Blanks Netflix for a userbase edge case (Score:1, Troll)
by Nkwe (604125)
Headline implies that the scope of the problem is much bigger than it is. While I don't like systemd, it's not like systemd took out all of Netflix.

The problem is systemd breaking unexpectedly (Score:5, Insightful)
by Anonymous Coward July 24, 2017 @12:22PM (#54867573)
The real problem here isn't that a handful of Linux users couldn't use Netflix.
The real problem is that, yet again, systemd has been involved in critical functionality breaking in an unusual and unexpected way.
It's not systemd itself, but a critical library that systemd used that's responsible. Systemd is responsible for the problem because it uses this flawed library.
There's no reason for systemd to be involved with resolving domain names. Linux got by just fine throughout the 1990s, the 2000s, and even a big part of the 2010s without systemd being involved. Yet now that systemd is involved, things are going to hell.
Long time Linux users will be very aware of how problematic systemd so often is in the dumbest of ways.
Maybe somebody who just started using Linux in the systemd era thinks it's acceptable for their system to sometimes not boot properly, or for the domain name resolution to break unexpectedly. But long time Linux users know it wasn't like that before systemd was forced on the Linux community, and they know that such breakage is just not acceptable.
This is just the latest in a long chain of problems involving systemd. It has gotten to the point where Linux's reliability is below that of the BSDs, of macOS, and as much as I hate to say it, even modern versions of Windows!
Systemd needs to go, at least from important distros like Debian and Ubuntu. If Fedora wants to screw around with systemd, then so be it. But the other distros should remove it immediately.

Reply to This Parent Share

1 hidden comment

Re: (Score:3)
by AJWM (19027)
Hear, hear!
Why the hell does an init system need a built-in DNS resolver anyway?
2 hidden comments

Re: The problem is systemd breaking unexpectedly (Score:1)
by Anonymous Coward
...which is an utterly retarded design.
Unix is a bunch of components by different authors, most with competitors, that use well-defined protocols to communicate. Unix works because stuff that sucks gets replaced, and no one person's vision defines what happens.
Systemd and Windows are defined by one small man's vision, not by protocols and competition. And when that man doesn't think usernames should have certain forms, well, fuck everyone else, right?

Re: (Score:2)
by Holi (250190)
Exactly how is this insightful? The parent is going on a rant about systemd when it was libidn2 that had the bug.
2 hidden comments

Re: (Score:2)
by Rakarra (112805)
His question was "why is systemd doing that instead of something else?"

Re: The problem is systemd breaking unexpectedly (Score:1)
by Anonymous Coward
Yeah, as anyone knows before systemd arised we were unable to resolve hostnames, thank you for proving you don't know shit about what you're talking about.

READ THE FUCKING COMMENT! It addresses that! (Score:2, Informative)
by Anonymous Coward

it was libidn2 that had the bug.
NO SHIT! Did you even bother to read the comment before replying to it, and before wrongly criticizing it?! OBVIOUSLY NOT! The comment you didn't read, yet still replied to, contained the following:
It doesn't matter if it was an external library that systemd used that's responsible. Systemd is responsible for the problem because it uses this flawed library.
By choosing to use this foreign library, the foreign library code effectively becomes part of systemd. If a user invokes systemd to perform some action, but systemd does the wrong thing because it uses a broken library, then it's both the library that's broken and it's systemd that

Re: (Score:3)
by ncc74656 (45571) *
Exactly how is this insightful? The parent is going on a rant about systemd when it was libidn2 that had the bug.
If systemd hadn't taken it upon itself to handle DNS resolution instead of sticking to its ostensibly primary job (initd), it would never have had reason to pull in libidn2 and fall to one of its bugs.
The Unix Way. systemd fails it.

Re:The problem is systemd breaking unexpectedly (Score:5, Insightful)
by squiggleslash (241428) on Monday July 24, 2017 @01:31PM (#54868083) Homepage Journal
No, the real problem is that a library, Libidn, that's used by resolver libraries including that apparently shipped with systemd has a bug in it. The library dates back to 2002, it's not even as if systemd was relying upon some bleeding edge library written specifically for it. And yes, it's best practices, when implementing something like international domains to use a respected third party library rather than trying to roll your own, so they haven't made an error in relying upon it.
This has nothing to do with systemd except for the fact the user happened to be using systemd at the time, and systemd happens to use this library. What next? A kernel bug gets blamed on systemd because systemd uses the kernel?
The submitter is trolling.
Reply to This Parent Share
twitter facebook linkedin
Flag as Inappropriate

