How does systemd use /etc/init.d scripts?

Asked 7 years, 9 months agoModified 3 years, 2 months agoViewed 224k times

173

I just switched to debian jessie, and most things run okay, including my graphical display manager wdm.

The thing is, I just don't understand how this works. Obviously my /etc/init.d/wdm script is called, because when I put an early exit in there, wdm is not started. But when I alternatively rename the /etc/rc3.d directory (my default runlevel used to be 3), then wdm is still started.

I could not find out how systemd finds this script and I do not understand what it does to all the other init.d scripts.

- When and how does systemd run init.d scrips?
- In the long run, should I get rid of all init.d scripts?

systemd | init-script | init | sysvinit

ShareImprove this questionFollow

edited May 21, 2020 at 14:08
Ravexina
2,490 1 18 33

asked Oct 2, 2015 at 10:46
Martin Drautzburg
2,488 3 16 18

Sorted by: Highest score (defa

2 Answers

229

chaos' answer is what some documentation says. But it's not what systemd actually does. (It's not what van Smoorenburg rc did, either. The van Smoorenburg rc most definitely *did not* ignore LSB headers, which insserv used to calculate static orderings, for starters.) The Freedesktop documentation, such as that "Incompatibilities" page, is in fact wrong, on these and other points. (The HOME environment variable in fact *is* often set, for example. This went wholly undocumented anywhere for a long time. It's now documented in the manual, at least, but that Freedesktop WWW page still hasn't been corrected.)

The native service format for systemd is the *service unit*. systemd's service management proper operates *solely* in terms of those, which it reads from one of nine directories where (system-wide) .service files can live. /etc/systemd/system, /run/systemd/system, /usr/local/lib/systemd/system, and /usr/lib/systemd/system are four of those directories.

The compatibility with van Smoorenburg rc scripts is achieved with a conversion program, named systemd-sysv-generator. This program is listed in the /usr/lib/systemd/system-generators/ directory and is thus run automatically by systemd early in the bootstrap process at every boot, and again every time that systemd is instructed to re-load its configuration later on.

This program is a *generator*, a type of ancillary utility whose job is to create service unit files on the fly, in a tmpfs where three more of those nine directories (which are intended to be used only by generators) are located. systemd-sysv-generator generates the service units that run the van Smoorenburg rc scripts from /etc/init.d, if it doesn't find a native systemd service unit by that name already existing in the other six locations.

systemd service management only knows about service units. These automatically (re-)generated service units are written to invoke the van Smoorenburg rc scripts. They have, amongst other things:

```
[Unit]
SourcePath=/etc/init.d/wibble
[Service]
ExecStart=/etc/init.d/wibble start
ExecStop=/etc/init.d/wibble stop
```

Received wisdom is that the van Smoorenburg rc scripts must have an LSB header, and are run in parallel without honouring the priorities imposed by the /etc/rc?.d/ system. This is incorrect on all points.

In fact, they don't need to have an LSB header, and if they do not systemd-sysv-generator can recognize the more limited old RedHat comment headers (description:, pidfile:, and so forth). Moreover, in the absence of an LSB header it will fall back to the contents of the /etc/rc?.d symbolic link farms, reading the priorities encoded into the link names and constructing a before/after ordering from them, serializing the services. Not only are LSB headers not a requirement, and not only do they themselves encode before/after orderings that serialize things to an extent, the fallback behaviour in their complete absence is actually significantly non-parallelized operation.

The reason that /etc/rc3.d didn't appear to matter is that you probably had that script enabled via another /etc/rc?.d/ directory. systemd-sysv-generator translates being listed in any of /etc/rc2.d/, /etc/rc3.d/, and /etc/rc4.d/ into a native Wanted-By relationship to systemd's multi-user.target. Run levels are "obsolete" in the systemd world, and you can forget about them.

