

Packages - Freedom Issue #1035

[your-system-sanity]: Non-Free Software From Third-party Package Managers (TPPM)

2016-06-19 08:03 PM - jxself

Status:	in progress	% Done:	23%
Priority:	feature		
Assignee:	freemor		
Category:			
Description			
From a conversation on the FSDG mailing list:			
http://lists.nongnu.org/archive/html/gnu-linux-libre/2016-04/msg00070.html			
http://lists.nongnu.org/archive/html/gnu-linux-libre/2016-04/msg00116.html			
A lot of programming languages have own packages managers: npm (CSS/JavaScript), Bower (Web), pip (Python), RubyGems (Ruby), CPAN (Perl), Cargo (Rust), many more. These things would qualify as "repositories" under the Free System Distribution Guidelines; and most do not limit themselves to only including free software. Until/unless Stallman's ideas of either convincing them to only include free software or develop a free replacement come along, I propose disabling such things in Parabola.			
UPDATE 2023-06:			
See this wiki page for repo evaluations, solution proposals, and the current state of progress of each.			
https://wiki.parabola.nu/TPPM_Liberation_Project			
UPDATE 2023-06:			
there is an on-going discussion on the FSDG mailing list - GNU is currently working on a solution for cargo - RMS is involved with the FSDG mailing list discussion and is suggesting that we move forward with this project - he has (amazed me) suggesting to stay with the weakest proposal: "do nothing - continue ignoring it" as we have been, at least until each can be addressed properly			
bill-auger and gnutoo still would rather exclude them all ASAP, pending a better solution - bill-auger still believes that the ideal solution above all, is to exclude them permanently and accept packaging requests for desirable language-specific library packages			
Subtasks:			
Freedom Issue # 1755: [asp] Links to Arch source repositories instead of Parabola ones			fixed
Freedom Issue # 3003: [lxd]: has dubious containers repositories (It can download and i...			confirmed
Freedom Issue # 3412: Evaluate the libreoffice extensions repository			unconfirmed
Freedom Issue # 3422: Docker configured to use a nonfree repository			confirmed
Bug # 3443: [python-setuputils]: build depends on blacklisted 'python-pip' - multiple d...			confirmed
Bug # 3338: [freecad]: depends on blacklisted 'qt5-webengine' and blacklisted 'python-pip'			fixed
Bug # 3432: cpupower: symbol lookup error: cpupower: undefined symbol: pci_filter_match...			confirmed
Bug # 3442: gajim-plugin-omemo: broken on update			fixed
Feature Request # 3510: why is python-pip no longer available?			open
Bug # 3581: [nodejs-bower] TPPM			confirmed
Related issues:			
Related to Packages - Packaging Request #1413: [partial fixes] Package deblob...			in progress
Related to Packages - Freedom Issue #2304: [wesnoth] Default add-ons server a...			info needed
Related to Packages - Freedom Issue #2260: [openttd] recommends OpenSFX, whic...			info needed
Related to libretools - Feature Request #2403: [libremakepkg]: eradicate mkso...			info needed
Related to Packages - Freedom Issue #123: wanted: blacklist for nonfree aur p...			not-a-bug
Related to Packages - Bug #1904: third-party package managers should not inst...			open
Related to Packages - Freedom Issue #2601: [ruby][rubygems][your-system-sanit...			fixed
Related to Packages - Feature Request #2353: [your-system sanity][python2-pip...			open
Related to Packages - Feature Request #2355: [your-system sanity][0ad] not d...			open
Related to Packages - Feature Request #2356: [your-system sanity][supertux2] ...			open
Related to Packages - Feature Request #2357: [your-system sanity][supertuxkar...			open
Related to Packages - Feature Request #2358: [your-system sanity][freeciv] no...			open

Related to Packages - Feature Request #2359: [your-system sanity][teeworlds] ...	open
Related to Packages - Feature Request #2360: [your-system sanity][xonotic-*] ...	open
Related to Packages - Feature Request #2361: [your-system sanity][minetest] D...	open
Related to Packages - Feature Request #2480: [your system sanity] Make it con...	in progress
Related to Packages - Freedom Issue #194: makepkg license check	not-a-bug
Related to Packages - Freedom Issue #2909: fwupd has a nonfree repository Was...	fixed
Related to Packages - Freedom Issue #2503: [kodi]: contains nonfree RAR and n...	confirmed
Related to Packages - Freedom Issue #3005: [geoipupdate][sn0int]: TPM for pr...	unconfirmed
Related to Packages - Freedom Issue #3000: [fwupd]: is an auto-updater , may ...	fixed
Related to Packages - Bug #3395: [ruby] Requires rubygems, which conflicts wi...	fixed

History

#1 - 2018-02-09 09:22 AM - asie

I think a better interim solution could be to patch these repository managers so that they only accept free software. However, this does not solve any privacy issues which those packages may cause, as those are not usually documented.

This shouldn't be too hard for some of them; npm for example abides by the SPDX License List standard (<https://spdx.org/licenses/>). Most of them seem to include licensing information in their packages, though some may not have a standardized way of denoting a specific license.

#2 - 2018-03-17 08:30 PM - bill-auger

also PECL for PHP, cabal-install for haskell, and no doubt various others

blacklisting them would be the "interim solution" - it would be a far larger set of tasks to determine how much work it would be to patch them one-by-one and then to craft and apply the patches

i will add that the FSDG actually prohibits any reference to "third-party repositories that are not committed to only including free software; even if they only have free software today" - so even filtering package according to their licenses may not be fully FSDG-compliant - and bear in mind that the license metadata in these repos is both voluntary and un-verified

as i understand, that is the main reason why parabola does not offer any tools that access the AUR repos - these language-specific repos are much of that same un-supervised, unrestrained, free-for-all sort of dumping ground for literally **whatever**

#3 - 2018-03-17 09:06 PM - bill-auger

i did a shallow survey of the guidelines for several such repos - just to determine the language they use (or neglect) to encourage proper licensing

PERL:

the PERL CPAN and repo guidelines has strong language such as "required" regarding license notices but seems to allow any license

PHP:

PHP PECL had a similarly strong requirement but with notable prescriptions regarding particular licenses; highly suggesting the 'PHP' license and even vaguely prohibiting "wrappers for GPL libraries" while allowing "Wrappers for libraries with license fees or closed sources libraries"

Note: wrappers for GPL (all versions) or LGPLv3 libraries will not be accepted.
Wrappers for libraries licensed under LGPLv2 are however allowed while being discouraged.

python and ruby:

the language of python pip and ruby rubygems repo guidelines is significantly weaker saying only <https://python.org/???>:

"every package should include a license file"

<https://packaging.python.org/en/latest/tutorials/packaging-projects/>:

You should always include ..., which license your package is available under, and ...
It's important for every package uploaded to the Python Package Index to include a license. This tells users who install your package the terms under which they can use your package. For help picking a license, see <https://choosealicense.com/>.

<https://rubygems.org/???>:

"you should specify a license"

rust:

although the rust cargo metadata file has a key for denoting the license; i could find no place on the web that actually recommends that anyone uses it

haskell:

the haskell cabal repo guidelines has the most libre standard of those i looked at, saying:

The code and other material you upload and distribute via this site must be under an open source license

it then includes links to the OSI and FSF for "information about free and open source software licenses" - so perhaps 'cabal-install' is the only of this class that could be retained without conflict

for any the others, which do not have such clear and stringent policies as cabal, it would need to be determined, whether it would be theoretically possible (with or without modifications to the package manager) to discern the license of each package before presenting it to the user - it so, then that could be a relatively simple solution, compared to the task of creating and maintaining filtered repos

TODO:

- flatpack
- appimage
- docker
- nuget
- npm
- guix (free culture concerns?)
- asp
- kodi
- are there more?

#4 - 2018-04-26 12:14 AM - bill-auger

- *Subject changed from Non-Free Software From Programming Language Package Managers to Non-Free Software From Third-party Package Managers*

just to add - this applies not only to language-specific package managers but as well to all third-party package managers such as flatpack, appimage, docker and whatever they cook up next

#5 - 2018-04-26 12:31 AM - bill-auger

- *Related to Packaging Request #1413: [partial fixes] Package deblobed version of gnome-software. Was: [gnome-software][discover] offers non-free software & shows incorrect licenses added*

#6 - 2018-07-18 11:08 PM - bill-auger

let us not forget 'nuget'

#7 - 2018-11-29 10:08 PM - bill-auger

i was just reminded that this "elephant in the room" applies even to guix at the intersection of "free software"/"free culture" (i.e. 'your-freedom' ^ 'your-artistic-freedom')

#8 - 2018-11-30 01:26 PM - freemor

Since we package our own pacman perhaps we could create a list of the third party package managers for it. So that if it sees a request to install those it'll throw warnings about:

```
You are installing a third party package manager!  
By installing it you assume all responsibility  
for system instability it may cause.  
Also as doing so bypasses the Parabola blacklist.  
You will have to be responsible for checking the  
freedom issues of any package you choose to install  
with this package manager in the future
```

or something to that effect.

#9 - 2018-11-30 01:44 PM - freemor

From where I sit using a Third party package manager is analogous to using AUR. We can't plug all these holes without creating a frustratingly restricted system. The end user needs to take some responsibility for what they install from non parabola repo places.

Yes, I do see the issue that the TPPMs (got tired of typing it out over and over) are in the Parabola repo. And could be argued as "leading people to non free...". But blacklisting the TPPMs is too restrictive and trying to patch them and then police all the packages that are improperly licensed, etc. would add a huge maintenance load.

Thus I think, Warm strongly up front about the issues possibly with a prompt requiring the user to type "Yes I understand that this may damage my systems stability and that freedom issues are my responsibility" well maybe not that long, but long enough they they have to think about what they are doing rather than just "Y <enter>" because they want to get on with what they were doing.

#10 - 2018-11-30 06:33 PM - freemor

Just a quick note:

[eschwartz](#) suggested alpm-hooks as an easy route to implement such warnings.

(mostly here so it doesn't slip from my mind if I choose to revisit this)

#11 - 2018-12-06 10:26 AM - bill-auger

freemor wrote:

So that if it sees a request to install those it'll throw warnings about:

instead of a warning how about putting them all in a new non-default repo - with your admonishments on a new wiki page describing how to enable the repo

also, the CLI prompt is probably suppressed by pacman GUI front-ends - we could patch octopi but thats not the only one

freemor wrote:

From where I sit using a Third party package manager is analogous to using AUR.

so should we put AUR GUIs in the new repo with the same warning? - that would seem to be one step in the opposite direction of this issue - it is surely not an omission that there are no AUR helpers in parabola - i think it is clear that AUR helpers, container fetchers, and other package managers should be treated as equivalent - (with the haskell PM being the notable exception above)

freemor wrote:

blacklisting the TPPMs is too restrictive and trying to patch them

i think a small number of these specify license in metadata - in theory, someone could write a filter - that would be the ideal "rescue" option - still this is several such non-trivial patches against several unfamiliar projects

#12 - 2018-12-06 01:12 PM - freemor

Excellent point about GUI PMs. Living on the CLI I often forget about the GUIs

Separate Repo is an interesting idea. And yes I'd vote to fire AUR helpers in there too. One could easily argue that having a point and click AUR package manager is "helping people install non-free..."

I do understand about the metadata in some TPPMs. My worry is that not only would the be the patching/maintaining burden that you mentioned to filter, but also a whole layer of then having to police those collections for projects with improper licensing data in their metadata.

