

FreeNAS - Bug #35539

Make BMC firmware version check more robust

06/21/2018 11:46 AM - Bonnie Follweiler

Status: Done	
Priority: No priority	
Assignee: Ryan Moeller	
Category: OS	
Target version: 11.2-BETA2	
Seen in: 11.2-BETA1	Needs Merging: No
Severity: Low	Needs Automation: No
Reason for Closing:	Support Suite Ticket: n/a
Reason for Blocked:	Hardware Configuration:
Needs QA: No	ChangeLog Required: No
Needs Doc: No	
Description	
This is from a Clean install of FreeNAS 11.2-INTERNAL8 (screenshots provided)	

Associated revisions

Revision ab74a038 - 07/10/2018 05:11 PM - Ryan Moeller

Make BMC firmware version check more robust

The BMC firmware for C2750D4I and C2550D4I mainboards shipped in the FreeNAS Mini product line had a broken watchdog timer that was fixed in revision 0.30. The version check we used to decided how to configure watchdogd assumed the format of the version string reported by ipmitool, and only checked a portion of the string.

This change performs more checks on the version string, failing more gracefully if the format is for any reason unacceptable.

Ticket: #35539

Revision 2a0ba286 - 07/11/2018 12:49 PM - Ryan Moeller

Load ipmi kernel module before invoking ipmitool

ipmitool does not load the module itself and will fail if the module is not loaded. The ipmi kmod is not loaded early enough by kld_list. The module is only needed this early on specific mainboards that require a BMC firmware revision check to work around a firmware bug.

Ticket: #35539

Revision 8c05a2f6 - 07/12/2018 01:20 PM - Ryan Moeller

Load ipmi module from loader.conf

This ensures the module is loaded in time to perform a BMC firmware revision check in rc.conf.local using ipmitool.

Ticket: #35539

History

#1 - 07/05/2018 11:39 AM - Alexander Motin

- Assignee changed from Alexander Motin to Ryan Moeller

Ryan, take a look at this please. I guess it can be couple of one-liners, some missing quotes or incorrect condition somewhere in shell scripts. The most interesting part is to find where it happens. :)

#2 - 07/05/2018 11:43 AM - Ryan Moeller

- Status changed from Unscreened to Screened

#3 - 07/06/2018 10:10 AM - Ryan Moeller

- Status changed from Screened to In Progress

#4 - 07/06/2018 05:26 PM - Ryan Moeller

- Severity changed from New to Low

I was unable to reproduce this issue in my test environment, but I have made a few noteworthy observations.

The error message itself indicates that an empty string is being tested by one of the integer comparison operators (-eq, -ne, -gt, -ge, -lt, -le). This is possible if an undefined or empty variable is quoted in the test, such as

```
# x could be undefined or simply empty
x=""
[ "${x}" -eq 0 ]
```

Without the quotes, a different error would be shown.

The locations of the two error messages are also informative. The first appearance occurs between the following lines in install.sh:

```
chroot /tmp/data /usr/sbin/mtree -deUf /etc/mtree/BSD.var.dist -p /var
# We need this hack due to sqlite3 called from rc.conf.local.
chroot /tmp/data /sbin/ldconfig /usr/local/lib
chroot /tmp/data /etc/rc.d/ldconfig forcestart
```

We see the end of the mtree output before the error, and the ldconfig output after.

The second appearance is during boot right after the root fs is mounted and before networking is brought up. The ldconfig rc script runs at this time. However, there don't appear to be any integer comparisons in that script directly.

Because the ldconfig script sources and calls functions from rc.subr, which pulls in various configuration files and other scripts, it is difficult to pinpoint the exact location of the error without being able to reproduce it. A potential file of interest is rc.conf.local, which has several locations matching the pattern described above:

```
...
if [ "${ssl}" -gt "0" ]
...
if [ "${count}" -gt "0" ]; then
```

```
...
elif [ "${product}" = "C2750D4I" -o "${product}" = "C2550D4I" ] \
  && [ "$(/usr/local/bin/ipmitool mc info 2> /dev/null \
    | grep ^Firmware | cut -d . -f 2)" -lt 30 ] ; then
...