Further reading

- *systemd-sysv-generator*. systemd manual pages. Freedesktop.org.
- "Environment variables in spawned processes". *systemd.exec*. systemd manual pages. Freedesktop.org.
- https://unix.stackexchange.com/a/394191/5132
- https://unix.stackexchange.com/a/204075/5132
- https://unix.stackexchange.com/a/196014/5132
- https://unix.stackexchange.com/a/332797/5132

ShareImprove this answerFollow edited Feb 25, 2019 at 10:06 answered Oct 2, 2015 at 20:31

JdeBP
66.6k 12 159 343

2 In Debian the system-generators directory doesn't live on /usr/lib, but /lib packages.debian.org/sid/amd64/systemd/filelist – Braiam May 25, 2016 at 13:55

9 This is a straight up amazing answer. Well done sir. – peelman Jan 13, 2017 at 12:13

1 Thank you, thank you, thank you for this! Dealing with a mix of Debian 8 and RH/CentOS 7 systems has made management of SysVInit vs Systemd service dependency management a bit of a headache but

this explanation of what systemd is doing has helped my understanding greatly. – Toby Mar 29, 2017 at 17:22

This generator does work. I'd also mention, for followers, that if you have an older version of systemd and an /etc/init.d script isn't set to "boot on start" it will still work as expected but won't show up in the show-units lists: unix.stackexchange.com/a/518894/8337 – rogerdpack May 14, 2019 at 15:15

This generator does work. I'd also mention, for followers, that if you have an older version of systemd and an /etc/init.d script that isn't set to "boot on start" it will still start/stop as expected using systemctl but won't show up in the show-units lists:unix.stackexchange.com/questions/517872/... also NB that you basically "can't" control these service by running /etc/init.d/xx directly anymore or systemd gets...confused as to what is running and what isn't :| – rogerdpack May 23, 2019 at 16:40

21

Systemd is backward compatible with SysV init scripts. According to LSB 3.1, the init script must have informational Comment Conventions, defining when the script has to start/stop and what is required for the script to start/stop. This is an example:

```
### BEGIN INIT INFO
# Provides: my-service
# Required-Start: $local_fs $network $remote_fs
# Required-Stop: $local_fs $network $remote_fs
# Default-Start:  2 3 4 5
# Default-Stop: 0 1 6
# Short-Description: start and stop service my-service
# Description: my-service blah blah ...
### END INIT INFO
```

This is a commented section which is ignored by SysV. On the other hand, systemd reads that dependency information and runs those scripts depending on that.

But there is one point, where systemd and SysV differ in terms of init scripts. SysV executes the scripts in sequential order based on their number in the filename. Systemd doesn't. If dependencies are met, systemd runs the scripts immediately, without honoring the numbering of the script names. Some of them will most probably fail because of the ordering. There are a lots of other incompatibilities that should be considered.

If there are init scripts and .service files for the same service, systemd will execute both, as soon as the dependencies are met (in case of the init script, those defined in the LSB header).

Share Improve this answer Follow

edited Oct 2, 2015 at 20:34 — muru, 69.4k, 13, 192, 290

answered Oct 2, 2015 at 11:31 — chaos, 47.4k, 11, 118, 144

Okay, but I also have a whole bunch of .service files in /lib/systemd/system/. What does systemd actually execute? Whatever is specified in the service files (in dependency order), the init.d scripts or both? – Martin Drautzburg Oct 2, 2015 at 13:46

@MartinDrautzburg I added that to the answer – chaos Oct 2, 2015 at 13:58

1 As a sidenote, Debian has just announced to dump LSB compatibility: article.gmane.org/gmane.linux.debian.devel.lsb/1103 – Jan Oct 2, 2015 at 14:22

2 systemd is anything BUT compatible with SysV scripts. Not only is that statement incorrect, but the referenced link makes it clear that it is only "mostly compatible" and the amount of effort needed to produce the same results is outrageously huge. – Julie in Austin Feb 12, 2019 at 19:08

2 I found out the hard way that if the file in /etc/init.d is a symlink, the systemd generator "systemd-sysv-generator" that scans for legacy init.d scripts will skip it. My gerrit file in init.d was a symlink to /home/gerrit2/gerrit/bin/gerrit.sh, I fixed it with: cd /etc/init.d; sudo unlink gerrit; sudo cp /home/gerrit2/gerrit/bin/gerrit.sh gerrit – Integrator Apr 13, 2020 at 5:58