Tails - Bug #11786
System often crashes during/after memory wipe since Linux 4.6
09/08/2016 11:30 AM - mercedes508

<table>
<thead>
<tr>
<th>Status:</th>
<th>Duplicate</th>
<th>Start date:</th>
<th>09/08/2016</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Normal</td>
<td>Due date:</td>
<td></td>
</tr>
<tr>
<td>Assignee:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Category:</td>
<td>Hardware support</td>
<td>% Done:</td>
<td>20%</td>
</tr>
<tr>
<td>Target version:</td>
<td></td>
<td>Estimated time:</td>
<td>0.00 hour</td>
</tr>
<tr>
<td>Feature Branch:</td>
<td></td>
<td>Spent time:</td>
<td>0.00 hour</td>
</tr>
<tr>
<td>Type of work:</td>
<td>Code</td>
<td>Starter:</td>
<td></td>
</tr>
<tr>
<td>Blueprint:</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Description
We received a few reports from users saying that memory wipe, previously working with Tails 2.5 isn't working anymore with Tails 2.6~rc1. I guessed because of Linux 4.6.

I'm not sure how this should be considered, even though it clearly is a regression.

Related issues:
- Related to Tails - Bug #10733: Run our initramfs memory erasure hook earlier
  - Resolved 12/09/2015
- Related to Tails - Bug #9707: Jessie: System sometimes does not poweroff after...  
  - Rejected 07/08/2015
- Related to Tails - Bug #12061: Long delay before memory wipe starts on Stretch
  - Resolved 12/21/2016
- Duplicated by Tails - Bug #11730: Memory erasure freezes more often on Stretch
  - Duplicate 08/26/2016
- Duplicates Tails - Bug #12354: Fix shutdown and memory wipe regressions on 3....
  - Resolved 03/20/2017

Associated revisions
Revision 7f88af1d - 09/16/2016 07:15 PM - anonym
Revert "Run our initramfs memory erasure hook earlier."

This reverts commits 185f53877ba90b621dbade4aebc4903e07e6ea82 and 71ac15e8a9d84bbb66260530bbdf1771a8adfffc.

Since we introduced this in Tails 2.6~rc1 we see a lot more issues with memory wiping both in our automated tests (locally, and on Jenkins) and among actual users.

Refs: #10733, #11786

Revision be9b18da - 12/22/2016 01:15 PM - intrigeri
Display more information after wiping memory (refs: #11786).

The idea is to enable:
- users to report more useful bugs;
- developers to better understand the failure we see in the test suite.
Work around Tails freezing during memory wiping.
This should be reverted once #11786 is fixed properly.
This commit is a backport from the tails/stretch of the two workarounds we implemented there.
Refs: #10776, #11786

Work around another instance of #11786.
I.e. when we get a kernel stack trace instead of a BusyBox shell.
Refs: #11786

Try to identify #11786 explicitly.
Actually this will only make us fail faster.
Refs: #11786

Can you please add some details to the "isn't working anymore" part? Description of what happens exactly, what's on screen, screenshots, would be useful.
Also, is there any chance that you get the bug reporters to test another, experimental ISO image, if I provided one?
Is this problem happening every time with 2.6~rc1 on the bug reporters' hardware?
This looks like the problem that's being discussed at #10733#note-21. It's unclear if that's caused by the Linux kernel upgrade, or by the changes introduced for #10733.
Can you please add some details to the "isn't working anymore" part? Description of what happens exactly, what's on screen, screenshots, would be useful.

We received a more detailed bug report:

Shutdown on Tails 2.5:
1. Screen shows "[ OK ] Started Modem Manager", followed by "You can now remove...", followed by "Starting new kernal"
2. Screen fills with asterisk as memory is wiped, followed by "clean exit"
3. System turns off

Shutdown on Tails 2.6~rc1:
1. Screen shows "You can now remove...", followed by "kexec_core: Starting new kernal"
2. Systems turns off
   Issue occurs every time.

[...]

