12: Hostname
17: Root Path
51: IP Lease Time
66: TFTP server name
67: Bootfile name
61: Client-identifier - DHCP Client
190: Username
191: Password

RFC2939
https://ipxe.org/howto/dhcpd

Network - PXE

- Two popular options
  - PXELINUX in SYSLINUX
  - iPXE (We use this)
    - Scripting
    - Proprietary Devices
    - TLS Authentication
- The PXE kernel is stored in the TFTP server's **tftpboot** directory
- The DHCP Server points to
  - Server IP
  - Filename
- The kernel loads a menu
  - PXELINUX - TFTP File
  - iPXE - TFTP File, HTTP URL, Script


[?] Debian installer main menu
Choose the next step in the install process:
Choose language
Configure the keyboard
Detect and mount CD-ROM
Load installer components from CD
Change debconf priority
Check the CD-ROM(s) integrity
Save debug logs
Execute a shell
Abort the installation
<Tab> moves; <Space> selects; <Enter> activates buttons

Network - PXE

```
# Debian preseed.cfg
# https://www.debian.org/releases/buster/example-preseed.txt
# Locale sets language and country.
d-i debian-installer/locale string en_US



# Keyboard selection.
d-i console-setup/ask_detect boolean false
d-i console-setup/layoutcode string us

### Network configuration
d-i netcfg/choose_interface select eth0
d-i netcfg/dhcp_timeout string 60
```

Package Management

Package Management

- .tar.gz was the original package format
  - Had to compile
  - Messy
- Package management systems
  - Create order from chaos*
  - Easy to install and remove packages
  - Contains
    - Binaries, shared libraries, manpages and configuration files
  - Handle dependencies
  - Run scripts before, during and after installation
    - systemctl

Package Management

- Package versions different than software versions

```
apt search strace | grep ^strace
strace/bionic,now 4.21-1ubuntu1 amd64 [installed]

strace -V
strace --version
UNKNOWN
```

Package Management

- System administrators can create packages
  - rpm for CentOS\Redhat
  - dpkg for Debian\Ubuntu
  - snap - self-contained, no dependencies
    - flatpak for Fedora
- Meta-packages or group installs (Redhat)
  - Package that contains nothing
  - Used to install other packages
  - “ubuntu-desktop”


Activities
Wed 11:35
Type to search...
Books
Boxes
Builder
Calculator
Calendar
Cheese
Clocks
Contacts
Documents
Evolution
Fedy
Files
Firefox
Font Manager
GNOME Recipes
GNU Image Ma...
LibreOffice Calc
LibreOffice Draw
LibreOffice Impr...
LibreOffice Writ...
Maps
MuseScore 2
Photos
pick
Frequent
All

rpm - Redhat\CentOS

- RPM Package Manager (rpm)
- Useful arguments
  - `-i (install), -U (upgrade), -e (erase), and -q (query)`
- Query all packages
  - `rpm -qa`
  - `rpm -qa | grep strace`
- List files in a package
  - `rpm -ql strace`
- Not often used alone
  - Dependency hell
    - `rpm -q --whatrequires strace`

dpkg - Debian

- Package manager for Debian
- Useful arguments
  - `-i (install), -r (remove)`
- Query all packages
  - dpkg -l
- List files in a package
  - `dpkg -L strace`
- Reconfigure - `dpkg-reconfigure`
  - `apt install slapd`
  - `dpkg-reconfigure slapd`

High-Level Package Managers

High-Level Package Managers

- Meta Packaging system
  - apt - Debian/Ubuntu
  - dnf - Fedora/Redhat/CentOS (formerly used yum)
- Simplifies
  - Finding packages
  - Automates installation,updating and OS upgrading
  - Handle inter-package dependencies
- Requires a significant amount of infrastructure
  - Maintainers
  - Web-servers
  - Certificates
- Distributed
  - Anyone can host a repository

High-Level Package Managers

- Red Hat Network
  - Commercial
  - Offers site-wide system management
  - Works with the proprietary Satellite Server
  - Supports apt and dnf
- APT*
  - Better documentation
  - Vibrant community
  - rpm support (alien)
  - dpkg + .deb
- dnf
  - RHEL\CentOS
  - RPMs
  - “Enterprise”

*preferred

Package repositories

- HTTP or FTP based
  - https://mirrors.ocf.berkeley.edu/ubuntu
- Supports a release
  - 18.04 bionic or 9.1 stretch
  - Package versions are frozen
    - 4.15 Kernel
  - Security updates - Backported
- Component
  - Subset of software
  - main, free, non-free, contrib
    - Debian Free Software Guidelines (DFSG)
      - https://www.debian.org/social_contract.en.html
  - GPL vs non-GPL
- Architecture
  - i686, amd64, and arm64

Package repositories

- Individual packages
  - Example: python3 (python3.6)
  - Make up components
  - Architecture-specific
  - Versioned independently
- “extras”
  - Multiverse and universe
    - Non-free components
  - Manually added
  - Examples
    - Oracle Java, and NVIDIA

