Installation Support for Large Scale Deployments

David Cantrell <dcantrell@redhat.com>
Software Engineer, Red Hat
Introduction

- Anaconda Information
- Fedora and RHEL Information
- Large Scale Deployment Topics
- Tools Available
- Kickstart Overview
- Advanced Installation Topics
- Trigger Scripts
- Modifications to the Installer
- Development
Presenter

David Cantrell

- Software Engineer at Red Hat
- Projects include anaconda, dhcp, and dhcpv6
- Primary focus is network software
- Currently involved with RHEL 4, RHEL 5, and Fedora
- Email is dcantrell@redhat.com
- http://dcantrel.fedorapeople.org/
What is Anaconda?

- Operating system installer
- Interfaces provided in GUI and text modes
- Unattended installs provided by Kickstart
- System setup for storage, network, services, and accounts
- Used by RHEL and Fedora (and others)
- Forms first impression of the product
- “First tool to run, first tool to uninstall”
How is Fedora related to RHEL?
Large Scale Deployment Problems

- Reproducible installations
- Ability to deploy to new hardware quickly
- Complex storage setup
- Hundreds of network interfaces
- Site-specific tuning and final configuration
- Minimal installation complexity
Solutions That Do Not Scale

• Imaging
  • With the exception of LiveCD or LiveUSB
• Interactive installs
Tools Available

- Anaconda and Kickstart
  - http://fedoraproject.org/wiki/Anaconda/Kickstart
-  ksvalidator
-  system-config-kickstart
-  yum and createrepo
  - yum plugins for added functionality
  - createrepo to generate site-specific package repos
- Cobbler (http://cobbler.et.redhat.com/)
  - Experimental provisioning software
Kickstart Basics

- Need a valid install command
- Interactive installs create `/root/anaconda-ks.cfg`
- Any number of `%pre` and `%post` scripts
- Components listed in the `%packages` section
- Defaults for many settings if not specified
- Required settings not specified will cause user prompting
Kickstart Advantages

- Easily reproducible installs
- Eliminates the need for interactive setup
- Can run unattended
- Not architecture dependent
- Can be defined per system or per task
Kickstart Disadvantages

- Kickstart file management
  - SOLUTION: version control
- Initial file creation and testing
  - SOLUTION: unpaid intern
- New RHEL or Fedora releases can deprecate or remove Kickstart commands
  - SOLUTION: use ksvalidator or check install.log for deprecation warnings, update your ks files
- Difficult to see what's going on
  - SOLUTION: vnc
Boot Arguments

- Kickstart specification
  - ks= argument
- Network configuration
  - ip= argument
- Boot CDs or DVDs
- Network boot
  - Automatic search using just the 'ks' option
- Driver disks
Disk Provisioning

- **part** command creates partitions (normal, raid, or pv)
- **raid** command creates software RAID volumes
- **volgroup** command binds pv partitions in to a vg
- **logvol** command creates LVM volumes
- All partitions are formatted unless you specify –noformat
- Alternatives:
  - Use a %pre script and invoke parted or sfdisk manually to partition the disk.
Disk Provisioning (cont.)

- part
- normal
- md
- pv
- volgroup
- logvol
- raid
Network Configuration

- During installation, anaconda will bring up all network devices listed in the ks file
- Prevent device configuration during installation by using the `onboot=no` switch
- Use `ksdevice=` to specify the network device to use during installation
  - `ksdevice=BOOTIF` means the boot interface
  - `ksdevice=ethX` to use a specific interface
  - `ksdevice=link` to use first interface found with active link
  - `ksdevice=XX:XX:XX:XX:XX:XX`
Network Configuration (cont.)