#13 - 2019-02-09 06:40 AM - bill-auger

i was just re-reading the FSDG today and it struck me as being quite clear on this issue:

```
Nor should the distribution refer to third-party repositories that are not committed to only including free software; even if they only have free software today, that may not be true tomorrow. Programs in the system should not suggest installing nonfree plugins, documentation, and so on.
```

so again, from my shallow research so far, it looks that the haskell cabal is the only one that we could consider to be "committed to only including free software"

#14 - 2019-02-09 07:11 AM - bill-auger

and FWIW, i also read the original discussion on the FSDG mailing list that occurred about a month before this issue was open, in which RMS suggested that if the repositories themselves could not be convinced to distribute free software only, then the best solution would be to extract the licensing information from each of these package manager's repo indices, if possible, and import only the free packages into a new separate repo for each package manager - that, he also suggested could be hosted by the FSF, if volunteers could accomplish the work of automated filtering

<https://lists.nongnu.org/archive/html/gnu-linux-libre/2016-04/msg00116.html>

#15 - 2019-02-27 08:03 PM - bill-auger

- Priority changed from freedom issue to discussion

#16 - 2019-02-27 08:18 PM - bill-auger

- Priority changed from discussion to freedom issue

#17 - 2019-02-27 08:18 PM - bill-auger

- Status changed from open to confirmed

#18 - 2019-04-28 11:08 PM - bill-auger

- Related to Freedom Issue #2304: [wesnoth] Default add-ons server allows non-free add-ons added

#19 - 2019-04-28 11:10 PM - bill-auger

- Related to Freedom Issue #2260: [openttd] recommends OpenSFX, which is known to be under a non-FSDG license added

#20 - 2019-07-21 04:15 PM - bill-auger

- Priority changed from freedom issue to feature

- Subject changed from Non-Free Software From Third-party Package Managers to [your-system-sanity]: Non-Free Software From Third-party Package Managers

freemor has been working on an idea tentatively named 'your-system-sanity' to present a more blatant warning about the software that TPPMs index and assist downloading, and the havoc they (especially pip) can wreak on the system, and to identify and isolate them all from the main repos, possibly into new repo like: [unsupported]

though its perhaps not as ideal of a solution that curated repos shared by other distros would be; it is better than the current situation

#21 - 2019-07-21 04:15 PM - bill-auger

- Assignee set to freemor

- Status changed from confirmed to in progress

#22 - 2019-08-01 02:25 AM - bill-auger

- Related to Feature Request #2403: [libremakepkg]: eradicate mksource() from abslibre added

#23 - 2020-01-14 03:00 PM - bill-auger

- Related to Freedom Issue #123: wanted: blacklist for nonfree aur packages added

#24 - 2020-01-14 03:01 PM - bill-auger

- Related to Bug #1904: third-party package managers should not install into /usr/bin added

#25 - 2020-01-14 03:03 PM - bill-auger

- Related to Freedom Issue #2601: [ruby][rubygems][your-system-sanity] Dependencies added

#26 - 2020-01-14 03:03 PM - bill-auger

- Related to Feature Request #2353: [your-system sanity][python2-pip] not detected added

#27 - 2020-01-14 03:03 PM - bill-auger

- Related to Feature Request #2355: [your-system sanity][0ad] not detected added

#28 - 2020-01-14 03:04 PM - bill-auger

- Related to Feature Request #2356: [your-system sanity][supertux2] not detected added

#29 - 2020-01-14 03:04 PM - bill-auger

- Related to Feature Request #2357: [your-system sanity][supertuxkart] not detected added

#30 - 2020-01-14 03:04 PM - bill-auger

- Related to Feature Request #2358: [your-system sanity][freeciv] not detected added

#31 - 2020-01-14 03:04 PM - bill-auger

- Related to Feature Request #2359: [your-system sanity][teeworlds] Downloads packages from server added

#32 - 2020-01-14 03:04 PM - bill-auger

- Related to Feature Request #2360: [your-system sanity][xonotic-*] Downloads packages from server added

#33 - 2020-01-14 03:04 PM - bill-auger

- Related to Feature Request #2361: [your-system sanity][minetest] Downloads packages from server, third-party package manager added

#34 - 2020-01-14 03:05 PM - bill-auger

- Related to Feature Request #2480: [your system sanity] Make it conflict like your-freedom added

#35 - 2020-04-02 01:32 AM - bill-auger

it just occurred to me that 'asp' would be in this category

#36 - 2020-07-10 07:03 AM - bill-auger

- Related to Freedom Issue #194: makepkg license check added

#37 - 2020-10-11 09:44 AM - bill-auger

- Related to Freedom Issue #2909: fwupd has a nonfree repository Was: Do check fwupd added

#38 - 2020-12-04 09:44 PM - grizzlyuser

What about patching out only the strings in package managers that refer to the domains hosting problematic repositories? PMs themselves can be free software, the problem is just those references to the repos.

It's quite likely they are already made configurable in some of the PMs, like Maven. For those projects lacking repo configuration, a simple one can be requested or even written (and it's a good thing to submit upstream anyway).

If for some given PM there's a public repo that aligns well with the FSDG, then replace upstream repo with that one. If not, then just leave the repo configuration empty and up to the user. Because in many cases, the repo can be, for example, hosted privately by a company, or even locally by a user.

As mentioned before, alpm-hooks can be used to inform users about the necessary configuration.

#39 - 2020-12-05 12:27 AM - bill-auger

that is essentially the solution proposed by RMS (comment # 14) - to have GNU host curated repos for each of these - the task of hacking the client, if necessary, is not a big obstacle - the work involved in creating, curating, and maintaining the thousands of packages, for each of about a dozen package managers, is quite significant though - it would be a rather large and demanding project; and would require long-term dedicated volunteers

#40 - 2020-12-05 07:12 AM - grizzlyuser

I meant just hacking the clients for the time being. Because repo hosting / maintenance is a tremendous effort, obviously. It would be great to have those, but unless there's enough interest from the community, those won't exist.

The point is to give a message that those repos are problematic. It's up to the user to choose the repo they want to use. Whoever needs to use the upstream repos, can always find them, just outside of libre distro.

#41 - 2020-12-05 08:21 AM - bill-auger

ok, i understand the proposal better now - to de-configure the clients, so that they are not usable OOTB; and require each user to configure them to a particular server - i think that is a very good solution - even if there is only one such public server available, it would satisfy the FSDG, by not endorsing it or actively leading users toward it - the 'your-system-sanity' proposal does not meet the FSDG nearly as well

i had a similar idea recently, regarding the AUR - the idea is to customize one of the AUR helpers supported by octopi, such that it does not ever index the AUR, nor connect to it in any way, unless configured by the user to do so, on a single package whitelist basis - only after the user has whitelisted a particular PKGBUILD, would the client connect to it's git repo, for fetching the recipe, and notifying of upgrades - because all AUR PKGBUILDS are in a git repo, the configuration could be maximally general, merely accepting a git URL; and the FSDG would be satisfied, because no software sources would be recommended, or referenced OOTB - that would have the additional benefit of compatibility with any git server, such as self-hosted personal or organization repos, github, etc - as long as there is a PKGBUILD in the root dir of the git repo, it would "just work"

it would be relatively simple to accomplish; and simple to use - we would not even need to document how to configure it - the client itself could simply prompt: "Enter a git URL where some

PKGBUILD may be fetched: ___" - another advantage to that proposal, would be that we would not need to package any of the PCR - we could maintain the PCR in the form of recipes only; and the PCR directory of abslibre could be the default (or only) source for the client's search index

#42 - 2021-01-18 09:24 PM - bill-auger

- Related to Freedom Issue #2503: [kodi]: contains nonfree RAR and nonfree addons added

#43 - 2021-01-18 09:24 PM - bill-auger

i have added 'kodi' to the list #2503 - i added that BR as the blacklist reference about a year ago; because i could find no discussion of it - unfortunately, the kodi team publishes a different set of add-ons for each version, as an indexed repository; which means that someone needs to sort through them upon each new major version release, and re-construct a replacement repo

up until the 'krypton' release, parabola was maintaining a replacement repo for this program - that contained nearly 1000 add-ons - since jan 2019, those 'krypton' add-ons have been obsolete - the current release 'leia' has around 1800 add-ons to sort through; and no one has sorted through them - so currently, no add-ons are available in the parabola 'kodi' - i tried to crudely symlink the existing repo as 'leia'; but it is apparently not working - users will get "Could not connect to repository"

#44 - 2021-01-28 03:32 PM - GNUtoo

I think we could also clarify what users can expect from Parabola at the same time.

I've started to do that here:

https://wiki.parabola.nu/Talk:Main_Page#How_does_Parabola_protects_users_against_nonfree_software

The idea is to make sure users do understand the boundaries of what's handled by Parabola and what's not and point them to other issues that can affect them at the same time to make sure they know about nonfree BIOS and so on (else some users might think that running Parabola is enough to have only free software).

We could also do the same for privacy and security.

Denis.

#45 - 2021-03-23 03:22 PM - telur

some list of TPPM for consideration:

```
pacman -Ss "package manager"
libre/pacman 5.2.2-1.parabolal (base-devel) [instalita] A library-based package manager with dependency sup
port
extra/nuget 5.8.0-1 Package manager for .NET.
community/apm 2.6.1-2 Atom package manager
community/apper 1.0.0-4 An application and package manager using PackageKit
community/bower 1.8.12-1 A package manager for the web
community/dpkg 1.20.5-2 The Debian Package Manager tools
community/dub 1.24.1-1 (dlang) Developer package manager for D programming language
community/fusesoc 1.12.0-1 Package manager and build abstraction tool for FPGA/ASIC development
community/helm 3.5.3-1 The Kubernetes Package Manager
community/nimble 1:0.12.0-1 Package manager for the Nim programming language
community/npm 7.6.3-1 A package manager for javascript
community/ocaml-findlib 1.8.1-8 OCaml package manager
community/opam 2.0.8-1 OCaml package manager
community/rpm-tools 4.16.1.2-1 RPM Package Manager - RPM.org fork, used in major RPM distros
community/shards 0.13.0-1 The package manager for the Crystal language
community/sn0int 0.20.1-1 Semi-automatic OSINT framework and package manager
pcr/guix 1.1.0-2 A purely functional package manager for the GNU system
pcr/nodejs-bower 1.8.2-1 A package manager for the web
pcr/pacman4console 1.3-1 A 9 level ncurses pacman game with editor, patched not to disturb our package mana
ger and to have
nice ghosts
```

also maybe putting some pointer in the end of system sanity TPPM block message would be nice like "for further info and feedback see <https://labs.parabola.nu/issues/1035>"

#46 - 2021-04-03 06:49 AM - bill-auger

- Related to Freedom Issue #3005: [geoiupdate][sn0int]: TPPM for proprietary(?) data added

#47 - 2021-04-03 06:52 AM - bill-auger

re: 'geoiupdate' and 'sn0int' - they may not install anything non-free, and probalby can not corrupt the system; but should probably be investigated

#48 - 2021-04-03 09:16 AM - bill-auger

- Related to Freedom Issue #3000: [fwupd]: is an auto-updater , may also download proprietary firmware added

#49 - 2022-04-07 07:28 PM - oaken-source

(copied verbatim from [#1755](#) because it belongs here as well)

blacklisting a program because it allows the user to choose to install non-free software sounds like a ludicrous proposal to me. pacman itself is capable of doing that.

I understand that TPPMs are a hot topic, and they deserve to be for a host of different reasons, but removing all of them with the goal of re-adding "liberated" versions -- meaning restricted to installing only free software? or only software acceptable under the FSDG? how would we even implement this? -- would cripple parabola to the point where it would no longer be possible (at least for me personally) to do any productive work on it. I use pip a lot in my daily work, albeit almost exclusively contained in virtualenv instances.

Using TPPMs presents a challenge, and contains certain risks. And we as a distribution with the goal of helping spread software freedom are clearly in a position to have to educate our users of these problems. But TPPMs are also a way through which entire ecosystems of free software are distributed through means which are beyond the scope of a distribution.

I would argue that just because a program is a TPPM that is **capable** of installing non-free software doesn't mean that it violates the FSDG as written. A TPPM that is made to **only** access non-free software is a different matter though. But most of the examples listed here are not (and in fact, this issue doesn't even deal with TPPMs, so feel free to copy / refer to this text from any relevant place on this issue tracker).

What I would propose is to include a warning on the most prominent TPPMs that they are capable of installing non-free software, and that users should avoid using them unless they know what they are doing. And not blacklist them, because then we lose users that need to use these programs for their free computing.

#50 - 2022-04-07 07:39 PM - bill-auger

but pacman does not offer/suggest any non-free software to install - it merely "allows" it, but does not "suggest" any - the [Degrees of Non-Free Software Tolerance](#) wiki article tries to make those distinctions concrete - i think we should refer to that as our pseudo policy, and to judge all TPPMs by the same metric, and to refine the definitions as necessary; to pinpoint exactly where is the FSDG expectation, and where parabola aims for

in short, the FSDG language is about suggestions, recommendations, "leading toward" - it is not to say that the program's only purpose is to do that; and it does not say that the program must not "allow" it - it prohibits any program or documentation which "leads the user toward non-free software", or websites which are known to do the same

#51 - 2022-04-07 07:53 PM - bill-auger

the proposal to adapt octopi into a specialized AUR-helper is exemplary of that distinction

a typical AUR helper (or the unmodified upstream octopi) allows searching for and installing non-free software, in its default configuration - for example, the user types "web browser", and google-chrome appears - the user simply clicks on google-chrome, then "install-it", and the program fetches, builds, and install it - that behavior is clearly not FSDG-fit - it is "suggesting" and "leading toward"

in contrast, the specialized proposal would not do any such thing - it would merely allow the user to type (manually) a URL to some arbitrary git repo, where a PKGBUILD may be found - that is clearly within the FSDG; because there is no suggestion or "leading" - it is merely "allowing" the user to install non-free software, but not accidentally nor by suggestion, but only very intentionally and manually - IMHO, that is equivalent to pacman -U, but not pacman -S

pacman -U is merely "allowing", but pacman -Ss is "suggesting"

#52 - 2022-04-07 11:49 PM - gap

To clarify, so we are all on the same page:

A free, general-purpose OS like Parabola is capable of providing you programs which can be used to do general-purpose tasks; that's the whole point. This necessarily includes installing or perhaps even writing, authoring, composing, or otherwise inventing new proprietary works.

