CentOS Stream on Desktop

or: How I Learned to Stop Worrying and Love LTS

CentOS Dojo, May 2021

Michel Salim
Production Engineer

FACEBOOK Infrastructure

Agenda

Desktop fleet primer

Desktop Linux @ Facebook

Provisioning & Management

Upstream collaboration

Desktop Fleet Primer

Desktop fleet primer

What do we manage?

- Employee devices
- Mostly laptops, some desktops
- Mostly macOS and Windows, some Linux

Desktop fleet primer

How does it work?

- Client Platform Engineering manages the bare metal experience of the fleet
- OS as a platform
- Individual teams can layer their own configuration
- Built on an Open Source foundation
- Chef, rpm/yum/dnf, chocolatey

Desktop fleet primer

How does it work?

- Community sets the direction
- We move fast; open source often moves faster
- Reuse, don’t reimplement
- Sharing our code means sharing the maintenance and having others extend it

Desktop Linux @ Facebook

Desktop Linux @ Facebook

History

- Started as a grassroot efforts
- Configuration management implemented as best effort
- Pivot to first-class Fedora support
- Branching out to CentOS Stream

Desktop Linux @ Facebook

Why Fedora?

- Align with the server fleet (CentOS Dojo 2020)
- Reuse existing tooling and infrastructure
- “Four Foundations”: Freedom, Friends, Features, and First
- Two-way conduit for collaboration

Desktop Linux @ Facebook

Why /not/ Fedora?

- Fast-moving with short support period (2x6 months + 1 month)
- Fix forward vs backporting security updates
- Third-party vendor support

Desktop Linux @ Facebook

Why CentOS Stream?

- Stable rolling release
- Binary compatibility
- Security updates
- Close relationship with Fedora, including EPEL
- Previous talks:
  - Flock 2019: Facebook ❤️ Fedora (and Chef)
  - Nest 2020: Fedora at Facebook: A pipeline to upstreams
  - FOSDEM 2021: Desktop Linux Management at Facebook

Desktop Linux @ Facebook

Why CentOS Stream?

- Before: CentOS is downstream from RHEL
- Now: CentOS Stream is the upstream for RHEL
- More opportunity for community input
  - E.g. Hyperscale SIG

Desktop Linux @ Facebook

LTS

- Release cycle comparable to LTS releases from you-know-who
- Extra stabilization process: while C9S is based on F34, it will be released months later
- This means we don’t have same incentive to be conservative with F34 features
  - Rather the opposite, we have significant features such as systemd-oomd and Pipewire in F34

Desktop Experience

CentOS on Desktop

Initial experience

- Works surprisingly well on Lenovo ThinkPad T-series
  - Even on the AMD T495s except for a harmless kernel warning
- Not so well on X1 Yoga
- GNOME 3.28 is new enough for most extensions we care about
- PSI actually enabled (need psi=1 passed at boot)
- Suspend works
- Internal webcam and Bluetooth don’t
- Hoping to address some of these as part of the CentOS Stream 9 release process

CentOS on Desktop

Initial experience

- Package availability is poor
- Flathub helps
- EPEL too, but there are frictions - will discuss later

Provisioning & Management

Provisioning & Management

Kickstart

- We traditionally do PXE network installations
- Transition to self-service imaging
- Modular kickstart builder
  - facebookincubator/linux-desktop-kickstarts
  - Lorax PR 1047 to fix mkksiso EFI boot
- Chef bootstrap
  - GitHub: facebookincubator/go2chef

Provisioning & Management

Packaging

- Standard c8s repos
- Vetted partial mirror of internal repo
- EPEL
- Hyperscale SIG
- RPM Fusion, maybe?
- Flatpak

Provisioning & Management

Packaging

- Internal packages are mostly built against the Facebook runtime
  - This includes e.g. the Python stack
- These work without issue on both Fedora and CentOS Stream
- Previous attempts to build natively for Fedora does not age well with Python 2 EOL
  - It’s time consuming to target 2 (sometimes 3) concurrent Fedora releases
- We now recommend either building upstream in Fedora / EPEL / Hyperscale or building against the runtime

Provisioning & Management

Major OS upgrades

- Fedora
  - beta testing
  - dnf system-upgrade
  - GNOME Software
- CentOS
  - Reimage
- Push notifications
- Btrfs motivation: snapshots