- Use `network` command to configure the interface, set onboot to no, use a `%post` script to flip ONBOOT back to yes at end of install.

```
%post --log=/root/finish.log
SCRIPTS=/etc/sysconfig/network-scripts
if [ -d ${SCRIPTS} ]; then
cd ${SCRIPTS}
for cfg in ifcfg-* ; do
    sed -i -e 's|ONBOOT=no|ONBOOT=yes|g' ${cfg}
done
else
    echo "${SCRIPTS} not found"
fi
```
Other System Configuration

- Time zone
- Root password (now with SHA-256 and SHA-384 encryption options...only in Kickstart)
- User accounts
- Enabled and disabled services
- Keyboard layout and language settings
- SELinux
Package Selection

- Add additional repositories with the `repo` command
- Packages and groups can be listed in the `%packages` section
- Groups specified with `@groupname`
- Prevent a package from installing with `-packagename`
- Wildcards supported: `*-devel`
- Install all packages available: `*`
- Only install core set of packages: `-*`
  - Don't do this
Package Selection (cont.)

- Added repositories need to be local collections or URLs
  - Repositories must be yum repos
  - Use `createrepo` to generate repodata
- Multiple `%packages` sections are combined into one for dependency resolution
- Anaconda uses yum to do dependency resolution
  - More packages == more resolution time
  - Dependency resolution overrides minus sign
Package Selection (cont.)

• Packages are organized into groups
• Groups have three categories for packages:
  • Mandatory
  • Default
  • Optional
• Use `yum groupolist` and `yum groupinfo` to see groups
Package Selection (cont.)

- Options for `%packages`:
  - `--default`
    - Install Mandatory and Default packages
- Options to `@groupname`:
  - `--nodefaults`
    - Install only Mandatory packages
  - `--optional`
    - Install Mandatory, Default, and Optional packages
Trigger Scripts

• Kickstart offers pre-installation and post-installation scripts
• You can have as many as you want
• Default interpreter is bash, but you can change that with the --interpreter option
• Trigger scripts can log output with the --log option
Trigger Scripts - \texttt{%pre}

- \texttt{%pre} runs before installation begins
- They are executed in the installer's environment
- Uses:
  - Examine or configure hardware before anaconda runs
  - Manual disk partitioning
  - Advanced disk configuration
  - Advanced network configuration
- Multiple \texttt{%pre} scripts can be specified and are executed in the order they appear in the file.
Trigger Scripts - %post

- Also called finishing scripts
- `%post` scripts run after installation has completed
- They use the newly installed system as a chroot environment (compared to where `%pre` runs)
- Pass `--nochroot` to `%post` if you do not want it to run in the newly installed environment
Trigger Scripts – Interactive %post

- Prompt for user input with the `zenity` command
- Only available for GUI installs, not text mode (use VNC!)
- Be careful when prompting users and perform input validation

```
%post
zenity -question --text "Install corporate VPN client?"
if [ $? -eq 0 ]; then
    # install vpn client
fi
```
Supplementing the Installer Environment

- **updates.img** can provide anaconda updates, or tools you want to use for installation
- Contents of updates.img gets copied to `/tmp/updates` in the installer environment
- Use it to supply additional tools you need to complete your installation
- These tools will be available to you during `%pre` and `%post` operations
- **NOTE**: For `%post`, be sure to use `--nochroot`, otherwise you won't be able to access `/tmp/updates`
Supplementing the Install Environment (cont.)

- Making an updates.img file:
  ```
  dd if=/dev/zero of=updates.img bs=1k count=4096
  mke2fs updates.img
  mkdir u
  mount -o loop updates.img u
  ```

- Use updates= boot parameter to supply updates at boot time

- If making a custom CD, put updates.img in the images/ subdirectory and anaconda will automatically find it
Customizing the Installer Environment

- Custom welcome graphic, create a new `splash.png` file and place it in your `updates.img`
- Custom header graphic, create a new `anaconda_header.png` and place it in your `updates.img`
- Be sure to use the ones found in `/usr/share/anaconda/pixmaps` to make sure you get the sizes right
Development

- http://fedoraproject.org/wiki/Anaconda
- anaconda-devel-list@redhat.com
- #anaconda on irc.freenode.net
- Report bugs at http://bugzilla.redhat.com/
Questions