The purpose of Parabola is to:

1. protect users from accidentally installing proprietary packages, although this is necessarily overridable as it would be impossible to make Parabola incapable of installing proprietary packages.
Moreover, this would probably backfire, as it would prevent ethical uses of proprietary packages, such as reverse-engineering.
Most importantly of all, trying to prevent the user from doing something would be unethical, because then it would be Digital Restrictions Management (DRM).
2. not push/lead users towards, or suggest proprietary packages, eg. by hosting them in its repos, including install instructions in documentation or the wiki, or otherwise assisting the user in the installation of proprietary packages.
This also includes the behaviour of Parabola hackers (see the last few comments in [#3225](#)), because having people directly talk with people, eg. on the issue trackers, mailing lists, or forum, discussing how to install proprietary software completely undermines all the other purposes of Parabola.

As for the issues of TPPMs, the ideal solution is for each TPPM to have an accompanying 100% libre repo.

By persuading the maintainers of upstream repos to make the upstream repos 100% libre, we would fix the root cause of the issue, instead of trying to clean them up after-the-fact.

This might not always be possible, after all, Parabola is itself an example of a libre fork of a project which does not agree with this, but we should at least try.

I genuinely believe that a lot of these repos, especially the ones full of language-specific development libraries, are more or less already ~95% libre, the maintainers already agree about being libre, and all we have to do is establish it as an official rule so they have a duty to remove packages in violation of those rules, thereby ensuring freedom for their users, not to mention putting freedom-conscious peoples' minds to rest, and taking a workload off the Parabola hackers.

#53 - 2022-04-08 02:09 AM - bill-auger

By persuading the maintainers of upstream repos to make the upstream repos 100% libre, we would fix the root cause of the issue, instead of trying to clean them up after-the-fact.

that is true; but that discussion ran its course 6 years ago, near the time when this ticket was created - you should read this entire ticket to catch up on the context - especially comment # 14 - that is probably the ideal plan; but it has yet to happen in all these years - it appears that hacking the package manager client (the crude "after-the-fact" solution that you dismissed), is the only one which is likely to ever yield a useful result

the best solution of all, is to not use any TPPM; but to add *whatever* to the distro standard repos - that is what guix has done, and also arch (so parabola already has that option too) - most python and haskell programs already will work without using their TPPM; because the most common libraries are packaged by arch - IMHO, i am not so inclined toward any plan to rescue the TPPM - although that, or if GNU would host libre repos, could benefit other distros; its not clear if any others are interested - if parabola is the only FSDG distro which, in practice, will actually benefit from our chosen solution, the wiser plan would be to package more libs for pacman - arch would probably accept them

I genuinely believe that a lot of these repos, ...are more or less already ~95% libre,

i think you are over-estimating greatly - i would guess most of them are about 50% libre (verifiable), and 50% unverifiable - many of those unverifiable packages may be libre, but offer no easy way to know which ones

all we have to do is establish it as an official rule so they have a duty to remove packages in violation of those rules,

that is not for "us" to do - all of those repos have their own guidelines - "they" would need to add strong licensing requirements and enforce them - if those people wanted licensing requirements, they would be in place already - and i dont mean "libre" licensing requirements - most i looked at, had no requirements at all - see comment # 3 - haskell and maybe PERL are the only ones that we could put any confidence in - maybe the situation has improved since then; but without an easy way to know which license some package has, blacklisting the TPPM is the only feasible option

#54 - 2022-04-29 03:43 AM - bill-auger

i found that the arch wiki lists tools to transform TPPM packages into arch packages - so they would only need to be audited - pckaging them should be easy

- Haskell: cblrepo, arch-hs
- Node.js: nodejs-npm2arch
- Python: pipman-git, pip2arch-git, python-pypi2pkgbuild
- Ruby: gem2arch, pacgem
- Rust: cargo-pkgbuild

there is at least one missing - i used one for PERL packages

#55 - 2022-05-04 01:22 AM - bill-auger

- Description updated

- Subject changed from [your-system-sanity]: Non-Free Software From Third-party Package Managers to [your-system-sanity]: Non-Free Software From Third-party Package Managers (TPPM)

#56 - 2022-05-04 01:28 AM - bill-auger

- Description updated

#57 - 2022-05-04 01:39 AM - bill-auger

- Description updated

#58 - 2022-05-04 01:41 AM - bill-auger

- Description updated

#59 - 2022-05-04 01:47 AM - bill-auger

- Description updated

#60 - 2022-05-04 01:56 AM - bill-auger

- Description updated

sry if this was noisy for anyone - this is the final edit for today

#61 - 2022-07-29 02:38 PM - GNUtoo

In the meantime, I've blacklisted python-pip and python2-pip. This makes sure that the discussions will not delay a fix for many more years.

It also enables people to not have to rush to fix it: If there is a long term solution that would require 0 maintenance on Parabola side, we'd just re-add python-pip when the fix is done.

I've also started to compile the information in a Libreplanet wiki page as this issue is common to many FSDG compliant distributions:

<https://libreplanet.org/wiki/Group:Software/research/ExternalRepositories>

edit1: metionned ExternalRepositories

edit2: add link in sentence.

#62 - 2022-07-29 03:43 PM - bill-auger

this small step is irrational though - why single out only pip?

this ticket represents dozens of packages, which all deserve exactly the same treatment (whatever that is)

the standing consensus was to leave them be - i know we discussed blacklisting them and accepting packaging requests instead; but if the new treatment is to blacklist them, then we must blacklist them ALL, if the rationale is to be defended sincerely

and that definitely includes guix - guix indexes and assists installing software which is explicitly forbidden by the FSDG - pip does not - but pip is the one that people are most likely to complain about (they already have complained while the plan was only being discussed)

like most of there FSDG grey-issues, i dont have a strong preference for which treatment is done; but i am extremely uncomfortable to apply a treatment (whatever that is) willy-nilly, to some examples but not to others

so what shall i tell the next user who asks me "why did you blacklist pip, but you kept npm, rubygens, guix, nuget, cpan, pecl, flatpack, appimage, docker, and a dozen others?"

#63 - 2022-07-29 04:21 PM - bill-auger

in case you are reading this via email, i added an important tidbit to my original reply, which i want to make absolutely clear

we must
blacklist them ALL, if the rationale is to be defended sincerely

and:
that definitely includes guix - guix indexes and assists installing software which is explicitly forbidden by the FSDG - pip does not

if we blacklist any of these, we are still left defending the uncomfortable question: "so why does parabola do this in the name of the FSDG, when the other FSDG distro do not?"

i do not want to answer either of those questions ever again - myself, i would not implement anything so inconsistently as to invite such questions - every time i answer such questions politely, i am apologizing (grudgingly) for the actions of other people, actions which i explicitly recommended against

to be clear, if we do nothing, then we are being consistent with the other FSDG distros, so the question will not be asked

if we blacklist all of them, then the answer to the questions is: "because we believe that those other distros are not following

the spirit of FSDG"

if we blacklist only one of them, and that one is not guix, then there is no justifiable answer to give - trisquel removed pip a while back (and only pip), but if a trisquel user questions the decision (and they have), i am not obligated to defend that decision

#64 - 2022-07-29 07:36 PM - gap

guix indexes and assists installing software which is explicitly forbidden by the FSDG

Has this issue been reported?

From your replies, it sounds like we need greater inter-FSDG-compliant-distro cooperation and organisation.

I agree the solution to this ticket needs to be implemented consistently across all FSDG-compliant distros and all TPPMs. However, we have no consensus on what the solution is, other than to leave it lingering because we do not know what to do.

As I stated previously, I think the only proper solution is to fix the root cause of the issue upstream instead of having to duplicate effort many times over by each distro implementing separate fixes. I genuinely believe we can make a solid case and be successful in persuading upstream to change the policies if only we cooperate between distros and form a plan.

Only after this results in failure should we consider the next best option: starting forked repos which import and blacklist packages in the same style as Parabola and Arch.

For the time being, patching out the upstream repo URLs should be a good temporary solution. Blacklisting the TPPMs entirely is too great a loss of features, which devs really need.

Starting our own project of packaging requests is not sustainable or portable; the package recipes for TPPMs cannot use Parabola repos, rather we would have to offer an inferior dev experience on Parabola and force devs to "be their own package manger" and manually check the recipes, and install the corresponding Parabola packages.

Needless to say, I don't rate the other "dubious" solutions.

#65 - 2022-07-29 07:47 PM - gap

I'd also like to add that I do not trust package license metadata whatsoever; they're often mislabeled. From all the packages I've seen before on Arch, the license field seldom contains all the licenses of the files in the package, or even the correct ones, often being ambiguous like "GPL" instead of "GNU-GPL-3.0-or-later" for example. Projects also seldom contain proper licensing documentation such as a Debian copyright file, and files often do not contain copyright headers. As such, filtering by package license metadata would be a mistake; it is imperative that we audit packages properly instead of relying on unreliable information.

IMO the package license metadata field should be replaced with a pointer to the full license documentation in the form of a standard file such as a Debian copyright file, although we can't do this for PKGBUILD files without breaking backwards-compatibility with Arch.

#66 - 2022-07-30 12:12 PM - bill-auger

Has this issue been reported?

yes - the FSF licensing officer in charge of the FSDG was fully aware of the situation, even before it actually happened - yet he allowed it to happen, and did nothing afterward

on that day, the reputation of the FSDG work-group was devastated; and the FSDG was effectively reduced to a trophy

now, each user must decide for themselves, which distro are actually following the guidelines, and which have strayed - of course, most would not know this, and will simply trust the trophies

it sounds like we need greater inter-FSDG-compliant-distro cooperation and organisation.

yes - that is needed and has been lacking for several years

I agree the solution to this ticket needs to be implemented consistently across all FSDG-compliant distros and all TPPMs.

only the FSF could make it consistent across distros - parabola
can only make it consistent across TPPMs

However, we have no consensus on what the solution is, other than to leave it lingering because we do not know what to do.

we have known what to do for a long time (comment # 14) -
its not so much about consensus, as feasibility

in order of increasing feasibility:

- A) convince the upstreams to institute clear licensing policies
- B) curate libre repos for all of them
- C) discard them all and start taking packaging requests (for PKGBUILDS)
- D) patch whichever TPPM clients do expose licensing metadata,
and discard the others, or disable their default repos
- E) patch all TPPM clients to disable their default repos
- F) put them all in a non-standard 'dubious' repo
- G) do nothing - sweep it under the rug like all other distros

we are still at the simplest stage (G) - freemor started on (F);
but that was only intended as a stop-gap ("well, we should do
something") plan - neither truly satisfy the FSDG IMHO

i am favoring C) discard them all and start taking packaging requests (for PKGBUILDS)

(C) is something we could accomplish immediately, with 100%^ effectiveness

I do not trust package license metadata whatsoever; they're often mislabeled.

indeed, those different options vary in effectiveness - but even
your favorite option (convince the repos to impose a policy)
does nothing to reduce mislabeling, unless they also audit all
packages, and declare the license themselves

we can make a solid case and be successful in persuading upstream to change the policies

i am not so optimistic - im quite certain that those people are
fully aware of copyrights and software licenses, and the dubious
nature of imposing a weak (or no) licensing policy

if they wanted a strong licensing policy, one would have been
in place years ago, probably on the first day when the service
was switched on

Starting our own project of packaging requests is not sustainable or portable

no one ever suggested that - RMS suggested that GNU would
maintain the repos

#67 - 2022-07-30 01:02 PM - jxself

FSF involvement isn't necessarily needed. The various endorsed
distros can also voluntarily cooperate amongst themselves and work
something out. Inter-distro cooperation doesn't have to require FSF
endorcement. The gnu-linux-libre mailing list was originally intended
to be for the purpose of such inter-distro cooperation.

Yes, and different distros may come up with different responses on
this. Doing (B) and possibly (D) would probably be best with
inter-distro cooperation.

Fortunately doing (A) doesn't preclude the possibility of also doing
other things.

#68 - 2022-07-30 01:16 PM - bill-auger

Fortunately doing (A) doesn't preclude the possibility of also doing

other things.

if the repos had strict libre-only licensing policies like haskell cabal, then nothing else would need to be done - it is the only one that i know of which is already FSDG-fit (or close enough - it requires an OSI license)

some cooperation from the upstream repos would be a requisite for patching the clients properly though - they would need at least to expose the licensing metadata (SPDX or whatever) - im pretty sure that relatively few of them do that now

#69 - 2022-07-30 01:51 PM - bill-auger

FSF involvement isn't necessarily needed. The various endorsed distros can also voluntarily cooperate amongst themselves

unfortunately, that is not likely to happen - pragmatically, we really need the FSF to play the "cat herder" role

if the FSF does not make it an imperative to follow all of the guidelines perpetually, then there is no incentive for distros to do so, because they have already been awarded the trophy, and are in no apparent danger of losing it

#70 - 2022-07-31 01:01 AM - gap

For the purposes of documentation, please may you link to the discussion which preceded this, if one exists, or an example of the proprietary packages in GNU Guix's repos?