Systemd is responsible for the libraries it uses! (Score:1)
by Anonymous Coward
Did you even bother to read the comment before replying to it, and before wrongly criticizing it?! OBVIOUSLY NOT! The comment you didn't read, yet still replied to, contained the following:
It doesn't matter if it was an external library that systemd used that's responsible. Systemd is responsible for the problem because it uses this flawed library.
By choosing to use this broken library, the broken library code effectively becomes part of systemd. If a user invokes systemd to perform some action, but systemd does the wrong thing because it uses a broken library, then it's both the library that's broken and it's systemd that's broken. Systemd can't be excused just be

Re: (Score:3)
by fahrbot-bot (874524)
What next? A kernel bug gets blamed on systemd because systemd uses the kernel?
Wait! Who uses who now? :-)
Sorry, I'm from the future, where there is no kernel, only systemd.
Fun facts, after subsuming the kernel, the last non-systemd user land utility remaining is Emacs. Lennart and his (remaining) crew started the battle to absorb Emacs in 2040, five years before his death, and it's still raging many years after that. There have been casualties on both sides. Lennart died in 2045 when the experimental "systemd-elisp" module controlling his cold, robotic heart turned out to contai

Re: (Score:1)
by Anonymous Coward
Great, now Poettering is going to take that as a death threat and write another livejournal about how mean the whole FOSS community is to him.

Re: (Score:3)
by zdzichu (100333)
Actually, the bug is not in libidn, but in libidn**2**. Or rather was – it got fixed rather quickly – https://gitlab.com/libidn/libi... [gitlab.com]
As for systemd, it uses libidn by default. libidn2 support is marked as **experimental** – reasonable decision as this bug shows.
The submitted article is pure flamebait - this is not a bug in systemd suite, but in 3rd party library; to experience this (already fixed) bug, distribution would have to have enabled experimental option. No sane distro does that.
Nb. The

Re: (Score:1)
by Anonymous Coward
>And yes, it's best practices, when implementing something like international domains to use a respected third party library rather than trying to roll your own, so they haven't made an error in relying upon it.
They have made an error in relying upon it. Please, let's not extend consensus-science-bullshit to computer science.
1 hidden comment

Re:The problem is systemd breaking unexpectedly (Score:5, Informative)
by dgatwood (11270) on Monday July 24, 2017 @01:39PM (#54868151) Homepage Journal
The real problem is that, yet again, systemd has been involved in critical functionality breaking in an unusual and unexpected way.
No, the real problem is that Netflix violated RFC 1034 section 3.5 [ietf.org] and RFC 1035 section 2.3.1 [ietf.org], which both explicitly say that hostnames must still conform to the old ARPANET restrictions, which allow only letters, numbers, and hyphens. **Underscores have never been legal in DNS hostnames**, and in spite of the pain this spec-compliant behavior has caused for some users, the systemd behavior is correct, and Netflix needs to fix whatever broken software they have that incorrectly created an invalid hostname containing an underscore.
The remarkable thing, frankly, is that **any** DNS resolver resolved that address, and more significantly, that the DNS servers actually responded to the request.
Reply to This Parent Share
twitter facebook linkedin
Flag as Inappropriate
1 hidden comment
Re: (Score:2)
by skids (119237)
the systemd behavior is correct
Yes, it is. Well. to the extent that systemd being used by the system for DNS resolution is correct, as opposed to using a real DNS resolver. The extra junk in systemd should only be used to bootstrap containers and VMs and should be replaced during boot with real services. And really, systemd and/or its packagers should ship a version without that crap for people not doing VM/container stuff so it doesn't get in their way or pull in unwanted dependencies.

Re: (Score:3)
by DamnOregonian (963763)
It's not quite that clear cut.
RFC 2181 11. Name syntax: The DNS itself places only one restriction on the particular labels that can be used to identify resource records. That one restriction relates to the length of the label and the full name. The length of any one label is limited to between 1 and 63 octets. A full domain name is limited to 255 octets (including the separators). The zero length full name is defined as representing the root of the DNS tree, and is typically written and displayed as ".". Those restrictions aside, any binary string whatever can be used as the label of any resource record. Similarly, any binary string can serve as the value of any record that includes a domain name as some or all of its value (SOA, NS, MX, PTR, CNAME, and any others that may be added). Implementations of the DNS protocols must not place any restrictions on the labels that can be used. **In particular, DNS servers must not refuse to serve a zone because it contains labels that might not be acceptable to some DNS client programs.** A DNS server may be configurable to issue warnings when loading, or even to refuse to load, a primary zone containing labels that might be considered questionable, however this should not happen

by default.
These days, it is up to the client to validate the labels being requested in its own context, but otherwise, anything goes.
The "client" in this instance, has been forced to use a resolver that decides to validate for all clients that may be using it, which is entirely incorrect behavior.