Hi elouann! This ticket is about memory wipe failing. In that report, the system turns off so memory wipe succeeded, so this seems off-topic here unless I've missed something ⇒ please report whatever other problem you wanted to report in a dedicated ticket. Thanks in advance! (If it's about the lack of visual feedback (asterisks) on screen: presumably that's caused by the upgrade to Linux 4.6, not much we can do about it, and having this feedback working or not has always been highly hardware-dependent; the best we can do is to ensure our doc/FAQ/known issues/whatever is good enough to avoid doc-reading-user confusion.)
In my case, 2.5 was fine, but on 2.6~rc1 the screen shows "You can now remove...", followed by "kexec_core: Starting new kernel", and it just hangs there. It never progresses to the next screen filled with asterisks and the system never turns off, I have to manually hold the power button until it turns off. So I assume the memory isn't getting wiped. This happens every time on 2.6~rc1.

intrigeri wrote:

Also, is there any chance that you get the bug reporters to test another, experimental ISO image, if I provided one?

I could test an experimental ISO if needed.

Can you please test this ISO image:

If you could do it before Monday it might influence what we do for the final Tails 2.6 release.
If you could do it before Monday it might influence what we do for the final Tails 2.6 release.

I tested the ISO, booting and shutting down several times, with the same results every time. Here are the results I got:

1. Screen shows "You can now remove...", followed by "kexec_core: Starting new kernel".
2. Screen fills with asterisks.
3. Kernel panic (see output below). It hangs at the end of the output.
4. System does not turn off automatically. Need to hold down the power button until it turns off.

So it now progresses past "Starting new kernel" to the screen filled with asterisks, which looks like memory wipe is working again, but now there's a kernel panic issue that prevents powering off.

Kernel panic output:

```
[ 5.938190] usb 2-4: string descriptor 0 read error: -22
modprobe: module microcode not found in modules.dep
Starting Wiping the memory, press Control-C to abort earlier. Help: "/usr/bin/sdmem -h"
Wipe mode is insecure (one pass with 0x00)
********************************************************************************
[Edit: ...]
********************************************************************************Killed
/scripts/init-premount/memory_wipe: line 57: can't fork
/init: /scripts/init-premount/ORDER: line 5: can't fork
[ 8.033091] Kernel panic - not syncing: Attempted to kill init! exitcode=0x00000200
[ 8.033091] [ 8.033143] CPU: 1 PID: 1 Comm: init Tainted: G          E   4.6.0-0bpo.1-amd64 #1 Debian 4.6.4-1-bpo8+1
[ 8.033192] Hardware name: [Edit: hardware redacted]
[ 8.033224] [Edit: hardware redacted]
[ 8.033274] [Edit: hardware redacted]
[ 8.033323] [Edit: hardware redacted]
[ 8.033371] Call Trace:
[ 8.033392] [<ffffffff813124c5>] ? dump_stack+0x5c/0x77
[ 8.033424] [<ffffffff8117069f>] ? panic+0x2f/0x226
[ 8.033455] [<ffffffff8107f448>] ? do_exit+0xb6/0xb0
[ 8.033486] [<ffffffff8107f4f9>] ? do_group_exit+0x10/0x10
[ 8.033519] [<ffffffff81003e28>] ? do_int80_syscall_32+0x58/0x160
[ 8.033552] [<ffffffff815ca56e>] ? entry_INT80_compat+0x3e/0x50
[ 8.033587] [<ffffffff813124c5>] ? dump_stack+0x5c/0x77
[ 8.033648] ---[ end Kernel panic - not syncing: Attempted to kill init! exitcode=0x00000200
[ 8.033648] [ 8.033648] 03/13/2020
```

03/13/2020
For the record, I've also observed this kernel panic on Tails with Linux 4.7 (KVM guest).

- Target version changed from Tails_2.6 to Tails_2.7

I received that screenshot. That user will soon test the following ISO:


With a Thinkpad x201

Tails 2.5:
Laptop shuts down ok, with asterisks and everything

Tails 2.6rc~1 and Tails 2.6:
Laptop shuts down but no asterisks.

tails-i386-feature_10733-reverted-2.6-20160918T1709Z-7f88af1+testing@f8e35e3:
Laptop shuts down ok, with asterisks and everything

Did you repeat these tests so there's some statistical significance in your results? Otherwise, can you please redo each test at least five times (clearly significant! :P) or so and report back?

Also, what results do you get from a recent [devel build](http://nightly.tails.boum.org) which has Linux 4.7?