Or, the ways in which

guix indexes and assists installing software which is explicitly forbidden by the FSDG

#71 - 2022-07-31 01:26 AM - bill-auger

the event, as it happened, was documented completely on the guix and FSDG mailing lists - that was in 2018 - [#2512](#) is as good as any document, if you are interested; but it would not be news to very many people - it seems that i am asked about it nearly once per week though still

#72 - 2022-07-31 01:28 AM - gap

Is that the full extent of the damage or does it go deeper; are there any other proprietary packages in GNU Guix's repos?

#73 - 2022-07-31 02:20 AM - bill-auger

- *Description updated*

#74 - 2022-07-31 02:25 AM - bill-auger

i dont know; but there only needs to be one example, to earn it a seat on the heap with the others - it would need to be modified with a filter, even if to filter only one package, or discarded, whatever the general plan is - gnutoo seems to prefer discarding them

until this ticket has a resolution, i would put webengine and any dependents on the list though, if any FSDG distros distribute those

#75 - 2022-07-31 05:49 AM - gap

Although I certainly do not defend this decision by the maintainers of GNU Guix's repos, AFAICT the decision appears not to be a statement of discarding the GNU FSDG or the principles of free software, but the unique, ugly result of this long-standing issue.

There is a difference between accepting whatever into a system's repos whilst disregarding freedom, just like all the other distros, and a unique, if mistaken choice made on a single, complex, long-standing issue.

Is this assessment correct?

If so, the issue of the GNU Guix repos including this one package can be easily remedied.

I think we've ascertained that the real, deep problem preventing us fixing this issue are that of the lack of cooperation and organisation between FSDG-compliant distros.

As this is the root cause of the blockade, I suggest we first focus on fixing it before attempting to begin one of the technical projects listed in the table in the description.

Until then, we cannot faesibly move on.

#76 - 2022-08-02 04:45 PM - GNUtoo

bill-auger wrote:

this small step is irrational though - why single out only pip?

Simply because I don't know the name of the other packages, and that to me it seemed pretty clear that we could not keep python-pip as-is because it conflicted with the FSDG (I explain the reasons of that below).

I'm also trying to fix an issue at the time, so I'd like to blacklist rubygems next (because it seems to have some consensus on the fact that it has no commitments to include only free software).

this ticket represents dozens of packages, which all deserve exactly the same treatment (whatever that is)

the standing consensus was to leave them be - i know we discussed blacklisting them and accepting packaging requests instead; but if the new treatment is to blacklist them, then we must blacklist them ALL, if the rationale is to be defended sincerely

Indeed, my intention was precisely to blacklist all the package manager that had repositories with no commitment to only include free software, and keep the package managers that do commit to only have free software packages.

and that definitely includes guix - guix indexes and assists installing software which is explicitly forbidden by the FSDG - pip does not - but pip is the one that people are most likely to complain about (they already have complained while the plan was only being discussed)

As far as I understood from this bug report, pip didn't have any commitment to only have free software, Guix in practice does (but it has other issues, see below).

And in the FSDG, there is a section about repositories:

The system should have no repositories for nonfree software and no specific recipes for installation of particular nonfree programs. Nor should the distribution refer to third-party repositories that are not committed to only including free software; even if they only have free software today, that may not be true tomorrow. Programs in the system should not suggest installing nonfree plugins, documentation, and so on.

So if I understood right, we should not refer to "third-party repositories that are not committed to only including free software", and if we ship python*-pip, we are doing that. Other programming language package managers that refer to "third-party repositories that are not committed to only including free software" have the exact same issues and I think we can safely blacklist them once we manage to link the package names to blacklist with the policies of the repositories they refer to.

From what I understood from the FSDG, the issue would only arise if the distribution refuses to fix the issue:

Most distribution development teams don't have the resources to exhaustively check that their distribution meet all these criteria. Neither do we. So we expect distros to occasionally contain mistakes: nonfree software that slipped through, etc. We don't reject a distribution over mistakes. Our requirement is for the distribution developers to have a firm commitment to promptly correct any mistakes that are reported to them.

So if I understood it right, we need to fix that issue in Parabola, and the easiest way is simply to blacklist problematic programming language package managers.

And as I understand, if we refuse to fix, then Parabola would stop being FSDG compliant and would need to be removed from the list.

But as I understand, Parabola still wants to keep being FSDG compliant, and the issue is more finding volunteers to fix the freedom issues we find.

So assuming I understood the FSDG right, my (harsh) action show that there is no consensus on the route to take, and that there is some misunderstanding somewhere (from either my part or people that decided not to fix), so as far as I understand, that also keeps Parabola FSDG compliant for now until Parabola somehow reaches a decision.

As for Guix, as far as I know Guix didn't refuse to fix bugs about "third-party repositories that are not committed to only including free software", they have other issues though (like referring to third-party repositories that are committed to include only free software but that themselves are not committed to not refer to "third-party repositories that are not committed to only including free software").

For that the FSDG needs to be clarified, or people need to become guix committers and go fix that.

If the clarification is to not have transitivity (like we could refer to a repository like Debian main that is 100% free software, but Debian main can refer to Debian nonfree) we would need to make it clear for users of FSDG compliant distributions in general that the distribution is only responsible for one level of indirection.

Though individual distributions can also choose to be more strict if they want.

like most of these FSDG grey-issues, I don't have a strong preference for which treatment is done; but I am extremely uncomfortable to apply a treatment (whatever that is) willy-nilly, to some examples but not to others

I agree with that.

At the same time we need to differentiate the fact that we rely on volunteers, and the decision made by the distribution.

Volunteers (like me) can simply fix bugs as they see it. Or fix one bug after the other. But for instance if rubygems is also removed (because someone found the time to do it), it is consistent.

And I personally try to remove everything that is not compliant with the FSDG, even if I use that software. For instance I blacklisted the tor-browser because it refers to an add-on repository that has non-free software, and I do use the tor-browser as well (without installing any add-ons).

so what shall I tell the next user who asks me "why did you blacklist pip, but you kept npm, rubygems, guix, nuget, cpan, pecl, flatpack, appimage, docker, and a dozen others?"

I've listed that here:

repository	reason
npm	I've no idea about its policies
rubygems	As far as I understand it should be blacklisted as well
nuget	I've no idea about its policies
cpan	should be removed but I've no idea which package has a cpan package manager
pecl	should be removed but I've no idea which package has a pecl package manager
flatpack	I've no idea about its policies
appimage	I've no idea about its policies
docker	I've no idea if it has a repository
guix	doesn't have non-free software, in the list of FSDG compliant distros + no policy in Parabola to exclude third party package managers like Guix

Can I go ahead and remove rubygems?

We for instance already removed packages using the LVFS repository because it contained non-free software.

I do understand that my move was a bit harsh to unilaterally do that for pip, but in the other hand, unless I misunderstood the FSDG or its intention, we have conflicting statements here, so for me what sounded like the best way to resolve that was to actually do the blacklist and start a discussion about it instead of letting the issue rot.

If it is all a big mistake from my part, I don't mind if people do revert my commits (assuming there are also good reasons to do it and that it doesn't break other stuff).

For instance a commit that tells precisely why my reasoning is wrong here or why having python-pip as-is is allowed would be very welcome.

I've also started writing documentation about this issue in a Libreplanet article (<https://libreplanet.org/wiki/Group:Software/research/ExternalRepositories>) in order to have resources that are usable by most FSDG compliant distributions in a place that is not specific to a distribution.

Denis.

#77 - 2022-08-02 05:03 PM - GNUtoo

gap wrote:

guix indexes and assists installing software which is explicitly forbidden by the FSDG

Has this issue been reported?

From your replies, it sounds like we need greater inter-FSDG-compliant-distro cooperation and organisation.

I agree the solution to this ticket needs to be implemented consistently across all FSDG-compliant distros and all TPPMs. However, we have no consensus on what the solution is, other than to leave it lingering because we do not know what to do.

For me what to do with programming language packages is pretty clear: not ship packages that use third party repositories that have nonfree software. For instance we patch browsers to not refer to the mozilla addon repository. In some cases there are alternatives being worked on (like for browser addons), and sometimes the software is just removed (like with fwupd that depended on LVFS). So basically we continue doing what we already do but apply it to programming languages repositories too.

What to do to improve the collaboration between FSDG distribution is less clear though.

Personally I'd like to rename or edit the description of the gnu-linux-libre mailing list to mention that it's about the FSDG (and not necessarily GNU/Linux or linux-libre as both aren't required for FSDG compliance).

And I would like for the distributions to be able to modify the FSDG document and come up with a process that would be fair, to be able to do that.

This way it would also involve more distributions as important things are decided.

I'm not sure if it would work though as I proposed the later on that mailing list and people probably weren't interested because they had to do more work.

As I stated previously, I think the only proper solution is to fix the root cause of the issue upstream instead of having to duplicate effort many times over by each distro implementing separate fixes.

I agree with that. For that we also need to be able to refer to third party repositories that are FSDG compliant. I've started listing some for browser addons for instance:

https://libreplanet.org/wiki/Group:Software/research/ExternalRepositories/BrowserAddons#Lists_of_FSDG_compliant_lists_of_addons

Guix would also be well suited to produce addons reusable by other distributions as it is the most strict distribution on reproducible builds and bootstrapable builds.

I genuinely believe we can make a solid case and be successful in persuading upstream to change the policies if only we cooperate between distros and form a plan.

I agree with that.

There is also the question of what to do in the short term. I think that blacklisting until we manage to fix the issue would work.

We are already doing that in general in FSDG compliant distributions: we don't ship a specific nonfree software (like the ath9k_htc firmware before it was released as free software) until a free replacement is made.

Only after this results in failure should we consider the next best option: starting forked repos which import and blacklist packages in the same style as Parabola and Arch.

There also might be in between solutions where we have both a local part and an upstream. For instance for u-boot I managed to move the code to deblob the source code in libreboot.

So Guix and other distributions can now use it (I'll probably send patches for that after my latest patch to add a more recent u-boot version in libreboot is accepted).

For the time being, patching out the upstream repo URLs should be a good temporary solution. Blacklisting the TPPMs entirely is too great a loss of features, which devs really need.

As long as it's FSDG compliant, I'm fine with it. People also really need some nonfree software, but Parabola doesn't ship that for instance. So if people install nonfree software on the side it's their problem. So just removing the URL and having instructions to add it back **outside of Parabola** can also work.

Replicant and Guix are in this situation: there is nonfree software repositories or instructions to install nonfree software on top of them completely outside of the main projects.

This makes it clear that the non-compliant software is not provided by the main distribution and users are perfectly aware that they are installing nonfree software.

Starting our own project of packaging requests is not sustainable or portable; the package recipes for TPPMs cannot use Parabola repos, rather we would have to offer an inferior dev experience on Parabola and force devs to "be their own package manger" and manually check the recipes, and install the corresponding Parabola packages.

If something cross distro is worked out, and that it's automatized a lot, it might be sustainable but it also needs people to work on it.

Denis.

#78 - 2022-08-02 05:16 PM - GNUtoo

gap wrote:

I'd also like to add that I do not trust package license metadata whatsoever; they're often mislabeled.

That happens often indeed.

I've some tricks to find that for GNU/Linux distributions:

- Debian has a list of files that is nonfree in debian/copyright in <http://salsa.debian.org/kernel-team/linux.git>
- Same with u-boot (there is at least 1 nonfree file in u-boot)
- sof-firmware (sound open firmware) contains nonfree firmwares: Intel firmwares are nonfree because they are signed and the hardware only accepts firmwares signed by intel, so users can't modify the firmwares.

So we would check these packages, and if they use ship nonfree software and that the distribution is ok about it. That only applies to software like debootstrap which is already patched in Parabola though (but not yet in Guix).

For programming language repositories we'd have to find similar packages, and the issue is that they will probably be programming language specific.

So we could try to ask ourselves which packages could likely contain nonfree software and find out what the rules are to be in the repository and how to fix it.

For distributions it's pretty straightforward but here the individual packages have way more autonomy so we probably need to find a way to test that by sending a bug report to packages that is labelled incorrectly until we find one that doesn't want to fix it, and report that to the package repository instead and see what happens.

Denis.

#79 - 2022-08-02 06:53 PM - GNUtoo

bill-auger wrote:

Has this issue been reported?

yes - the FSF licensing officer in charge of the FSDG was fully aware of the situation, even before it actually happened - yet he allowed it to happen, and did nothing afterward

What is public about that? Is the names of the package(s) known, the guix bug report(s) known? Was there a public discussion about the fact that Guix didn't follow the FSDG?