```

The first two get those variables from sqlite queries, which I expect would not be so inconsistent between clean installations, even on different systems.

The ipmitool line looks especially interesting. Would this happen to be one of those products?

#5 - 07/09/2018 03:25 PM - Ryan Moeller

- Status changed from *In Progress* to *Blocked*

- Reason for *Blocked* set to *Need additional information from Author*

Testing on a FreeNAS Mini 2.0 with baseboard product name "C2750D4I" and IPMI management controller firmware revision "0.35" did not reproduce the error, however it is possible that a different hardware configuration or firmware revision triggers the bug.

Bonnie, can you describe the hardware configuration with regard to the baseboard product name and the management controller firmware revision, which can be determined by the following shell commands:

```
# dmidecode -s baseboard-product-name | head -1
# ipmitool mc info | grep ^Firmware

```

#6 - 07/11/2018 07:34 AM - Bonnie Follweiler

- Status changed from *Blocked* to *Unscreened*

This is the result:

```
freenas# dmidecode -s baseboard-product-name | head -1
C2750D4I
freenas# ipmitool mc info | grep ^Firmware
Firmware Revision      : 0.35
freenas#

```

#8 - 07/11/2018 12:34 PM - Ryan Moeller

- File *IMG_6336.jpeg* added

- Status changed from *Unscreened* to *In Progress*

- Reason for *Blocked* deleted (*Need additional information from Author*)

Bonnie and I have worked together to confirm that the cause of the problem is indeed the BMC firmware revision check.

During install and early boot, the ipmi kernel module is not loaded. This causes the ipmitool command to fail, resulting in an empty string as the number being checked.

The cause of the error was hidden by stderr of ipmitool being redirected to /dev/null. Removing the redirection allowed the error to be identified, as shown in the attached image.

In my previous testing on a similar system, I failed to reproduce the error because the ipmi module eventually gets loaded and so the test I performed on a running system passed. I did not have remote access to the console to observe the error at boot.

Impact:

This bug causes systems with a known issue in the BMC firmware to incorrectly configure the watchdog daemon, in addition to the error message.

Solution:

The fix will be to load the ipmi module before the ipmitool is invoked in /etc/rc.conf.local.

Acceptance criteria:

The error message should not appear during installation or boot on a system with main board "C2750D4I" or "C2550D4I".

A system with main board "C2750D4I" or "C2550D4I" and BMC firmware prior to revision 0.30 should correctly generate the watchdogd flags in /tmp/rc.conf.freenas at boot time to work around the firmware bug. It is not essential to correctly generate those flags during installation, as the configuration is regenerated each boot.

#9 - 07/12/2018 02:57 PM - Ryan Moeller

- Subject changed from *On the console during the initial install, and during each reboot, a line [: : bad number shows up to Make BMC firmware version check more robust*

- Status changed from *In Progress to Ready for Testing*

- Needs Merging changed from *Yes to No*

Master PR: <https://github.com/freenas/freenas/pull/1510>

#10 - 07/12/2018 06:00 PM - Dru Lavigne

- Target version changed from *Backlog to 11.2-BETA2*

- Needs Doc changed from *Yes to No*

#12 - 07/13/2018 09:46 AM - Bonnie Follweiler

- Status changed from *Ready for Testing to Passed Testing*

- Needs QA changed from *Yes to No*

Test Passed in FreeNAS-11.2-MASTER-201807131206
(Build Date: Jul 13, 2018 15:11)

#13 - 07/13/2018 02:16 PM - Dru Lavigne

- Status changed from Passed Testing to Done

Files

bad Number 1.jpg	139 KB	06/21/2018	Bonnie Follweiler
bad number 2.jpg	156 KB	06/21/2018	Bonnie Follweiler
IMG_6336.jpeg	154 KB	07/11/2018	Ryan Moeller