



Pid Eins

レナート
لينارت

[Google+](#)

[systemd](#)

[PulseAudio](#)

[Avahi](#)

[Repositories](#)

[Imprint](#)

POSTED ON MI 20 APRIL 2011

systemd for Administrators, Part VIII

Another episode of [my ongoing series on systemd for Administrators](#):

The New Configuration Files

One of the formidable new features of `systemd` is that it comes with a complete set of modular early-boot services that are written in simple, fast, parallelizable and robust C, replacing the shell "novels" the various distributions featured before.

Our little *Project Zero Shell*^[1] has been a full success. We currently cover pretty much everything most desktop and embedded distributions should need, plus a big part of the server needs:

- Checking and mounting of all file systems
- Updating and enabling quota on all file systems
- Setting the host name
- Configuring the loopback network device
- Loading the SELinux policy and relabelling `/run` and `/dev` as necessary on boot
- Registering additional binary formats in the kernel, such as Java, Mono and WINE binaries
- Setting the system locale
- Setting up the console font and keyboard map
- Creating, removing and cleaning up of temporary and volatile files and directories
- Applying mount options from `/etc/fstab` to pre-mounted API VFS
- Applying `sysctl` kernel settings
- Collecting and replaying readahead information
- Updating `utmp` boot and shutdown records
- Loading and saving the random seed
- Statically loading specific kernel modules
- Setting up encrypted hard disks and partitions

- Spawning automatic gettys on serial kernel consoles
- Maintenance of Plymouth
- Machine ID maintenance
- Setting of the UTC distance for the system clock

On a standard Fedora 15 install, only a few legacy and storage services still require shell scripts during early boot. If you don't need those, you can easily disable them and enjoy your shell-free boot (like I do every day). The shell-less boot systemd offers you is a unique feature on Linux.

Many of these small components are configured via configuration files in `/etc`.

Some of these are fairly standardized among distributions and hence supporting them in the C implementations was easy and obvious. Examples include: `/etc/fstab`, `/etc/crypttab` or `/etc/sysctl.conf`. However, for others no standardized file or directory existed which forced us to add `#ifdef` orgies to our sources to deal with the different places the distributions we want to support store these things. All these configuration files have in common that they are dead-simple and there is simply no good reason for distributions to distinguish themselves with them: they all do the very same thing, just a bit differently.

To improve the situation and benefit from the unifying force that systemd is we thus decided to read the per-distribution configuration files only as *fallbacks* -- and to introduce new configuration files as primary source of configuration wherever applicable. Of course, where possible these standardized configuration files should not be new inventions but rather just standardizations of the best distribution-specific configuration files previously used. Here's a little overview over these new common configuration files systemd supports on all distributions:

- `/etc/hostname`: the host name for the system. One of the most basic and

trivial system settings. Nonetheless previously all distributions used different files for this. Fedora used `/etc/sysconfig/network`, OpenSUSE `/etc/HOSTNAME`. We chose to standardize on the Debian configuration file `/etc/hostname`.

- `/etc/vconsole.conf`: configuration of the default keyboard mapping and console font.
- `/etc/locale.conf`: configuration of the system-wide locale.
- `/etc/modules-load.d/*.conf`: a drop-in directory for kernel modules to statically load at boot (for the very few that still need this).
- `/etc/sysctl.d/*.conf`: a drop-in directory for kernel sysctl parameters, extending what you can already do with `/etc/sysctl.conf`.
- `/etc/tmpfiles.d/*.conf`: a drop-in directory for configuration of runtime files that need to be removed/created/cleaned up at boot and during uptime.
- `/etc/binfmt.d/*.conf`: a drop-in directory for registration of additional binary formats for systems like Java, Mono and WINE.
- `/etc/os-release`: a standardization of the various distribution ID files like `/etc/fedora-release` and similar. Really every distribution introduced their own file here; writing a simple tool that just prints out the name of the local distribution usually means including a database of release files to check. The LSB tried to standardize something like this with the `lsb_release` tool, but quite frankly the idea of employing a shell script in this is not the best choice the LSB folks ever made. To rectify this we just decided to generalize this, so that everybody can use the same file here.
- `/etc/machine-id`: a machine ID file, superseding D-Bus' machine ID file. This file is guaranteed to be existing and valid on a systemd system, covering also stateless boots. By moving this out of the D-Bus logic it is hopefully interesting for a lot of additional uses as a unique and stable machine identifier.

- `/etc/machine-info`: a new information file encoding meta data about a host, like a pretty host name and an icon name, replacing stuff like `/etc/favicon.png` and suchlike. This is maintained by `systemd-hostnamed`.

It is our definite intention to convince *you* to use these new configuration files in your configuration tools: if your configuration frontend writes these files instead of the old ones, it automatically becomes more portable between Linux distributions, and you are helping standardizing Linux. This makes things simpler to understand and more obvious for users and administrators. Of course, right now, only systemd-based distributions read these files, but that already covers all important distributions in one way or another, except for one. And it's a bit of a chicken-and-egg problem: a standard becomes a standard by being used. In order to gently push everybody to standardize on these files we also want to make clear that sooner or later we plan to drop the fallback support for the old configuration files from systemd. That means adoption of this new scheme can happen slowly and piece by piece. But the final goal of only having one set of configuration files must be clear.

Many of these configuration files are relevant not only for configuration tools but also (and sometimes even primarily) in upstream projects. For example, we invite projects like Mono, Java, or WINE to install a drop-in file in `/etc/binfmt.d/` from their upstream build systems. Per-distribution downstream support for binary formats would then no longer be necessary and your platform would work the same on all distributions. Something similar applies to all software which need creation/cleaning of certain runtime files and directories at boot, for example beneath the `/run` hierarchy (i.e. `/var/run` as it used to be known). These projects should just drop in configuration files in `/etc/tmpfiles.d`, also from the upstream build systems. This also helps speeding up the boot process, as separate

per-project SysV shell scripts which implement trivial things like registering a binary format or removing/creating temporary/volatile files at boot are no longer necessary. Or another example, where upstream support would be fantastic: projects like X11 could probably benefit from reading the default keyboard mapping for its displays from `/etc/vconsole.conf`.

Of course, I have no doubt that not everybody is happy with our choice of names (and formats) for these configuration files. In the end we had to pick something, and from all the choices these appeared to be the most convincing. The file formats are as simple as they can be, and usually easily written and read even from shell scripts. That said, `/etc/bikeshed.conf` could of course also have been a fantastic configuration file name!

So, help us standardizing Linux! Use the new configuration files! Adopt them upstream, adopt them downstream, adopt them all across the distributions!

Oh, and in case you are wondering: yes, all of these files were discussed in one way or another with various folks from the various distributions. And there has even been some push towards supporting some of these files even outside of systemd systems.

Footnotes

[1] Our slogan: "*The only shell that should get started during boot is gnome-shell!*" -- Yes, the slogan needs a bit of work, but you get the idea.

Category: projects

[← BACK TO INDEX](#)

© Lennart Poettering. Built using Pelican. Theme by Giulio Fidente on [github](#).