Ceph - Bug #22125

systemctl commands use literal '*'

11/14/2017 04:11 PM - Matthew Richardson

Status:	Rejected		**% Done:**	0%
Priority:	Normal		**Spent time:**	0.00 hour
Assignee:	Boris Ranto			
Category:	build			
Target version:	v12.2.1			
Source:			**Affected Versions:**	
Tags:			**ceph-qa-suite:**	
Backport:			**Pull request ID:**	
Regression:	No		**Crash signature (v1):**	
Severity:	3 - minor		**Crash signature (v2):**	
Reviewed:				

Description

Many systemd commands which are run in the rpm scripts use escaped asterisks, e.g.:
https://github.com/ceph/ceph/blob/master/ceph.spec.in#L1004

```
/usr/bin/systemctl preset ceph-disk@\*.service ceph.target
```

This escaping causes systemd to escape it as a literal asterisk, resulting in files/links named e.g. 'ceph-disk@\x2a.service' which obviously doesn't work or do anything useful. (the process for escaping is documented in the systemd-escape man page). If globbing was intended, no escaping is necessary (globbing support was added in systemd 209).

History

#1 - 11/15/2017 02:41 PM - Sage Weil

- Status changed from New to Triaged

- Assignee set to Boris Ranto

#2 - 11/15/2017 08:51 PM - Boris Ranto

- Status changed from Triaged to Rejected

These lines are first interpreted by bash. The commands are supposed to use the literal * for glob (which they do since bash will strip away the \ character). The escaping is there for bash not to interpret the *. If we did not escape it and there were some files named e.g. ceph-disk@file1.service and ceph-disk@file2.service then bash would interpret the star and run

```
# /usr/bin/systemctl preset ceph-disk@file1.service ceph-disk@file2.service ceph.target
```

which is definitely wrong. This is even worse with shells like zsh where if the * was not escaped and there were no files matching the expression you would get an error like this

```
# echo abc*def
zsh: no matches found: abc*def
```

This (the zsh bit) is not that much of an issue with respect to spec files since rpm runs the scripts in bash but I just thought it is worth mentioning since I have seen a lot of bugs in code where people expected the default shell to be bash and did not escape the * which caused some serious trouble for some people. (and it even caused some serious trouble with bash)

I am closing this since it is not a bug, feel free to re-open if you have some piece of information that I might be missing.

#3 - 11/16/2017 10:56 AM - Matthew Richardson

OK - I've re-grokked what is happening, and yes, the * does need to be escaped to prevent shell file globbing. However this got me thinking and I believe there's a more fundamental issue here.

systemd interprets * literally in certain situations - but uses it as a wildcard in others.

This is because it is impossible for some systemctl commands (like enable or preset) to know what to expand wildcards to. For example, on a host with 2 existing osds enabled:

```
> ls /etc/systemd/system/ceph-osd.target.wants/
ceph-osd@0.service  ceph-osd@1.service

# Expect a no-op, since the services are already enabled.
> systemctl enable ceph-osd@\*.service
Created symlink from /etc/systemd/system/ceph-osd.target.wants/ceph-osd@\x2a.service to /usr/lib/systemd/syste
m/ceph-osd@.service.

> ls /etc/systemd/system/ceph-osd.target.wants/
ceph-osd@0.service  ceph-osd@1.service  ceph-osd@\x2a.service

# Try to disable all ceph-osd services
> systemctl disable ceph-osd@\*.service
Removed symlink /etc/systemd/system/ceph-osd.target.wants/ceph-osd@\x2a.service.

# run again, in case previous presence of literal * stopped wildcard expansion
> systemctl disable ceph-osd@\*.service

> ls /etc/systemd/system/ceph-osd.target.wants/
ceph-osd@0.service  ceph-osd@1.service
```

As you can see, enable and disable (and others like preset) always treat * as literal.

As a result I suspect that several of the systemctl commands in the rpm scripts aren't doing whatever they were intended to do.

#4 - 11/16/2017 12:35 PM - Boris Ranto

You are right that disable/enable treats * as literal (and it makes sense there) but we do not use them in the scripts. The preset itself should do the * expansion (and in both my test systems it does). There might be a bug in the systemd version that you use that does not do that though (I do recall there were a couple of issues with * expansion in some versions of systemd :-/).

fyi: I tested with systemd-231-17 in fedora and systemd-219-20 in rhel.

#5 - 11/21/2017 08:21 PM - Nathan Cutler

@Matthew: For completeness, can you write which version of systemd you tested with?

#6 - 11/22/2017 01:30 PM - Matthew Richardson

Hi - it's Scientific Linux: systemd-219 - installed from rpm systemd-219-30.el7_3.3

However... I've rebuilt my test box to try and replicate this bug from scratch, and now I can't. Something obviously created the '\x2a' service files in the last iteration, but they don't seem to be being created this time. My last install was an upgrade from a running jewel cluster, whereas this time it was a fresh install, so that might be related, but at this point I'm happy for this to be closed and I'll just keep an eye out for it in future, and re-open if I can find a way to reliably reproduce this (or if anyone else hits the same situation).

Sorry for this dead-end!