In Tails 3.0, enigmail does not seem to be able to connect to the configured HKPS SKS keyserver pool. Details are in #11905#note-5.

**Related issues:**
- Related to Tails - Feature #11905: Consider dropping custom keyserver setting... Resolved 11/11/2016
- Blocked by Tails - Bug #12202: GnuPG can't talk to keyservers on Stretch Resolved 01/31/2017

**Associated revisions**

**Revision 638f8be8 - 01/31/2017 05:46 PM - intrigeri**
Patch Torbirdy to allow not breaking keyserver communication when using GnuPG v2.1+, and to use a better default keyserver.

These patches were adjusted to apply on top of Debian's patched Torbirdy. It would be awesome if the Debian delta could be turned from code patches into configuration some day.

refs: #11948

**Revision 726f9bd7 - 01/31/2017 06:21 PM - intrigeri**
Torbirdy: enable extensions.enigmail.already_torified (refs: #11948).

**Revision e9551434 - 01/31/2017 06:21 PM - intrigeri**
Torbirdy: drop our custom keyserver configuration.

I've imported patches, that I'll submit upstream shortly, that change the Torbirdy's default keyserver to be the one we want.

refs: #11948

**Revision 39660a16 - 02/01/2017 09:36 AM - intrigeri**
Test suite: adjust CONFIGURED_KEYSERVER_HOSTNAME to match current settings.

refs: #11948
History

#1 - 11/17/2016 04:28 PM - bertagaz
- Related to Feature #11905: Consider dropping custom keyserver settings on Stretch added

#2 - 11/17/2016 05:00 PM - bertagaz
Seems that the master branch of enigmail contains patches about Tor and dirmngr. They are quite old, but not released it seems, not sure when they will. We may want to ping upstream about that.

#3 - 11/17/2016 08:31 PM - bertagaz
Two settings in ~/icedove/profile.default/pref.js might be in play:

- extensions.enigmail.agentAdditionalParam: has 'http-proxy' keyserver option set, which is certainly bad and should be removed.
- extensions.enigmail.keyserver: contains the HKPS sks pool keyserver URI.

#4 - 11/18/2016 09:16 AM - bertagaz
- Assignee set to bertagaz

Assigning to me, I'll work on this one.

#5 - 11/18/2016 09:21 AM - intrigeri
- Affected tool set to Email Client

#6 - 11/18/2016 10:12 AM - bertagaz
- Status changed from Confirmed to In Progress
- % Done changed from 0 to 10
- Type of work changed from Research to Code

bertagaz wrote:

- extensions.enigmail.agentAdditionalParam: has 'http-proxy' keyserver option set, which is certainly bad and should be removed.

That's definitely the source of this problem. By removing the --keyserver-options --http-proxy=socks5h://127.0.0.1:9050 in Icedove prefs.js, enigmail can download keys from the keyserver.

#7 - 11/18/2016 10:17 AM - intrigeri

That's definitely the source of this problem. By removing the --keyserver-options --http-proxy=socks5h://127.0.0.1:9050 in Icedove prefs.js, enigmail can download keys from the keyserver.

It'll be awesome if we can drop all duplicated settings and have their canonical version in one single place.

If you need to edit stuff in /etc/skel/, please add a note somewhere so that we think about documenting the adjustments persistent users will need to
intrigeri wrote:

It'll be awesome if we can drop all duplicated settings and have their canonical version in one single place.

Then the branch I pushed does that. Enigmail key fetching/publishing works with this little line added, but for Tails install without existing Icedove persistence only.

If you need to edit stuff in /etc/skel/, please add a note somewhere so that we think about documenting the adjustments persistent users will need to do.

That's not really in this directory, but users that have persistence will still have to edit a setting themselves to modify this option and remove the wrong keyserver proxy setting.

Here's a way they can do it:

Click on the Icedove menu -> Enigmail -> Preferences. Click on the "Display Expert Settings and Menu" button, then on the "Advanced" tab that appeared and remove the "--keyserver-options --http-proxy=socks5h://127.0.0.1:9050" option in the "Additional parameters for GnuPG" text area.