To be able to adjust our strategies collectively in the free software world we would need good information on that.

only the FSF could make it consistent across distros - parabola
can only make it consistent across TPPMs

However, we have no concensus on what the solution is, other than to leave it lingering because we do not know what to do.

As far as I understand, the distributions are free to choose the way they want to fix the problems as long as the solution is consistent with the FSDG. But we can propose valid and invalid solutions.

For instance keeping nonfree software inside Parabola on purpose is not a solution, removing the package with nonfree software is a possible solution, patching it is another solution, replacing it is yet another solution.

So like for removing nonfree software, we could propose also propose valid and invalid solutions for third party package managers on the Libreplanet wiki for instance. I've already started making pages about that.

That is said, beside the freedom to choose an actual solution, we also need to collaborate between distributions, and that's the part we have an issue with.

A way to do it would be to agree on a plans to replace a known package managers (but not necessarily commit to use it to avoid distribution fighting each others) and setup infrastructure for that and try to rely on contributions for helping adding packages. We'd need at least 1 maintainer to review patches, who would have 0 work to do if nobody send patches.

We could probably ask help from the FSF for the infrastructure not to have to spend resources on maintaining it.

I think that Guix itself is very well suited to replace repositories like pip as it has code that can already parse repositories like pip (with guix import). So it could probably be taught to make compatible repositories. It could also be taught to parse the free software directory for instance. So we need to bring Guix on board too and first fix things in Guix if that's necessary.

And for browser addons, since we have cross platform xpi, so a cross distribution solution could be relatively easy to setup. It could even be made optional where users would choose whether or not to use it. This way it would not conflict with existing distributions practices.

But again here that requires work from people, and I think that the issue here is that no one has time to commit to work on things like that, so people try to find way around that to get the problem magically solved. But given the scale of the issue, the problem won't fix itself.

So as solution I propose to use solution that makes the distribution compliant as fast as possible (like with Parabola's blacklist) and to work to apply that solution to all FSDG compliant distributions.

We might also need to structure a bit our plan too, like find which distributions we can contribute to (Guix, Parabola, Replicant and Trisquel do accept patches, I'm not sure about PureOS and Dragora), find allies that have commit access in the distributions, and so on.

Personally I'm already involved in trying to implement cross distro collaboration with u-boot, because there too we have a need of an upstream project that distributions could contribute too. I still need to wait for some patches to be merged before proposing the solution to Guix.

For Guix bugreport on debootstrap, I'd need examples of nonfree software the distributions already supported by upstream debootstrap. I didn't have the time to look yet, but if someone wants to do that we could advance on this topic.

we have known what to do for a long time (comment # 14) -
its not so much about consensus, as feasibility

in order of increasing feasibility:

- A) convince the upstreams to institute clear licensing policies
- B) curate libre repos for all of them
- C) discard them all and start taking packaging requests (for PKGBUILDs)
- D) patch whichever TPPM clients do expose licensing metadata, and discard the others, or disable their default repos
- E) patch all TPPM clients to disable their default repos
- F) put them all in a non-standard 'dubious' repo
- G) do nothing - sweep it under the rug like all other distros

I propose to find the shortest path to keep the distros FSDG compliant and then trying to find a longer term solution.

So anything fast is OK for me. Even (C) is OK as a quick solution, but we would need more discussion if (C) was chosen as a more long term solution.

Else it would not be consistent with the FSDG as anyone could keep any nonfree software until it's replaced by free software, and not even have the time to do A,B,D,E,F or G. The same applies to browsers addons.

we are still at the simplest stage (G) - freemove started on (F);
but that was only intended as a stop-gap ("well, we should do something") plan - neither truly satisfy the FSDG IMHO

And about G, Guix could also say that Parabola doesn't do it, so why should they do it? We need to start somewhere and the easiest way I think is to do the quickest thing to keep the distro compliant and then try to really fix. And the nice thing is that it's easy to fix in Parabola.

And once it's fixed in Parabola I might even have the time to go fix it in other distros as well, starting from the issues I know best.

Replicant was also in the same situation with F-Droid, at first I thought it was compliant, but then after looking into more details I found deep issues (like applications to install any gratis software from google play), so we tried to convince upstream to fix it, and we lost many years trying to convince them and it's still not fixed. So at the end it's easier to just remove it and try to fix later on. We even made precise plans to fix but we are now stuck needing someone to implement them.

And we're even stuck with dependency control issues: Contributing to F-droid requires to build it, and building it depends on maven central, and with maven central, often you only get binaries with no way to know where is the full and corresponding source code. So we might need to fix that first, and there Guix is helping too as there is someone packaging maven and so on and with Guix we can build some Android source code at least, though there is still some work needed to build android applications with it.

The alternative would be to not fix it, ever and continue shipping a package manager with problematic software in it. So now at least people do not think that what is in F-Droid is all good, even if they end up installing it anyway.

i am favoring C) discard them all and start taking packaging requests (for PKGBUILDs)

(C) is something we could accomplish immediately, with 100% effectiveness

I'm OK for that, let's do it.

Denis.

#80 - 2022-08-02 08:21 PM - GNUtoo

bill-auger wrote:

the event, as it happened, was documented completely on the guix and FSDG mailing lists - that was in 2018 - [#2512](#) is as good as any document, if you are interested; but it would not be news to very many people - it seems that i am asked about it nearly once per week though still

About that bug report, it's hard to understand if Guix listed all the licenses of Chromium or not because they did a huge job listing most of them.

So I asked around and I was told that some of the licenses were still missing, and I hope that the person who told me that is right because while it's simple to verify a license, it's harder to verify for a missing license in a project that big.

So if I understood well what I was told when asking around, that the issue here is how to classify software whose license is unknown.

We are used to usual cases:

Case	Software can be included in FSDG compliant distro
Free software license is applied correctly	Yes
copyrightable work without license	No (nonfree)
uncopyrightable work	Yes

The issue is that in chromium the code source comes from too much projects under too much licenses. If it was simply lacking a license it would be cristal clear that it could not be included because the code is big enough to be copyrightable.

Here to add to the complexity, Guix actually listed many of the licenses. So the question is if they listed them all, and if not if it's acceptable to include something like that in FSDG distributions. Here I assume they didn't list everything because I was told so, but I could not check.

Replicant is in the exact same situation because it uses Android, and Android has no package definitions and so the licensing information is complex to get (through they are available in the settings on the phone).

And Parabola is also in a similar situation (because we take packages from Arch and so we can **and do** inherit problematic packages without having reviewed them).

In all these 3 cases the distribution is not proactive: it doesn't review and then include the software, it's rather reactive: it fixes issues when found.

So this is a more complex issue that isn't even mentioned in the FSDG.

So I'd like to get the very basics right first because the FSDG has no mention at all of cases like that. And fixing the easy cases would help getting on the right tracks for fixing that harder issue.

And I feel that we have some tension here, and it would be really sad to have a war between distros being listed as FSDG compliant (with the FSF being used as the weapon), because at the end of the day the enemy is clearly nonfree software, and certainly not distributions that can collaborate together to fix the issue we have here.

If we have wars inside free software, free software is probably dead, and that's not something that we can afford right now (we also have a ecological crisis of an enormous scale and we need free software to help there too).

People that contribute to some of these distributions might not be on the same page, so that is an issue we need to fix **globally**. If there is a new Parabola developer that starts including nonfree software in Parabola for instance, it's that person that need to be dealt with (in a friendly way if possible), not Parabola that needs to be punished somehow.

So we do need allies inside Guix to fix things there too (like debootstrap for instance).

And since we badly need to not have that war, we could try to collaborate **with people within distributions** precisely on fixing FSDG compliant issues.

Personally I've already contributed to Guix, Parabola, Trisquel, Replicant and I'd also like to contribute to Hyperbola and LibreCMC. And I can only push code directly in Replicant and Parabola. Though in Replicant I only do it after sending the patches ending up in the images for reviews.

So I don't feel having Distro X versus Distro Y but I rather see that we have problems to solve globally in at least all FSDG distributions I've contributed to.

And if Distro X harms distro Y (through punishment by using the FSF for that for instance), distro Y might want revenge, and it starts becoming completely out of hand, so we need to have a mentality where we try to fix things and not try to punish distributions, else it could end up in a war.

That said if a distribution chose to follow the FSDG guidelines, they also have to at least try to follow them, else there is a consistency issue.

And if someone is reading that and wants to contribute, there is probably a lot of freedom related bug reports open in Parabola and we also have a very serious FSDG issue to fix in Replicant 6.0 as it doesn't compile yet on an FSDG compliant distribution (there is still work that needs to be done for that).

And I think Replicant 6.0 is still compliant because this is a bug and I've already worked many hours to try fix it (trying various distributions, adapting software, etc) and there was progress made, but the work still needs to be finished and I've only 24 hours in a day and also many other FSDG compliances issues to fix in FSDG compliant distributions in general (and we've fixed several important FSDG compliance issues in the last Replicant release for instance, so we do progress).

If instead the solution is to punish distributions, we would probably have to remove them all, because probably all of them have issues. Instead we need to fix these issues and only try to get distribution unlisted as the absolute last resort when the distribution do things that is very clearly forbidden in the FSDG and that the distribution (not a single contributor) really wants to continue doing that on purpose (not due to the lack of help, etc).

This has happened for ReactOS because ReactOS really wanted to ship nonfree software in their package manager.

As for the pressure to have one FSDG distribution ship software that the other distributions don't want to ship (that argument was probably mentioned on gnu-linux-libre at some point), I think that's even a good thing as long they comply with the FSDG. It's normal to have different policies in different distributions, and we also need to see things in a practical way.

If distro Z can't do the work and that distro Y does it instead, that gives users of distro Z an easy solution.

For instance if someone decided to make a repository of addons for browsers or Android package without even recompiling them, that would conflict with the distribution policies but maybe not with the FSDG as long as there is way to fix (for instance by reviewing the source code, removing problematic packages, etc).

So it would still give users a way to install the free software addon they really need.

And if distro A is in the list of FSDG compliant distros and ship a problematic package P that is nonfree, the damage is not done on the other FSDG compliant distributions. It's rather done on the users that didn't know that package P was nonfree. And if it's not fixed on purpose, it also harms our communication strategy generally speaking as promoting distros becomes complicated.

And if FSDG distributions don't ship P and that users still want to install P, I've no issue with that as long as they don't bring other users to use P as well (that can happen if it's a communication program that has no free implementation for instance). What is important here is that FSDG distribution don't ship nonfree software. What users do on top is a completely different issue.

So the question is also here the goal of FSDG compliant distributions. And for me these distribution should not ship nonfree software regardless of if users install nonfree software on top or not. These are two different issues.

This ensures that at least that distribution is fully free (and goes beyond that as the FSDG has additional requirements that for instance ensure that users don't accidentally install nonfree software without knowing it).

And here's some examples of practical advantages of that freedom:

- if you use RYF compliant hardware (computers, GPUs, etc), the distribution will not run nonfree software behind your back (if you don't it usually will).
- You know what works with free software or not
- The distribution is legally clean, so if other distributions are attacked, the FSDG ones will be OK.
- The distribution doesn't make users accept licenses, so you can practice reverse engineering to replace nonfree software with free software.

In the free software movement, we're also not in a position to completely get rid of nonfree software in all the distributions. So instead the freedom of FSDG distribution brings many advantages as explained above (like all freedoms in general). And we can really adopt strategies that leverage the existence of FSDG compliant distributions to improve freedom globally.

An example of that is that the existence of FSDG compliant distributions probably played a big role in getting the ath9k_htc firmware freed.

People could also try to get funding to solve issues that affect users of FSDG compliant distributions. NLnet can also fund many tasks that improves software freedom (with a big focus on having the changes upstreamed or benefit as much projects as possible).

So if it's framed right, we could even get funding to solve that mess. For instance there are real attacks due to dependencies being out of control, and in npm that made the headlines in the technical press at least.

So it's very likely to be able to propose a project like that to NLnet and get funded.

And having something like Guix is a very strong asset here as it is very flexible. Its packages can be used to make make docker VM, Debian packages, etc. It could easily be taught to make appimages, parabola packages, and probably even repositories. The downside is that it's not a drop-in replacement for npm, pip, etc, but at least it really solves the dependencies issues as it's somehow bootstrapable and reproducible. So getting funding for working on it is easier.

Denis.