/etc/apt/sources.list

```
#Basic

deb http://httpredir.debian.org/debian buster main contrib
non-free

#Backports

deb http://httpredir.debian.org/debian buster-backports main
contrib non-free

apt search wireguard
```

Backports

- Backports are packages taken from the next Debian release
- Recompiled for usage on Debian stable
- Easily upgraded
- Recommended to only select single backported packages that fit your needs
- https://backports.debian.org/

Advanced Packaging Tool (APT)

- Mature
  - First released in 1999 as part of Debian 2.1 (Slink)
- Supports updates and upgrades
  - Moving from one release to another.
  - 18.04 -> 20.04
  - yum does not support release upgrades, but dnf does
- Distros contain many metapackages (yum calls them groupinstalls)
  - linux-image-generic
  - xfce-desktop
  - build-essential
- Includes low level utilities
  - apt-file
  - apt-cache

Commands

```
apt update          # apt downloads lists
                    # from repos defined in
/etc/apt/sources.list
apt search neovim
# Look for NeoVIM
apt show neovim
# Show neovim information
apt-cache madison neovim                                  # Show
Versions
apt-cache madison gcc
# Show GCC versions
apt -s install  gcc=4:7.3.0-3ubuntu2           # Downgrade
```

Debian Packaging

https://www.debian.org/doc/manuals/packaging-tutorial/packaging-tutorial.en.pdf

What is Debian?

- **GNU/Linux distribution**
  - coreutils: https://www.maizure.org/projects/decoded-gnu-coreutils/
  - gcc
- 1st major distro developed “openly in the spirit of GNU”
- **Non-commercial**, built collaboratively by over 1,000 volunteers
- 3 main features:
  - **Quality** – culture of technical excellence
    We release when it’s ready
  - **Freedom** – devs and users bound by the Social Contract
    Promoting the culture of Free Software since 1993
  - **Independence** – no (single) company babysitting Debian
    And open decision-making process (do-ocracy + democracy)
- **Amateur** in the best sense: done for the love of it

Debian Packages

- .deb files (binary packages)
  - A very powerful and convenient way to distribute software to users
  - One of the two most common package formats (with RPM)
- Universal:
  - 30,000 binary packages in Debian
    - most of the available free software is packaged in Debian!
  - For 12 ports (architectures), including 2 non-Linux (Hurd; KFreeBSD)
  - Also used by 120 Debian derivative distributions

.deb Packaging Format

- .deb file: an ar archive

```
$ ar tv wget_ 1 .12 -2.1 _i386 . deb
rw-r--r--  0/0        4 Sep  5 15:43  2010 debian-binary
rw-r--r--  0/0     2403 Sep  5 15:43  2010 control.tar.gz
rw-r--r--  0/0   751613 Sep  5 15:43  2010 data.tar.gz
```

  - debian-binary: version of the deb file format, "2.0\n"
  - control.tar.gz: metadata about the package
    control, md5sums, (pre|post)(rm|inst), triggers, shlibs, . . .
  - data.tar.gz: data files of the package