If we're motivated, we could add a function in our /usr/bin/icedove wrapper that modify this setting if an existing persistence for Icedove is detected.

bertagaz wrote:

Here's a way they can do it:

Click on the Icedove menu -> Enigmail -> Preferences. Click on the "Display Expert Settings and Menu" button, then on the "Advanced" tab that appeared and remove the "--keyserver-options --http-proxy=socks5h://127.0.0.1:9050" option in the "Additional parameters for GnuPG" text area.
As discussed elsewhere yesterday, please add this to a (new) news/version_3.0.mdwn page. While you're at it, please also include in there the upgrading instructions for ~/.gnupg/, that I drafted myself in the 3.0~alpha1 announcement.

If we're motivated, we could add a function in our /usr/bin/icedove wrapper that modify this setting if an existing persistence for Icedove is detected.

I say let's not bother for now, and instead 1. add manual instructions for a while; 2. create a ticket to look into automating the parts that have the best cost/benefit ratio, closer to the final release. OK? If you agree, please file that ticket :)

Now, regarding the change brought by this branch: am I guessing correctly that we're effectively forking the default value of extensions.torbirdy.custom.extensions.enigmail.agentAdditionalParam, to remove one parameter from it? If yes, then: where does the default version of this setting live? When that default value changes in the future, how will we think about updating our forked version as well? All this boils down to: maybe we should dynamically extract the default value at build time, and from that compute our modified version and write it to /etc/xul-ext/torbirdy.js?

Also, if I got the code correctly, Enigmail will add the http-proxy= keyserver option if proxyHost is set. It looks like if EnigmailPrefs.getPref("respectHttpProxy") is false, then proxyHost will be false as well, and then Enigmail won't try to use a HTTP proxy. I've tested nothing of this, but it looks to me like we could "just" set Enigmail's respectHttpProxy pref to false, and not have to fiddle with agentAdditionalParam at all, which incidentally solves the problem I've raised above. I don't know how it affects upgrading. What do you think?

---

#10 - 12/22/2016 07:13 PM - intrigeri
intrigeri wrote:

bertagaz wrote:

Here's a way they can do it: […]

As discussed elsewhere yesterday, please add this to a (new) news/version_3.0.mdwn page. While you're at it, please also include in there the upgrading instructions for ~/.gnupg/, that I drafted myself in the 3.0~alpha1 announcement.

This is left to do. I might give it a try later today or tomorrow since that's a piece of functionality I'd like to see working in 3.0~beta1; if I fail to, then I'll give this back to bertagaz :)

If we're motivated, we could add a function in our /usr/bin/icedove wrapper that modify this setting if an existing persistence for Icedove is detected.

I say let's not bother for now, and instead 1. add manual instructions for a while; 2. create a ticket to look into automating the parts that have the best cost/benefit ratio, closer to the final release. OK? If you agree, please file that ticket :)

05/15/2020
Will do. But really, this seems to be a problem in Torbirdy upstream itself: it enforces GnuPG settings that can only work with GnuPG 1.x. Sadly, it's pretty hard to enforce torification of GnuPG 2.x network operations without doing it for the user's dirmngr as a whole (been there, done that for parcimonie), so I've no clue how they'll handle this. So looking at it again today with a fresh mind, it seems that on the short term, Torbirdy should have a "I'm using GnuPG 2.x, I have configured it to use Tor, and I know what I'm doing" hidden option, that we could enable in Tails.

Now, regarding the change brought by this branch: am I guessing correctly that we're effectively forking the default value of extensions.torbirdy.custom.extensions.enigmail.agentAdditionalParam, to remove one parameter from it? If yes, then: where does the default version of this setting live? When that default value changes in the future, how will we think about updating our forked version as well? All this boils down to: maybe we should dynamically extract the default value at build time, and from that compute our modified version and write it to /etc/xul-ext/torbirdy.js?

The problem is very real, but let's not bother with this solution: let's instead improve Torbirdy for us and everyone else at once, as suggested above.