#81 - 2022-08-03 12:13 PM - bill-auger

we need to fix that issue in Parabola, and the easiest way is simply to blacklist

it is agreed that something should be done; but all proposed solutions entail blacklisting, necessarily - the dilemma is, do we blacklist them first and fix them later (or never), or do we fix them first then blacklist them

we do have something of a policy about this - from the blacklist README:

```
1. Start by trying to liberate the package in whatever way possible
   in order to make it useful in freedom. If that is not possible or is
   too much work or will be too high-maintenance relative to the program's
   usefulness, try finding some alternate replacement program
   that accomplishes the same task. If that turns out to be is similarly
   unfruitful, simply continue blacklisting the package with no replacement.
```

what it does not suggest, is how much effort to put into the liberation effort before surrendering to defeat, or how long that procedure should take before it would be considered as negligence - surely 6 years is excessive though

according to the table in the OP, the optimal solution is to blacklist them and never fix them - but that is because only the workload factor of the desirability/workload ratio is considered - the desirability factor is difficult to determine without a user poll or something - the desirability factor may be rather high for some of these, though i would argue otherwise

that also keeps Parabola FSDG compliant for now until Parabola somehow reaches a decision.

but it does not do that - it only addresses one known example - it leaves dozens of other known examples untreated - it is not "too harsh" - it is either not harsh enough, or it falls far short of it's intended impact

regarding the table you added, i had started such a table years ago (see comment # 3) - i intended that we would fill-out that table completely, before taking any action - i should move that information into the OP also

I'm not sure about PureOS and Dragora

AFAIK, Dragora does not distribute any TPPMs - it is probably the "purest" distro on the planet, in every sense of the word

#82 - 2022-08-03 12:36 PM - bill-auger

As for Guix, as far as I know Guix didn't refuse to fix bugs about "third-party repositories that are not committed to only including free software", they have other issues though (like referring to third-party repositories that are committed to include only free software but that themselves are not committed to not refer to "third-party repositories that are not committed to only including free software").

that was super confusing - i dont know what you meant by that - but i will be absolutely clear what i meant

- from parabola's perspective, guix *is* a TPPM
- guix distributes chromium
- chromium is explicitly forbidden by the FSDG
- the guix TPPM will suggest chromium, and assist users to install it
- the guix maintainers know that it is an FSDG freedom bug, and have refused to fix it

therefore, from parabola's perspective, the guix program and it's repos are not FSDG-fit - this is not about guix (the distro) and their policy regarding third-party repos - the guix maintainers are clearly not committed their own promise to follow the FSDG; so we must treat it like the others

the main reason why i have been reluctant to move forward, is because i want to apply the same criteria to all TPPMs fairly, but was hoping to avoid the embarrassment of highlighting that there is a GNU project which hosts the very sort of third-party repos, which the FSDG requires us to avoid directing users toward

but now you forced the issue; so if pip has to go, then guix has to go too - guix should have gone first, on the day that chromium was added to it - chromium in guix was not a "mistakes: nonfree software that slipped through" - it was added by the lead maintainers, in full knowledge that it was forbidden - they contend that it is acceptable free software; but that claim is irrelevant - if the FSDG work-group accepts that claim, i may not bother to contest it - but nonetheless, chromium is explicitly forbidden by the FSDG today - that ugly fact should not be ignored

AFAIAC, parabola is already guilty of violating the FSDG, because we knew that guix allows installing chromium, and we did nothing about it, beyond reporting the freedom bug to guix - guix is clearly not going to remove chromium; and the FSDG is not going to accept it any time soon, if ever - i think the guidelines are sufficiently clear already, to warrant its removal from parabola, unless we are going to patch it

Guix actually listed many of the licenses. So the question is if they listed them all

that is not the pertinent question - it is not sufficient to compile a "list of detected license files" - IIRC, debian did that years before guix existed - a tool like fossology or the debian license-checker can find common license files in minutes - but a heap of permissive licenses, gathered from a 4GB code-base, demonstrates absolutely nothing, WRT which files each license applies to, or which files were never published under any license, but were simply inserted under some tree branch, by someone other than the author, who knows who?, who knows when?, who remembers why?

it is infeasible to prove that the chromium code-base is either distributable, or not

guix contends that the infeasibility of proving the code-base to be not distributable, equates to software freedom

i contend the opposite - the infeasibility of proving that the code-base IS distributable, disqualifies it from any defensible conclusion

if you knew for a fact that there was a file lurking in that code base, which is a verifiable copyright violation, but you were not told which file that was, it would be the proverbial "needle in a haystack" to locate it - that would need to happen before you could begin to invent a liberation procedure - in this case, guix is merely assuming that there are no needles in the haystack, because the haystack is too large to search

i do not add any new package to parabola, before sifting through the entire haystack to the best of my ability, until my ambition is exhausted - it often leads down a rabbit-hole; and in those cases, i cut my losses and apologize to the person who requested it - i would not blindly accept it, then dare the community to prove me wrong, knowing that the task is so difficult, that no one will ever bother to try - i would be ashamed of myself

that is exactly the "open-source" mentality isnt it? (to prefer convenience, functionality, or "awesome-ness", over the assurance of freedom)

let us not derail this ticket onto the chromium track though - that is well discussed all over; and there is nothing new to add - i may revisit this ticket later, and prune most of the off-topic discussion - this ticket is way too long already

the only point i want to make WRT this ticket today, is: "if pip goes, then guix goes with it" - i am not going to resign from that position - the FSDG requires us not to make chromium available - the only way to comply with that, it to get rid of guix, or patch it

And having something like Guix is a very strong asset here as it is very flexible. Its packages can be used to make make docker VM, Debian packages, etc. It could easily be taught to make appimages, parabola packages, and probably even repositories.

ok, then you just volunteered to patch it - or else, you just played the "open source" card (a willingness to tolerate dubious, limited, or marginal freedoms, in favor of utility, convenience, or popularity) - oaken-source very much wants to keep pip; so maybe oaken-source will patch pip

i am not much motivated to patch any of them - most, if not all, of those upstreams publish standalone binaries of the TPPM - if people are willing to introduce third-party software into their system, then there is no reason not to get the TPPM from the same third-party - compared to the third-party packages that these will be used to install (normally by the *hundreds*), the TPPM itself is but one more

#83 - 2022-08-03 05:11 PM - GNUtoo

bill-auger wrote:

but it does not do that - it only addresses one known example - it leaves dozens of other known examples untreated - it is not "too harsh" - it is either not harsh enough, or it falls far short of it's intended impact

By "harsh" I meant that a group of Parabola contributors took a decision and I acted unilaterally against that decision without even consulting people before.

So here I had very good reasons to do that, and I think that the end result (some tension + moving back the discussions in the good direction) was worth it, but that doesn't make these action not harsh (as it probably created some tension).

Denis.

#84 - 2022-08-03 06:54 PM - GNUtoo

bill-auger wrote:

As for Guix, as far as I know Guix didn't refuse to fix bugs about "third-party repositories that are not committed to only including free software", they have other issues though (like referring to third-party repositories that are committed to include only free software but that themselves are not committed to not refer to "third-party repositories that are not committed to only including free software").

that was super confusing - i dont know what you meant by that - but i will be absolutely clear what i meant

I meant that from Guix perspective, Guix ships an unmodified debootstrap.

That unmodified debootstrap has installation scripts for distributions like Ubuntu.

Ubuntu itself is not committed to only refer to repositories with free software.

And the FSDG only that:

The system should have no repositories for nonfree software and no specific recipes for installation of particular nonfree programs. Nor should the distribution refer to third-party repositories that are not committed to only including free software

Bug reference: <https://issues.guix.gnu.org/42964>

So if we take the FSDG by the letter and not look at what it tries to do (which is not completely clear either), it is compliant.

So unless I manage to prove that one of the distros ships nonfree software in the repositories mentioned by debootstrap (usually the main repository) it's probably going to stay like that.

So to fix the underlying issue we need some clarification of the FSDG, and also to see things in a practical way. For instance if the FSDG allows debootstrap to refer to debian main and that with this position there is a way to prevent users from accidentally installing nonfree software, then I'm OK with it.

So like with code and commit messages, we can look at the practical aspects to decide.

- from parabola's perspective, guix *is* a TPPM

Yes

- guix distributes chromium

Yes

- chromium is explicitly forbidden by the FSDG

I don't see chromium mentioned in the guidelines¹, so maybe you meant that chromium is incompatible with the FSDG. And so far I didn't find any rationale for that.

The only rationale I found was for deciding to modify the FSDG to exclude chromium, which is a very different proposition.

¹<https://www.gnu.org/distros/free-system-distribution-guidelines.html>

- the guix TPPM will suggest chromium, and assist users to install it
- the guix maintainers know that it is an FSDG freedom bug, and have refused to fix it

For me it's not clear if it's an FSDG freedom bug or an unconfirmed bug in the FSDG itself.

therefore, from parabola's perspective, the guix program and it's repos are not FSDG-fit - this is not about guix (the distro) and their policy regarding third-party repos - the guix maintainers are clearly not committed their own promise to follow the FSDG; so we must treat it like the others

My question is also if Parabola is really the place where we should decide if a given distribution is FSDG fit or not.

A better idea would be to use the GNU Linux libre mailing list for that and have a wider conversation on (1) how to put the procedure in places to make the distributions take decisions, and (2) take a decision on that.

Once the decision is taken we can then explain the rationale of all that publicly and then try to make the gnu project website maintainers just apply the decision.

the main reason why i have been reluctant to move forward, is because i want to apply the same criteria to all TPPMs fairly, but was hoping to avoid the embarrassment of highlighting that there is a GNU project which hosts the very sort of third-party repos, which the FSDG requires us to avoid directing users toward

As I see it we can still do the same thing as usual: find non-compliant packages and deal with it. It could be done one package at a time.

The advantage is that if we can't commit enough time to fix it all, we at least have something, and that's better than nothing.

And personally I usually find the packages I use or know first.

And both approach are not mutually incompatible: if pip is already removed, that doesn't prevent removing all other programming language packages in one go. And if rubygems is removed, it won't even be an issue (a simple rebase should be enough to not loose the work).

but now you forced the issue; so if pip has to go, then guix has to go too -

The FSDG is clear on not including repositories with no commitment to only have free software.

So if I didn't remove pip we still had to deal with the issue.

And if we follow the rationale backward, if we kept pip, we should also have kept LVFS which clearly had nonfree software, and we should not filter Arch linux repositories at all, and so Parabola should not exist since almost all its software is already in arch or in Aur. So the backward rationale clearly doesn't work.

Another way of seeing it was to make an explicit exception for programming packages managers, but that conflicts with the FSDG.

So all that brings choices to be made by Parabola:

- Do we continue following the FSDG, personally the response is yes, especially because users do expect that.
- What policies did or will Parabola take that is more strict than the FSDG.

For the later we already took the decision to officially only support ARM computers that boot with free software.

Unofficially we can add support for computers with nonfree boot software, but there is no guarantee that it will continue working, and companies can't make the claim that their hardware which has nonfree bootloader is officially supported by Parabola, just that it works for now and might break in the future.

Decisions can also be revisited. Though we don't really have clear processes for taking or revisiting decisions, but at least it seemed to have worked last time we took one.

So we could:

- Come up with a more formal process for decisions
- Put precise decisions to discussions at least

And since we already did a lot of work in that thread, we can resume all the positions we took to not have to repeat that again.

For instance given the above we could ask the following questions to users and contributors:

- Should Parabola stay FSDG compliant? Or should it decide to make an exception that is incompatible with the FSDG to keep all programming language package managers and be removed from the list of FSDG compliant distributions? [We would also explain in details the issue linking pip and rubygems to the FSDG non compliance].
- Should we exclude all third party package managers even if they are FSDG compliant? [TODO: find some rationale and add it somewhere. So far I didn't find any good rationale yet for it, but at least it seems to be a valid question as it aligns well with one of the solution you suggest].

I didn't find any other consistent propositions.

Explicitly forbidding Guix would be discrimination against Guix if it has no rationale.

If we add a rationale, we would probably need to also not use Arch Linux repositories (more on that below), and Parabola is not in a position to do it right now, so it doesn't make sense to take a decision like that until we are actually capable of doing that.

guix should have gone first, on the day that chromium was added to it - chromium in guix was not a "mistake: nonfree software that slipped through" - it was added by the lead maintainers, in full knowledge that it was forbidden - they contend that it is acceptable free software; but that claim is irrelevant - if the FSDG work-group accepts that claim, i may not bother to contest it - but nonetheless, chromium is explicitly forbidden by the FSDG today - that ugly fact should not be ignored

The issue here again is how to verify a claim like that.

For you it is clear in your head, but not for me and probably not for the FSF either. So you need to have compelling argument that this is actually the case.

You gave compelling arguments that explained why shipping chromium had issues, but that is a very different issue than being non FSDG compliant.

For instance if there is a decision taken by a court, when it's done right, the decision is often justified. In some revolutions, (like the Paris commune of 1971), the decisions (for instance to destroy a monument) are also justified.

That justification enable anyone to understand why it was taken that way and it also enables discussion about it (some people might disagree on political ground for instance).