Provisioning & Management

Minor upgrades

- dnf-automatic
- sharding for internal packages

Provisioning & Management

Chef

- Chef for config management
- Philosophy: https://tinyurl.com/mgxb923
  - Layered configuration through attribute-based APIs
  - Separation of policy and mechanism
  - Idempotency
  - Configuration as programming
- Cookbooks in source control
- Develop locally, test on real machines

Provisioning & Management

Chef

- Documentation, best practices and tooling
  - GitHub: facebook/chef-utils
  - GitHub: facebookincubator/go2chef
  - GitHub: facebook/taste-tester
- Cookbooks
  - GitHub: facebook/IT-CPE
  - GitHub: facebook/chef-cookbooks

Provisioning & Management

Chef example: screensaver

```
node['cpe_dconf']['settings']['screensaver'] = {
  'org/gnome/desktop/screensaver' => {
    'lock-enabled' => 'true',
    'lock-delay' => {
      'value' => 'uint32 0',
      'lock' => false,
    },
  },
  'org/gnome/desktop/session' => {
    'idle-delay' => 'uint32 600'
  }
}
```

Provisioning & Management

Chef example: flatpak

```
node.default['cpe_flatpak']['ignore_failure'] = true
node.default['cpe_flatpak']['manage'] = true
pkgs = %w{
  com.spotify.Client
  com.obsproject.Studio
}

pkgs.sort.each do |pkg|
  node.default['cpe_flatpak']['pkgs'][pkg] = 'flathub'
end
```

Provisioning & Management

Current status

- Pilot
- Community supported, maybe full support with c9s
- Hyperscale SIG

Upstream collaboration

Upstream collaboration

- Fedora Changes
  - BtrfsByDefault (F33)
  - BtrfsTransparentCompression (WIP, F34)
  - EnableSystemdOomd (WIP, F34)
- Internal group of packagers
  - We package FB open source projects
  - We try and get software dependencies into EPEL or Hyperscale
- EPEL Packagers SIG

Upstream collaboration

EPEL Packagers

- Problem statement
  - EPEL is opt-in, many Fedora packagers don’t care about RHEL/CentOS
    - Hoping this might change over time as CentOS Stream takes off
  - Friction in getting packages we care about (and their deps) branched off
  - We’re addressing this both on the tooling and policy side
- Tooling
  - Collaborators can access only allowlisted branches
  - It’s now possible for collaborators to request new branches
  - It will soon be possible for them to push updates
- Policy
  - Right now, if the primary maintainer does not respond to a branch request for EPEL, the only recourse is triggering the non-responsive maintainer process
    - Both time consuming and might appear hostile
  - Once the technical work is done, we want to have a fast-track process for granting access to `epel*` only to the SIG if the maintainer does not respond

Upstream collaboration

Hyperscale SIG

- Problem statement
  - Hyperscalers deploy CentOS at scale
  - Companies individually reinventing the same workload (e.g. backporting more recent packages from Fedora)
  - Pooling resources make sense
- While targeted at servers this actually help desktop usecases too
  - Now that systemd 248 is available we can use oomd
  - Being able to ship an alternate kernel and installation images means we can use Btrfs
    - Transactional upgrades
  - Being able to use newer RPM features faster
    - e.g. zstd support was in Facebook's internal FTL (fast thin layer) backport before it made it to CentOS
    - similar future updates will be in Hyperscale

Upstream collaboration

ELN

- Enterprise Linux Next
- Continuous rebuild of Rawhide with EL rather than Fedora macros
- Currently only packages expected to be in EL are built
- Being able to add additional packages will make this a nice staging ground for EPEL work
  - Separate workload based on ELN
  - Explicitly add leaf packages we want to be available
  - Content Resolver will also pull in dependencies
    - We can make sure these build and branched when the next EPEL is cut

Upstream collaboration

Wishlist

- More recent LTS kernel
  - Being hashed out in Hyperscale, how to get this is signed is an issue
- Major upgrades
- More focus on validating desktop use cases
  - Playing this by ear now, if there’s significant interest a desktop/workstation SIG might make sense

Thank you!

Questions?

FACEBOOK Infrastructure