Related to Bug #9707: Jessie: System sometimes does not poweroff after memory erasure added
#14 - 09/28/2016 07:50 AM - anonym
- Related to Bug #11730: Memory erasure freezes more often on Stretch added

#15 - 09/29/2016 07:15 PM - Anonymous
Can confirm no asterisk just power off
in 2.5 everytime with asteriks then off with 2.6 no single time with asteriks just power off

#16 - 10/02/2016 10:33 AM - intrigeri
- Related to deleted (Bug #11730: Memory erasure freezes more often on Stretch)

#17 - 10/02/2016 10:33 AM - intrigeri
- Related to deleted (Bug #11730: Memory erasure freezes more often on Stretch)

#18 - 10/02/2016 10:33 AM - intrigeri
- Duplicated by Bug #11730: Memory erasure freezes more often on Stretch added

#19 - 10/02/2016 10:52 AM - intrigeri
- Subject changed from Memory wipe fails more often since Linux 4.6 to System often crashed during/after memory wipe since Linux 4.6
- Assignee changed from emmapeel to intrigeri
- QA Check deleted (Info Needed)
- Type of work changed from Test to Code

So, it seems that we have at least two problems here:

1. the progress feedback (asterisks) has disappeared in some cases; that's fixed by reverting the changes introduced at #10733; presumably memory wipe succeeds anyway, so it's a minor UX problem only. I'm not convinced it's worth working on at this point; if someone feels differently, please file a separate ticket as requested on #11786#note-4.
2. the system crashes somehow during/after memory wipe (with kernel panic and/or being dropped to a busybox shell) with Linux 4.6 and 4.7, and as a result fails to poweroff; it looks like memory wipe might have succeeded in all cases though; it also looks like the problem is related to the system being out of memory for some reason. Next debugging step is to look more closely at our memory wipe initramfs script, and check what can possibly cause too much memory pressure. IMO we can do this without waiting for emmapeel's results, so reassigning to myself (but emmapeel: please do answer the questions anonym asked you).

#20 - 10/02/2016 11:28 AM - intrigeri
- Subject changed from System often crashed during/after memory wipe since Linux 4.6 to System often crashes during/after memory wipe since Linux 4.6

#21 - 10/02/2016 12:02 PM - intrigeri
anonym, emmapeel, elouann, goupille, mercedes508: did you see this bug with a system that has no more than 2GB of RAM?

#22 - 10/02/2016 12:24 PM - intrigeri
Bug reporters & frontdesk: did you see this bug on systems that are are not "64-bit CPU + strictly more than 2GB of RAM"?
I've verified that busybox behaves sanely wrt. how we deal with background processes: in a busybox shell within our initramfs, (sleep 30) & (sleep 20) & (sleep 10) & wait returns after 30 seconds. I've also verified that killall sdmem works in that context.

I suspect that our "one instance of sdmem per 2 GiB of RAM" tweak causes this bug on 64-bit systems with strictly more than 2GB of RAM, where we uselessly create a risky situation with a bunch of sdmem process each trying to fill all memory, and rely on the OOM killer to behave properly... and
for some reason we see that it sometimes tries to kill init. This does not explain the case when poweroff can't fork though: when we reach this point, if everything had gone well all sdmem processes would have been killed so there's no reason why there's no memory left, so I really don't understand what's happening there.

Still, I propose that we use the "one instance of sdmem per 2 GiB of RAM" tweak on 32-bit systems only (#8183 already drops it with $619c536a458177e7972ad8e0086a4174ee57a60d$). I'm not sure it will fix this bug, but it makes sense to first eliminate some useless complexity, which can help us debug the problem further if it's not fixed by that change. I'll prepare a branch that does so.

#23 - 10/02/2016 12:28 PM - intrigeri
- Status changed from Confirmed to In Progress
- % Done changed from 0 to 10
- Feature Branch set to bugfix/11786-simplify-memory-wipe-on-64-bit

intrigeri wrote:

Still, I propose that we use the "one instance of sdmem per 2 GiB of RAM" tweak on 32-bit systems only (#8183 already drops it with $619c536a458177e7972ad8e0086a4174ee57a60d$). I'm not sure it will fix this bug, but it makes sense to first eliminate some useless complexity, which can help us debug the problem further if it's not fixed by that change. I'll prepare a branch that does so.

Done on the topic branch, let's see what Jenkins thinks about it.

#25 - 10/03/2016 08:06 AM - intrigeri
Interesting: with that change in, according to Jenkins only ~2.5-3GB is wiped. The video shows that sdmem prints "done" before we display our "Happy dumping" message. I wonder if that's caused by our 32-bit userspace (sorry I'm quite clueless in such low-level stuff).