So here I think that anyone needs to be able to verify claims like that.

It's like when doing commits, you can optimize them for making each commit as easy as possible to review.

Here I had to resort to trusting people about this issue to understand what it was about. So we are very far from that situation.

A wiki page that would document the issue in a very clear and concise way would work for me. We could then discuss it.

Maybe that page already exist, if so it would be important to add the link in this discussion.

A practical example I can think of is that the FSDG has the following:

Data that isn't functional, that doesn't do a practical job, is more of an adornment to the system's software than a part of it. Thus, we don't insist on the free license criteria for non-functional data.

And I think it is a problem to not have the ability to modify art works, because art is as important as Wikipedia: it is culture and produces a shared understanding of the world. The precise words used in of a protest song can be extremely important for instance. Not being able to modify it is extremely problematic.

That said I don't know if it's strategically sound or not to avoid nonfree non-functional data. But assuming it is, I would then then disagree with the FSDG.

Then I've several choices. I could for instance try to require free data in at least a GNU/Linux distribution I contribute to, and then also require that for Replicant and a small distribution, and see how it works out. If it's practical I could then argue that the FSDG needs to be modified.

But I would not tell the distributions that they need to do that because the FSDG requires it. I would tell precisely the opposite to also try to gather data that shows this is practical to do (because nonfree art could be found in a lot of places, so showing that it works would be a good idea).

And I would need really compelling arguments here because the distribution doing that would require way more work and might not even be bootable anymore. I could for instance try to add a your-culture-freedom in Parabola to see if it could work out, and look at other distributions (like Debian) to see if it works with similar requirements.

AFAIAC, parabola is already guilty of violating the FSDG, because we knew that guix allows installing chromium, and we did nothing about it, beyond reporting the freedom bug to guix - guix is clearly not going to remove chromium; and the FSDG is not going to accept it any time soon, if ever - i think the guidelines are sufficiently clear already, to warrant its removal from parabola, unless we are going to patch it

We also have many more issues. There is a lot of freedom bugs open. And we badly need to recruit people to share the load of the maintenance. Reviewing patches in a timely manner would help a lot with that I think.

In addition we have a serious issue with the PKGBUILD licenses. So at least we now have some PKGBUILDS that are 100% free software, but this issue is far from being resolved. This is really worrying as it's very difficult to fix as we could fix our PKGBUILDS, but we also need to fix ARCHs as well.

Guix actually listed many of the licenses. So the question is if they listed them all

that is not the pertinent question - it is not sufficient to compile a "list of detected license files" - IIRC, debian did that years before guix existed - a tool like fossology or the debian license-checker can find common license files in minutes - but a heap of permissive licenses, gathered from a 4GB code-base, demonstrates absolutely nothing, WRT which files each license applies to, or which files were never published under any license, but were simply inserted under some tree branch, by someone other than the author, who knows who?, who knows when?, who remembers why?

it is infeasible to prove that the chromium code-base is either distributable, or not

guix contends that the infeasibility of proving the code-base to be not distributable, equates to software freedom

i contend the opposite - the infeasibility of proving that the code-base IS distributable, disqualifies it from any defensible conclusion

What happens if we hold Parabola to the same standard?

We just need to take not a given package but rather a snapshot of the Parabola mirror at any given point, and with that we don't know if Parabola is compliant or not. This is because we take packages blindly from the various Arch Linux.

And again that is not mentioned in the FSDG and you could argue quite the opposite:

Most distribution development teams don't have the resources to exhaustively check that their distribution meet all these criteria. Neither do we. So we expect distros to occasionally contain mistakes: nonfree software that slipped through, etc. We don't reject a distribution over mistakes. Our requirement is for the distribution developers to have a firm commitment to promptly correct any mistakes that are reported to them.

Maintenance

To be listed, a distribution should be actively maintained, and should give the GNU Project a clear and specific way to report problems of nonfree software that we find out about. It should also inform us when the problems we have reported are fixed.

Both paragraph are not incompatible with checks done after the fact and I think that if there we were to have a precise requirement about it it should be in the FSDG because to me it's not clear at all that the FSDG implies to do check a priori.

You could even do things that are counter to ways distributions typically work with the FSDG, for instance you can have an FSDG compliant distribution whose goal is to have as much vulnerabilities as possible (for instance to enable people to train to exploit security vulnerabilities).

You can probably also have distributions that do not build the software from source and have other means to deal with non compliant packages (like a filter system).

You can also have distributions that are not operating systems (like a browser addons distribution), and it would probably be considered as a small system distributions.

The FSDG and individual distribution policies are two very different things.

Parabola could take more strict decisions for software (Guix did for instance as they require software to be bootstrapable and reproducible).

The question is that if we take such a decision, we need to define it so that we don't end up applying it in ways we didn't intend. And we also need to have good rationale for it.

if you knew for a fact that there was a file lurking in that code base, which is a verifiable copyright violation, but you were not told which file that was, it would be the proverbial "needle in a haystack" to locate it - that would need to happen before you could begin to invent a liberation procedure - in this case, guix is merely assuming that there are no needles in the haystack, because the haystack is too large to search

I can make a false claim about that for all the Parabola packages. And then we'd have the exact same issue.

Note that I'm not claiming that we should not proactively review things, I'm only claiming that the FSDG doesn't require that, and that doing this would be a decision that needs to be taken independently by the distributions.

And in that case like with every decisions, we need to think about it (like see what the effects would be, discuss it etc).

i do not add any new package to parabola, before sifting through the entire haystack to the best of my ability, until my ambition is exhausted - it often leads down a rabbit-hole; and in those cases, i cut my losses and apologize to the person who requested it - i would not blindly accept it, then dare the community to prove me wrong, knowing that the task is so difficult, that no one will ever bother to try - i would be ashamed of myself

I've a very different approach: I often do add packages without having read all the source code (I often only look at it very rapidly).

But I also use intuition to try to find nonfree software and so far that worked quite well.

The issue here is that a lot of packages are added without being reviewed at all. These packages comes from the various Arch Linux distributions.

So in my case I am more successful finding nonfree software by thinking about where nonfree software could be hidden and verifying things.

It's a bit like security reviews. You know common patterns of where to find vulnerabilities, so you start with that. And for that you need to also have good models of how the politics, economy, human emotions works to understand where these nonfree software or security issues could be.

And for instance I can't review all the Linux source code, there is just too much source code. So here too we come up with strategies to learn about or detect nonfree software.

Linux-libre has a lot of code to do that, including code that can recognize instructions from data. And sometimes even with all that new nonfree software is found by humans (because they looked at the right place).

So we cannot review everything. Now the question could be how to be most effective in finding nonfree software.

A proposal would be to again recruit many more volunteers and maybe start building all the packages ourselves. That would be a tremendous work so we'd need many more volunteers.

So with the FSDG (not in Parabola), we might not want to put the bar too high and end up discriminating against distributions with less people and with less time available. Though we don't want to put the bar too low either, and for that the reviews to accept distribution is a good thing as if all usual packages are modified to be FSDG compliant, we have a good enough state to start from. And people would probably try to find nonfree software and the distribution is probably found when there is no more nonfree software to be found easily.

And once the distribution is in the list, the same procedure still applies: if people find new nonfree stuff it should be fixed.

that is exactly the "open-source" mentality isn't it? (to prefer convenience, functionality, or "awesome-ness", over the assurance of freedom)

Open source has nothing to do with that issue. Here we're definitely talking about freedom which open source doesn't care about (beside non discrimination against fields of endeavor).

And where to put the bar isn't related to open source at all.

let us not derail this ticket onto the chromium track though - that is well discussed all over; and there is nothing new to add - i may revisit this ticket later, and prune most of the off-topic discussion - this ticket is way too long already

the only point i want to make WRT this ticket today, is: "if pip goes, then guix goes with it" - i am not going to resign from that position - the FSDG requires us not to make chromium available - the only way to comply with that, it to get rid of guix, or patch it

Again I don't see why the FSDG requires to remove Guix's Chromium.

And having something like Guix is a very strong asset here as it is very flexible. Its packages can be used to make make docker VM, Debian packages, etc. It could easily be taught to make appimages, parabola packages, and probably even repositories.

ok, then you just volunteered to patch it - or else, you just played the "open source" card (a willingness to tolerate dubious, limited, or marginal freedoms, in favor of utility, convenience, or popularity) - oaken-source very much wants to keep pip; so maybe oaken-source will patch pip

Open source is not concerned with users freedom at all.

If we take this rationale further we could state that free software should not exist because it only runs on hardware that has nonfree software in it. That logic obviously doesn't work (because we need free software to design hardware too), so the conclusion is that we can either be against computers (with the rationale that the freedom status is not good enough) or we can adjust a bar between what's ok and not OK strategically and discuss that bar. Given that most activists use computers (including ecology activists), I think pursuing the later is more efficient.

So the question is where to put that bar, and it's not necessarily at the most extreme border because doing compromises (for improving freedom in the long run) would be "open source".

We need to think strategically so again here we could look at the effects of placing the bar here or there and try to see where it would maximize freedom in the short and longer term and see how to articulate that strategy over time.

And given that in Parabola we can be more strict than the FSDG, we have the power to make changes like that in Parabola. For instance not officially supporting ARM computers with nonfree bootloader is a strategic decisions that we took. We could also have took the decision to support as many ARM computers as possible given the FSDG constraints.

What we cannot do is be less strict than the FSDG (because users expect Parabola to follow the FSDG). And there is room for interpretation too in the FSDG.

i am not much motivated to patch any of them - most, if not all, of those upstreams publish standalone binaries of the TPPM

Nice, so at least we have one less thing to care about.

edit1: Added (more on that below) to "If we add a rationale, we would probably need to also not use Arch Linux repositories, and Parabola is not in a position to do it right now, so it doesn't make sense to take a decision like that until we are actually capable of doing that."

edit2: moved the "(more on that below)" mentioned in the edit1 from the end of the sentence to after "not use Arch Linux repositories".

Denis.

#85 - 2022-08-04 06:42 AM - bill-auger

That unmodified debootstrap has installation scripts for distributions like Ubuntu. Ubuntu itself is not committed to only refer to repositories with free software. And the FSDG only that:

The system should have no repositories for nonfree software and no specific recipes for installation of particular nonfree programs. Nor should the distribution refer to third-party repositories that are not committed to only including free software

So to fix the underlying issue we need some clarification of the FSDG, and also to see things in a practical way. For instance if the FSDG allows debootstrap to refer to debian main and that with this position there is a way to prevent users from accidentally installing nonfree software, then I'm OK with it.

i think the FSDG is already clear enough about that - the first sentence of that paragraph is most clear

A free system distribution must not steer users towards obtaining any nonfree information for practical use, or encourage them to do so.

"steer" is the operative word - the rule of thumb would be: if someone uses *any* software distributed directly by the distro (eg: debootstrap or pip), will that software "steer" (lead to, suggest, recommend, etc) to acquiring non-free software?

if debootstrap is configured only for debian main, and does not suggest enabling any other specific repos (like the debian installer does with a checkbox), then the answer is "no" - the user will not find/get any non-free software, by using that program in its intended/configured ways

if debootstrap is configured to install non-free software from debian or ubuntu, or if the user is presented with that option, then that is "steering" users toward non-free software

most TPPMs fail that test, even without a search feature, because if you `tpm install foo`, which you believe is libre, because 'foo' is BSD-licensed, the TPPM will automatically

install all dependencies of 'foo', some of which may be non-free

the part you quoted: "third-party repositories that are not committed to only including free software" is not adding anything substantial - the policies of the third-party are not pertinent to the FSDG - the distro maintainers are vouching for the third-party repo, by making it available to their users - that warning about third-party policies is only a warning to the distro maintainers, not to trust repos which *incidentally* have no non-free software today, unless it also has some policy which would prevent non-free software from being published, and help the distro to correct the mistake - the distro is responsible for that mistake, by vouching for that third-party

the distinction is usually very obvious - debian has a strict policy which would prevent non-free software from entering that specific repo, and debian vets all packages - pip and most others have only one repo for everything, with vague policies, and they vet nothing

for that reason, i would be willing to vouch for debian main and the haskell repos, due to their strict policies - WRT the FSDG, their mistakes are actually our mistakes; though i would be confident that debian and haskell would help us to correct such mistakes - if they did not fix reported freedom bugs, i would remove those tools from the distro too; regardless of their advertised policy

conversely, if it were possible to filter the repo, than that would also meet the FSDG; because the tool would not be steering toward non-free software - so again, the third-party's policy is not relevant to the FSDG

So unless I manage to prove that one of the distros ships nonfree software in the repositories mentioned by debootstrap (usually the main repository) it's probably going to stay like that.

