Installing Software

Really only two ways

Build (compile) from source

Copy binaries

Build

Pros

- One set of files can support many "targets"
- Relatively small packages

Cons

- Requires a compiler on the target
- Can be quite an involved process
- Users get access to the source code
- Publisher has less control

MOHAWK COLLEGE

Copy Binaries

<u>Pros</u>

- Publisher has a lot of control
- Easier for user

<u>Cons</u>

- More targets to maintain
- More work for the publisher to package

mohawk
COLLEGE

Build Process

- Acquire the source package (download, tape, CD)

- Unpack the files (zip, tar)

- Configure the build (configure script)

- Build (make)

- Install to final location (make install)

Copy Process

- Copy the correct binaries to the correct spot on the system*

There are many ways to do this.

Copy Processes

Two basic strategies:

1) Simple copy

2) Provide some sort of install script

Copy or Build ???

Build is popular for FOSS

Build is popular with hackers

Copy is popular with administrators

Copy is popular with commercial software vendors

Basic Linux Software Management Structure

- Packages (RPM, Debian Packages)

- Package Manager (YUM, APT)

  - YUM manages RPM packages

  - APT manages Debian Packages

Commercial Unix systems have their own package management systems

III mohawk
COLLEGE

RedHat Package Management

RPM

- Each component is packaged into a single RPM file

- File extension `.rpm` `(zip-3.0-1.el6.i686.rpm)`

- Contains:

  - Files

  - Scripts

  - Dependencies

- Maintains a local database of what has been installed

RPM and Dependencies

- Each package can require zero or more packages

- Package will not install if dependencies are not present

- This can be very frustrating to manage directly.

   (we will get a glimpse at this frustration in the lab)

What can be done about this dependency problem?

Package Manager

- Manages dependencies

- RedHat Enterprise Linux uses Yellow Updater Modified (YUM)

YUM

- Groups packages into Repositories

- Automatically resolves dependencies

- Each distribution maintains repositories of packages <u>that have all been tested together.</u>

- Repositories can be local or accessed over the 'Net

YUM and RHEL

- To access RHEL update repositories a system must be licensed by Red Hat and must be registered with Red Hat.

- Yum still works from local repositories, like we use in the labs.

- RedHat offers a free license tier under the Red Hat Developer program

Should I use YUM, RPM, or Build?

A general rule of thumb…

Use the packages provided by your distribution if at all possible, they have been tested with each other.

Only resort to fetching individual RPMs or building from source when the software you need is not part of you distribution.

Demonstration

Let's walk through a few software installs like we will in the lab…