#26 - 10/03/2016 05:57 PM - anonym
intrigeri wrote:

Interesting: with that change in, according to Jenkins only ~2.5-3GB is wiped. The video shows that sdmem prints "done" before we display our "Happy dumping" message. I wonder if that's caused by our 32-bit userspace (sorry I'm quite clueless in such low-level stuff).

I am very certain our 32-bit userspace is the issue. After all, a 32-bit program will not "learn" how to deal with 64-bit pointers just because the kernel knows it -- in the compiled code, the size of pointers are 32-bit still, and no matter what you do you cannot fit a 64-bit pointer. 4 GiB of virtual memory per process is the hard limit.

The only way to save this KISS approach, I guess, is to ship a statically linked 64-bit sdmem in our initramfs... ;)}
intrigeri wrote:

anonym, emmapeel, elouann, goupille, mercedes508: did you see this bug with a system that has no more than 2GB of RAM?

Note that our automated test suite defaults to VMs with exactly 2 GiB of RAM, and it frequently happens when running it. FWIW I also run other Tails VMs with that exact amount of RAM, with the same problem.

#28 - 10/04/2016 07:08 AM - intrigeri

Note that our automated test suite defaults to VMs with exactly 2 GiB of RAM, and it frequently happens when running it. FWIW I also run other Tails VMs with that exact amount of RAM, with the same problem.

OK, too bad. So my hypothesis was wrong and it's not worth fixing the issue with my branch (that I'm going to delete).

#29 - 10/04/2016 07:08 AM - intrigeri

- Feature Branch deleted (bugfix/11786-simplify-memory-wipe-on-64-bit)

#30 - 10/09/2016 08:13 PM - intrigeri

FWIW, since the beginning of October:

- among 22 test suite runs on the stable branch (Linux 4.6), I see 5 such failures
- among 6 test suite runs on the devel branch (Linux 4.7), I see no such failure

I've just started a few runs on the devel branch to confirm whether the move to 4.7 has fixed this bug.

Meanwhile, frontdesk: please have bug reporters test recent builds from the devel branch.

#31 - 10/10/2016 02:50 PM - intrigeri

- Assignee changed from intrigeri to bertagaz
- % Done changed from 10 to 20
- QA Check set to Info Needed

Update -- since the beginning of October:

- among 24 test suite runs on the stable branch (Linux 4.6), I see 6 such failures => 25% failure rate
• among 13 test suite runs on the devel branch (Linux 4.7), I see no such failure => 0% failure rate

… so it might indeed be that the upgrade to 4.7 fixes this bug.

Next steps:

• emmapeel reports about testing current devel on ThinkPad X201 (she already knows she has to do that).
• the RM for Tails 2.7 (bertagaz) comments about the idea of upgrading to Linux 4.7 in our next point-release (cherry-picking a few commits from feature/11818-linux-4.7 should be enough, but perhaps it's more involved if it implies either bumping our debian APT snapshot or importing src:linux 4.7 into our custom APT repo): bertagaz, what do you think? Is this bug bad enough in your opinion to go through this and the risk of other regressions?

#32 - 10/10/2016 04:09 PM - emmapeel

So, I have tested several times each with a Thinkpadx201 and:

- Tails 2.6:
  When clicking restart, the message about ‘now you can take your USB out’ appears, and after approx. 15 seconds of it, the laptop restarts (no stars! :S)
- tails-i386-devel-2.7-20161005T0955Z-bc6a783
  No change with relation to 2.6
- Tails 2.5:
  Similar to Tails 2.6, takes the same amount of time, but I get the screen full as asterisks before restarting, at some point inside the 15 seconds window (around 10 secs)

#33 - 10/13/2016 12:33 PM - bertagaz

- QA Check changed from Info Needed to Dev Needed

emmapeel wrote:

So, I have tested several times each with a Thinkpadx201 and:

- Tails 2.6:
  When clicking restart, the message about ‘now you can take your USB out’ appears, and after approx. 15 seconds of it, the laptop restarts (no stars! :S)
- tails-i386-devel-2.7-20161005T0955Z-bc6a783
  No change with relation to 2.6
- Tails 2.5:
  Similar to Tails 2.6, takes the same amount of time, but I get the screen full as asterisks before restarting, at some point inside the 15 seconds window (around 10 secs)

So this branch does seem to fix it in Jenkins, and does not bring real trouble on bare metal it seems. Sounds like a bugfix that could be merged for 2.7, specially since #11818 seems to fix the main bug that prevented it to be merged for 2.6. Hopefully it won't bring others.