Also, if I got the code correctly, Enigmail will add the http-proxy= keyserver option if proxyHost is set. It looks like if EnigmailPrefs.getPref("respectHttpProxy") is false, then proxyHost will be false as well, and then Enigmail won't try to use a HTTP proxy. I've tested nothing of this, but it looks to me like we could "just" set Enigmail's respectHttpProxy pref to false, and not have to fiddle with agentAdditionalParam at all, which incidentally solves the problem I've raised above. I don't know how it affects upgrading. What do you think?

I tried that and it didn't work.

The closest related ticket I could find is: https://trac.torproject.org/projects/tor/ticket/19971
I'd like to see this regression fixed in 3.0~beta1, so taking over.

- Blocked by Bug #12202: GnuPG can't talk to keyservers on Stretch added

intrigeri wrote:

bertagaz wrote:

Here's a way they can do it:

Click on the Icedove menu -> Enigmail -> Preferences. Click on the "Display Expert Settings and Menu" button, then on the "Advanced" tab that appeared and remove the "--keyserver-options --http-proxy=socks5h://127.0.0.1:9050" option in the "Additional parameters for GnuPG" text area.

As discussed elsewhere yesterday, please add this to a (new) news/version_3.0.mdwn page. While you're at it, please also include in there the upgrading instructions for ~/.gnupg/, that I drafted myself in the 3.0~alpha1 announcement.

Actually, that's not needed apparently: any such change made in Enigmail prefs is overridden by Torbirdy on next Icedove startup, so patching /etc/xul-ext/torbirdy.js as the branch does would be enough.

Seems that the master branch of enigmail contains patches about Tor and dirmngr. They are quite old, but not released it seems, not sure when they will. We may want to ping upstream about that.

Indeed, Enigmail master branch has code to use Tor for keyserver operations, and it seems to support both GnuPG 1 and 2. Once that's released and ready for production, some parts of the Torbirdy code that are about Enigmail can go away, which is good since its doing such a bad job at it. But that doesn't seem to be for their 1.9 series, so we can't rely on it for 3.0.
So, after re-reading this ticket and testing / looking into various things as reported above, we have several options:

- just merge the proposed branch
  - pros: it's free on the short term
  - cons: our overriding of Torbirdy defaults can bite us in the future, in ways that are hard to notice
- improve the proposed branch somehow, to increase the chances when notice when Torbirdy defaults change
  - pros: safe vs. Torbirdy defaults change
  - cons: requires Tails-specific work that will become useless some day
- patch Torbirdy locally to prevent it from setting options that break our use case
  - pros: it's cheap
  - pros: safe vs. Torbirdy defaults change
  - pros: it might even be that our patch would be accepted upstream, if it only affects Tails (look for vendor.name in Torbirdy source history)
  - cons: will probably require rebasing the patch on some Torbirdy upgrades, until a proper solution is implemented upstream
- improve Torbirdy upstream so it nicely supports our use case:
  - pros: safe vs. Torbirdy defaults change
  - cons: lots of work to detect GnuPG 2 + already torified dirmngr, that will be wasted once Enigmail does it itself

At this point I prefer the third option by far. I'll implement it and propose it to Torbirdy upstream, raising my concerns about investing time in a generic solution vs. the future of Enigmail.

---

#18 - 01/31/2017 06:28 PM - intrigeri

- % Done changed from 50 to 60
- Feature Branch deleted (bugfix/11948-remove-keyserver-proxy-option-from-enigmail)

Patches submitted on https://trac.torproject.org/projects/tor/ticket/19971 and applied on feature/stretch, so bugfix/11948-remove-keyserver-proxy-option-from-enigmail won't be needed. Let's see what Jenkins thinks.

#19 - 02/01/2017 10:06 AM - intrigeri

- Status changed from In Progress to Resolved
- % Done changed from 60 to 100
- QA Check deleted (Dev Needed)

intrigeri wrote:

Let's see what Jenkins thinks.

Actually, we have no automated test for that. Manual testing passes though.