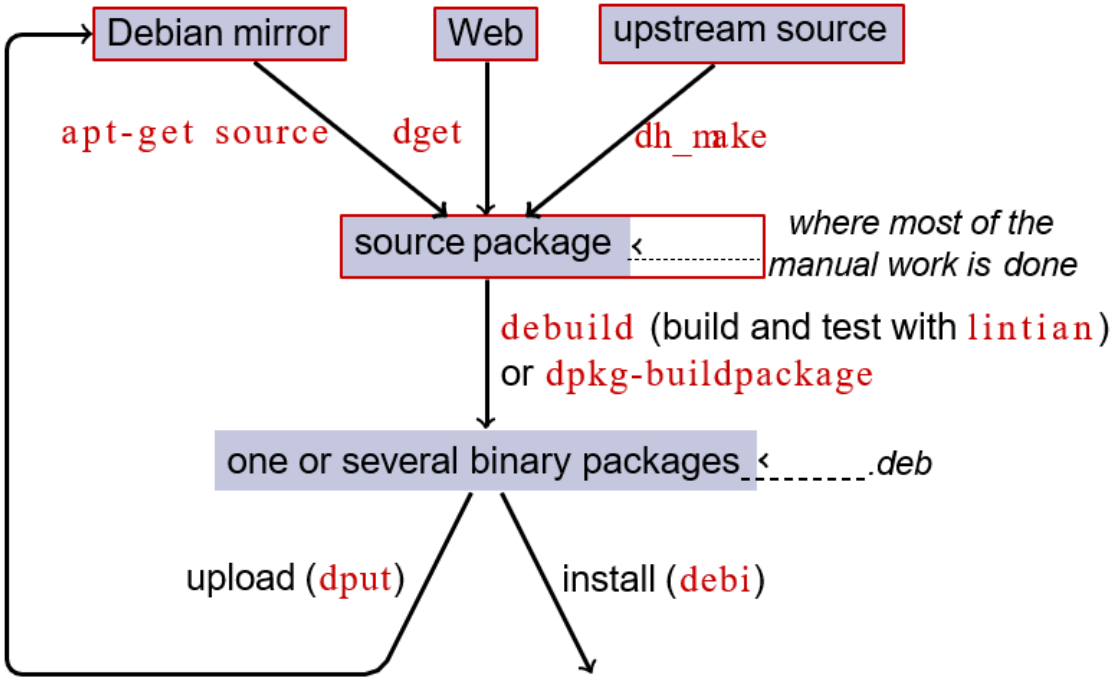
.deb Packaging Format

- You could create your .deb files manually http://tldp.org/HOWTO/html_single/Debian-Binary-Package-Building-HOWTO/
- But most people don’t do it that way

Tools

- A Debian (or Ubuntu) system (with root access)
- Some packages:
  - **build-essential**: has dependencies on the packages that will be assumed to be available on the developer's machine (no need to specify them in the Build-Depends: control field of your package)
    - includes a dependency on dpkg-dev, which contains basic Debian-specific tools to create packages
  - **devscripts**: contains many useful scripts for Debian maintainers
- Many other tools are available, such as debhelper, cdbs, quilt, pbuilder, sbuild, lintian, svn-buildpackage, git-buildpackage, . . .
- Install them when you need them.

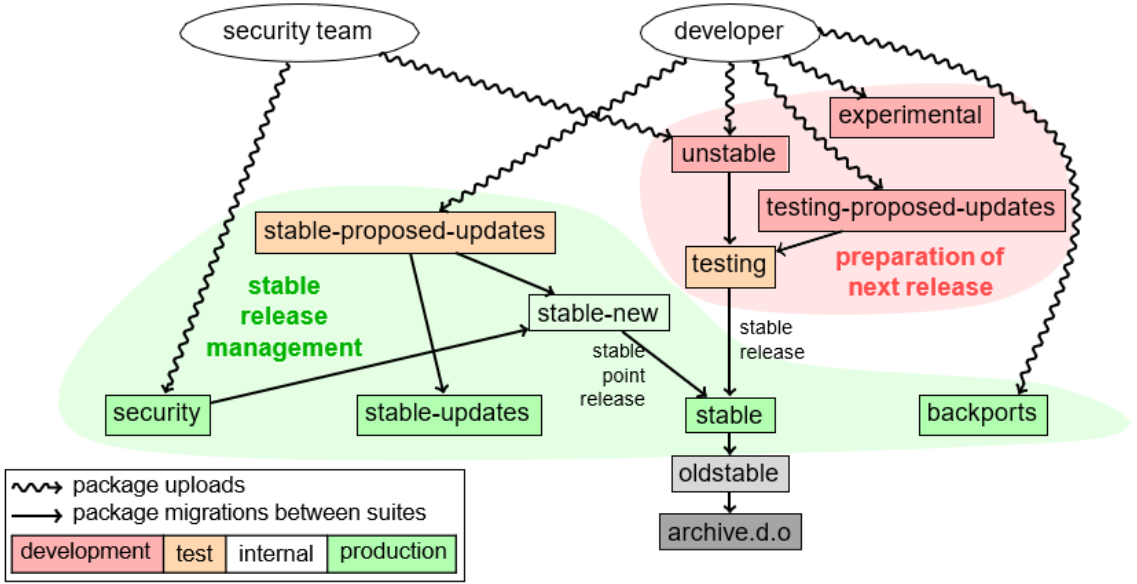
Workflow


Debian mirror
Web
upstream source
apt-get source
dget
dh_make
source package
where most of the manual work is done
debuild (build and test with lintian)
or dpkg-buildpackage
one or several binary packages
.deb
upload (dput)
install (debi)

Rebuilding dash

1. Install packages needed to build dash, and devscripts
   `sudo apt-get build-dep dash`
   (requires deb-src lines in `/etc/apt/sources.list`)
   `sudo apt-get install --no-install-recommends devscripts  fakeroot`
2. Create a working directory, and get in it:
   `mkdir /tmp/debian-tutorial ; cd /tmp/debian-tutorial`
3. Grab the dash source package
   `apt-get source dash`
   (This needs you to have deb-src lines in your `/etc/apt/sources.list`)
4. Build the package
   `cd dash-*`
   `debuild -us -uc` (-us -uc disables signing the package with GPG)
5. Check that it worked
   There are some new `.deb` files in the parent directory
6. Look at the `debian/` directory
   That’s where the packaging work is done

Workflow



Based on graph by Antoine Beaupré. https://salsa.debian.org/debian/package-cycle

Testing

- The package has been in "unstable" at least for 2-10 days (depending on the urgency of the upload).
- The package has been built for all the architectures which the present version in testing was built for.
- Installing the package into testing will not make the distribution more uninstallable.
- The package does not introduce new release critical bugs