I'll give it a try. Hope I'll find an easy way to include this kernel.
bertagaz wrote:

I'll give it a try. Hope I'll find an easy way to include this kernel.

Any progress update / ETA? Do you need help?

#35 - 10/29/2016 01:24 PM - bertagaz

intrigeri wrote:

Any progress update / ETA? Do you need help?

Progress yes, more tests to analyze, dead line en of the week end best (specially now that merging the other branches just went out of my next moves).

#36 - 10/29/2016 02:05 PM - intrigeri

Progress yes, more tests to analyze,

What's the branch? I see none with "11786" in its name.

#37 - 11/02/2016 07:55 AM - intrigeri

[Moved to #11885]

#38 - 11/02/2016 08:46 AM - intrigeri

I guess that this might be relevant and is worth having a closer look at:

linux-latest (75) unstable; urgency=medium

* From Linux 4.7, the iptables connection tracking system will no longer automatically load helper modules. If your firewall configuration depends on connection tracking helpers, you should explicitly load the required modules. For more information, see <https://home.regit.org/netfilter-en/secure-use-of-helpers/>.

-- Ben Hutchings <ben@decadent.org.uk> Sat, 29 Oct 2016 01:53:18 +0100
#39 - 11/02/2016 01:40 PM - anonym
- Blocks Bug #10776: Step “I shutdown and wait for Tails to finish wiping the memory” fails when memory wiping causes a freeze added

#40 - 11/06/2016 10:46 AM - intrigeri
Will this be fixed by the branch for #11885?

#41 - 11/06/2016 12:41 PM - bertagaz
intrigeri wrote:

Will this be fixed by the branch for #11885?

I'm not sure yet, I've seen failures when trying at home to run a 4.7 kernel, but that was with the 4.7.2. I think this ticket needs to wait a bit more to see how the newer 4.7.8 one works regarding this. I also want to test it a bit on bare metal.

#42 - 11/08/2016 08:21 PM - bertagaz
I've removed the outdated related branch that existed. Linux 4.7 is now in all main branches, so we'll track this bug though their respective Jenkins runs. We can create a more meaningful and active branch for this ticket.

#43 - 11/13/2016 08:59 AM - intrigeri

#44 - 11/16/2016 10:17 AM - intrigeri
And a user just reported that he still sees the bug on Tails 2.7.

#45 - 11/17/2016 05:37 PM - bertagaz
- Target version changed from Tails_2.7 to Tails_2.9.1

#46 - 11/27/2016 09:54 PM - sonicsnail
I tested again with Tails 3.0~alpha1, which is on Linux 4.8.0. Now the machine successfully powers off, whereas on 2.6~rc1 through 2.7 it would hang. I presume the memory is successfully wiping too because there's a delay between “You can now remove...” and powering off, but no asterisks are shown.

#47 - 12/14/2016 08:11 PM - anonym
- Target version changed from Tails_2.9.1 to Tails 2.10

#48 - 12/18/2016 09:55 AM - intrigeri
- Assignee changed from bertagaz to anonym

anonym, can you take this one? It was on bertagaz plate since he was the RM but that's over now, and I think this is now on the Foundations Team's plate.
FWIW the workaround added in our automated test suite (cd66f0b) doesn't work on current feature/stretch apparently: all I see during memory wipe is a black screen (at least in features/usb_install.feature), so if there's a kernel panic we don't see the busybox error message. This blocks updating parts of the test suite for Stretch, e.g. large parts of the USB and persistence tests. Bare metal is not affected (at least one one MacBook Pro I tried this on).

I'm currently testing a workaround to this problem, that should apply to Tails 2.x and 3.x if it works.

intrigeri wrote:

FWIW the workaround added in our automated test suite (cd66f0b) doesn't work on current feature/stretch apparently: all I see during memory wipe is a black screen (at least in features/usb_install.feature), so if there's a kernel panic we don't see the busybox error message.

Err, no: it's "just" that it takes a long while before the message about memory erasure is displayed; I'll file a separate ticket to track this, and meanwhile I'll teach the test suite to be more patient.

Related to Bug #12061: Long delay before memory wipe starts on Stretch added

- Assignee changed from anonym to intrigeri

anonym, can you take this one? It was on bertagaz plate since he was the RM but that's over now, and I think this is now on the Foundations Team's plate.

You seem to be busy with dogtail optimizations, and this problem makes the feature/stretch sprint more painful than it could be, so I worked on it a bit. Here's what I did:

- wrt. the long delay before starting memory wipe: tracked by #12061, workaround'ed on feature/stretch (851f3e7a28910f913e7169a7acc879586afeb5a and 40332999865a1a15f3a083c47b39a9b7fd518)
- wrt. what the "can't fork" problem comes from:
  - made the memory wipe script more verbose (be9b190da9911e0fd1a733e18c44b159136291e) on all base branches; I want more data!
  - created a bugfix/11786-retry-shutdown branch that retries the poweroff/reboot operation a number of times.
  - created a feature/stretch+11786-retry-shutdown branch that integrates bugfix/11786-retry-shutdown into feature/stretch and removes the workaround we had added in cd66f0b49a4773489db3a1bf15294255d681ecd for this problem (not sure why the workaround is not on stable and devel, BTW: anyway).

Next steps: trigger enough builds+tests of these branches on Jenkins to get meaningful data, and test locally to see what happens with these branches merged in (I'll merge them for all my feature/stretch local dev builds in the next days).
intrigeri wrote:

- wrt. what the "can't fork" problem comes from:
  - made the memory wipe script more verbose ([be9b18da8681c6fd1a73de18dc44b15813f6281e](be9b18da8681c6fd1a73de18dc44b15813f6281e)) on all base branches; I want more data!

So it seems that memory wiping succeeds, but then attempting to run poweroff triggers the "can't fork", which kills the script. I'm trying something else on feature/stretch+11786-wait-before-poweroff and bugfix/11786-wait-before-poweroff.

intrigeri wrote:

I'm trying something else on feature/stretch+11786-wait-before-poweroff and bugfix/11786-wait-before-poweroff.

Crap, even calling sleep triggers "can't fork": with busybox, even builtins are copies of the original busybox multi-call binary, so calling "sleep" is not an option; and "sleep" is not a builtin in dash nor bash anyway.

intrigeri wrote:

- Assignee changed from intrigeri to anonym

I've tried everything I could, and failed. Help!

- Assignee changed from anonym to intrigeri

Just to check you hypothesis that sleep would help us, try busy waiting with something like this instead:

```
X=0
until [ "$X" -gt 1000000 ]; do X="$(($X + 1))"; done
```

The number might need adjustment depending on your hardware, of course. On my hardware that results in a ~10 "sleep".
anonym wrote:

Just to check your hypothesis that sleep would help us, try busy waiting with something like this instead:

This won't work either, since in busybox [ requires forking.

One option (until we can rely on PaX' memory sanitization or something) proposed by anonym would be to rewrite that section of the script in C and statically compiling it. It doesn't seem to be worth the effort though: this regression doesn't affect everyone (e.g. even without this new bug, memory wiping fails on lots of hardware), and it's not a security problem (memory is wiped). So for the test suite, we'll rely on our workarounds; and we give up for now on actual bug for human users. Let's hope we'll be able to replace our current implementation with something more robust some day.

Patches welcome though: the script lives in config/chroot_local-includes/usr/share/initramfs-tools/scripts/init-top/memory_wipe.

intrigeri wrote:

One option (until we can rely on PaX' memory sanitization or something) proposed by anonym would be to rewrite that section of the script in C and statically compiling it.

C, or anything else that can be statically compiled, e.g. Go or Rust.

intrigeri wrote:

One option (until we can rely on PaX' memory sanitization or something) proposed by anonym would be to rewrite that section of the script in C and statically compiling it.

C, or anything else that can be statically compiled, e.g. Go or Rust.

Related to Bug #12354: Fix shutdown and memory wipe regressions on 3.0-betaN added

Status changed from In Progress to Duplicate
Let's track the next steps on #12354.

#63 - 04/05/2017 03:53 PM - intrigeri
- Related to deleted (Bug #12354: Fix shutdown and memory wipe regressions on 3.0-betaN)

#64 - 04/05/2017 03:53 PM - intrigeri
- Duplicates Bug #12354: Fix shutdown and memory wipe regressions on 3.0-betaN added

Files
| Screenshot.png | 602 KB | 09/24/2016 | mercedes508 |