ok, and here is my opinion - why should *you* need to prove that? - i think you are being much too generous - if anything, you should only need to raise a reasonable doubt, that the tool is likely to install or suggest non-free software - the distro should be required to disprove the suspicion, or stop steering users toward that suspicious software somehow

the bottom line is, that unless the FSF is willing to assert its authority in cases where a distro is not "correcting mistakes" or the distro introduces them knowingly, then yes, "it's probably going to stay like that", regardless of any user complaints - if any other user (or the distro maintainers) dismiss the suspicions, because "... but the tool is so useful", the distro has no incentive to correct that mistake - the distro has more incentive to please it's users or itself instead

without oversight, the FSDG reduces to the honor system - the responsibility of auditing software, is upon each user, rather than the distro maintainers (exactly as if there was no FSDG) - in that case, people may as well use debian or any other distro, and apply the FSDG to it themselves, per their own judgment

I don't see chromium mentioned in the guidelines¹, so maybe you meant that chromium is incompatible with the FSDG. And so far I didn't find any rationale for that.

no, i meant literally: "chromium is explicitly forbidden, by name" - i assumed that you were aware of that - donald made it so in 2018, and the only distro which distributed chromium (pureos) removed it shortly afterward - i wrote about that on the parabola dev list, just a few months ago

the chromium web browser is on the "List of software that does not respect the Free System Distribution Guidelines" wiki page¹ - the presence of any software on that list, prevents any distro from being endorsed by the FSF, if it distributes any of the software on that list, without applying an accepted liberation procedure; and puts the ongoing endorsement in

jeopardy for any distro which later accepts any of them
un-treated - currently, the only accepted liberation procedure
for chromium is: "use icecat"

[1]:

https://www.libreplanet.org/wiki/List_of_software_that_does_not_respect_the_Free_System_Distribution_Guidelines#chromium-browser

the explicit requirement is the checklist item: "Programs
commonly known to have freedom issues are liberated or excluded"

https://libreplanet.org/wiki/Template:FSDG_Checklist

that checklist is the complete criteria for the "Endorsement
Process"; and donald confirmed that all criteria are binding
perpetually, for all FSDG distros

https://libreplanet.org/wiki/Incoming_distros#Endorsement_Process

that is a reasonably solid "QED", IMHO - i specifically asked
donald to confirm that, to remove all doubt

My question is also if Parabola is really the place where we should decide if a given distribution is FSDG fit or not.

it is not; and i would rather not discuss that - but we
are responsible for vouching for the software in third-party
repos - we must avoid steering users toward any software on the
FSDG "ugly list", which is not treated according to the accepted
recommendation

So if i didn't remove pip we still had to deal with the issue.

we do still need to deal with the issue, WRT the dozens of other
TPPMs - we can not justify blacklisting only one - they all need
to stay or go together, in a single commit, all with this
ticket as the bug reference; because any liberation procedure
could be expressed generically for all of them (whichever one
from the table in the OP)

then, we actually need to continue identifying more of them,
which are not yet listed on this ticket - these are only the
obvious, widely-known ones

- What policies did or will Parabola take that is more strict than the FSDG.

"init-freedom" and "free-culture" are more strict than the FSDG

That said I don't know if it's strategically sound or not to avoid nonfree non-functional data. But assuming it is, I would then then disagree with the FSDG.

you would not be in doubt, if you were an artist or musician, and
cared about "artistic freedom", analogously to software freedom
- it does not disagree or conflict with the FSDG - it is an
extra requirement, beyond what the FSD requires

i contend the opposite - the infeasibility of proving that the code-base IS distributable, disqualifies it from any defensible conclusion

What happens if we hold Parabola to the same standard?

all software distributors should be held to that standard - if
anyone contests some software in the parabola repos, we should be
willing and able to defend it to some reasonable degree; or we
should patch it, or remove it - that is the entire point of the
FSDG - if users must prove which distro packages are non-free,
they could use any distro, and prove that to themselves only,
without complaining to anyone

indeed, we could hold icecat and iceweasel up to the same light;
though no one ever has - if people expressed reasonable doubts

about those, then parabola should not distribute them, unless we, or the FSDG work-group could prove that the suspicions are unfounded

but the fact today, is that the FSDG recommendation is "use icecat - not chromium" - unless that changes, parabola is meeting that standard of defensibility

#86 - 2022-08-04 12:23 PM - GNUtoo

bill-auger wrote:

So unless I manage to prove that one of the distros ships nonfree software in the repositories mentioned by debootstrap (usually the main repository) it's probably going to stay like that.

ok, and here is my opinion - why should *you* need to prove that? - i think you are being much too generous - if anything, you should only need to raise a reasonable doubt, that the tool is likely to install or suggest non-free software - the distro should be required to disprove the suspicion, or stop steering users toward that suspicious software somehow

It's easy to make FUD, and often users make confusions in bug reports, or have different interpretations of the FSDG, or want to force the distribution to change policy without having wide discussions before.

So I don't see what is wrong with requiring some investigation before actions are taken.

These are distribution decisions and the FSDG doesn't require to remove everything that is suspicious.

If I wrongly claim that linux-libre is nonfree because it uses the Linux trademark, I think that the distribution still has the right to not remove linux-libre without first having that claim validated somehow.

the bottom line is, that unless the FSF is willing to assert its authority in cases where a distro is not "correcting mistakes" or the distro introduces them knowingly, then yes, "it's probably going to stay like that", regardless of any user complaints - if any other user (or the distro maintainers) dismiss the suspicions, because "... but the tool is so useful", the distro has no incentive to correct that mistake - the distro has more incentive to please it's users or itself instead

That's not what the bug report said. They asked me to prove that nonfree software was in some of these distributions.

without oversight, the FSDG reduces to the honor system - the responsibility of auditing software, is upon each user, rather than the distro maintainers (exactly as if there was no FSDG) - in that case, people may as well use debian or any other distro, and apply the FSDG to it themselves, per their own judgment

I don't understand the connection here. Are you assuming that none of the distribution follow the FSDG and most importantly that none try to correct mistakes?

I disagree with the fact that users are alone here. People do work collectively on FSDG compliant distributions. And users can either bugreport or fix things and either propose the fix to the distribution (through a patch) or push it directly if they are able and allowed to.

With distributions that don't take these bug reports into account, users are indeed alone.

If instead you agree that it's not possible to review everything proactively because there is too much code, and that instead we must ban suspicious software, not everybody has the same definition of suspicious, so unless people regroup and add somehow add policies that enable to work together, people would be alone there.

Here the issue would be to manage to express a policy with words that can easily be understood.

Something like "Should Parabola remove suspicious software" is way too subjective.

Though if you manage to find a way to express that, we could put that to the discussion like I suggested in my previous comment (with "Should Parabola stay FSDG compliant? [...]").

I don't see chromium mentioned in the guidelines¹, so maybe you meant that chromium is incompatible with the FSDG. And so far I didn't find any rationale for that.

no, i meant literally: "chromium is explicitly forbidden, by name" - i assumed that you were aware of that - donald made it so in 2018, and the only distro which distributed chromium (pureos) removed it shortly afterward - i wrote about that on the parabola dev list, just a few months ago

No I was not aware of it. Do you have links to it?

the chromium web browser is on the "List of software that does not respect the Free System Distribution Guidelines" wiki page⁴ - the presence of any software on that list, prevents any distro from being endorsed by the FSF, if it distributes any of the software on that list, without applying an accepted liberation procedure; and puts the ongoing endorsement in jeopardy for any distro which later accepts any of them un-treated - currently, the only accepted liberation procedure for chromium is: "use icecat"

[1]:

https://www.libreplanet.org/wiki/List_of_software_that_does_not_respect_the_Free_System_Distribution_Guidelines#chromium-browser

There is some issue with that rationale. First anyone can put programs on that list despite the warning at the beginning of that page.

And in fact I've edited it myself last year without going through the procedure, precisely because the procedure didn't work. If I recall well I sent suggestions to the gnu linux libre mailing list and nothing happened, so I ended up doing the work directly. If I recall well people also did complain that this page was outdated and so on.

Ideally I'd like to edit that page and change the procedure on it.

Then there is a deeper issue. For me that list isn't something authoritative we necessarily have to follow.

The entries have a Recommended Fix, and I don't see why we should blindly follow it.

If I take a game (chromium-bsu), the Recommended Fix is

Debian, at least, has permission to distribute it under the Clarified Artistic License since version 0.9.14.1. Distribute that (or something based on it) and you're fine.

An alternative which is perfectly fine as per the FSDG would be to remove that package, and here I don't see why removing the package would be forbidden.

As for chromium the Recommended Fix is "Remove program/package", but you could also fix it.

Now it mentions two problems:

- (1) Copyright or license of some code is unclear
- (2) Links to proprietary plugins.

Here I assume that at least (2) is not present in Guix.

As for (1) I explained that it might actually be allowed by the FSDG in my previous comments. If it's not allowed then we must have a wider discussion about it due to the repercussions about such rule.

Even a very narrow rule could have wide repercussions: Replicant doesn't have packages for instance, so you cannot have a rule that only deals with packages.

the explicit requirement is the checklist item: "Programs commonly known to have freedom issues are liberated or excluded"

There is indeed an "OR" here.

https://libreplanet.org/wiki/Template:FSDG_Checklist

that checklist is the complete criteria for the "Endorsement Process"; and donald confirmed that all criteria are binding perpetually, for all FSDG distros

https://libreplanet.org/wiki/Incoming_distros#Endorsement_Process

that is a reasonably solid "QED", IMHO - i specifically asked donald to confirm that, to remove all doubt

Is there a link to that?

My question is also if Parabola is really the place where we should decide if a given distribution is FSDG fit or not.

it is not; and i would rather not discuss that - but we are responsible for vouching for the software in third-party repos - we must avoid steering users toward any software on the FSDG "ugly list", which is not treated according to the accepted recommendation

So I suggest that in Parabola we trust that distributions that are on the list of FSDG compliant distributions are actually FSDG compliant?

That doesn't prevent to work upstream to actually get other distributions removed or added to that list.

That also doesn't prevent Parabola to forbid certain third party package managers on other grounds.

So if I didn't remove pip we still had to deal with the issue.

we do still need to deal with the issue, WRT the dozens of other TPPMs - we can not justify blacklisting only one - they all need to stay or go together, in a single commit, all with this ticket as the bug reference; because any liberation procedure could be expressed generically for all of them (whichever one from the table in the OP)

then, we actually need to continue identifying more of them, which are not yet listed on this ticket - these are only the obvious, widely-known ones

Exactly, I'll remove rubygems as soon as I find the time.

- What policies did or will Parabola take that is more strict than the FSDG.

"init-freedom" and "free-culture" are more strict than the FSDG

I'm aware that we also have non-systemd inits like openRC, but I didn't hear of a free culture policy in Parabola. I'll need to find infos on that.

Are they optional for contributors? Or do we need some policy about them? Or do we just need more awareness about them ?

What happens if we hold Parabola to the same standard?

all software distributors should be held to that standard - if anyone contests some software in the parabola repos, we should be willing and able to defend it to some reasonable degree; or we should patch it, or remove it - that is the entire point of the FSDG - if users must prove which distro packages are non-free, they could use any distro, and prove that to themselves only, without complaining to anyone

Here I was talking about the fact that Parabola uses Arch Linux repositories without reviewing them first. Like with chromium, the review is done after the fact.

indeed, we could hold icecat and iceweasel up to the same light; though no one ever has - if people expressed reasonable doubts about those, then parabola should not distribute them, unless we, or the FSDG work-group could prove that the suspicions are unfounded

but the fact today, is that the FSDG recommendation is "use icecat - not chromium" - unless that changes, parabola is meeting that standard of defensibility

If the idea is to use reasonable doubt, it's very different from just doubt. There is a part of subjectivity in it, but you can also argue rationally about it.

So here it means that we would deal with that through discussions which is fine per me. And that's exactly what we are doing with Guix and Guix's chromium here.

Though for me if the goal is to understand if Guix is FSDG compliant, that discussion has to go upstream at some point.

Else it would really be strange if the FSDG compliance of a distribution was decided unilaterally in an obscure Parabola bugreport.

And assuming that we are right or wrong about it without upstream discussions would also be really problematic.

But it's also perfectly fine to start discussing it here, as it could help clarifying our thought before asking a question about that upstream.

edit1: fix list formatting in Now it mentions two problems:

Denis.

#87 - 2022-08-04 03:25 PM - bill-auger

There is some issue with that rationale.

For me that list isn't something authoritative we necessarily have to follow.

The entries have a Recommended Fix, and I don't see why we should blindly follow it.

the list is authoritative; because the FSF licensing officer made it so - the FSF licensing officer is the only authority on the matter - if anyone disagree with that rationale, then someone should convince the FSF licensing officer to change that rule; but no one should ignore it

the FSDG endorsement process requires all distros to follow those recommendations - i believe that i have demonstrated this, beyond any doubt - after those (then) new rules were codified, i prodded donald to confirm publicly, that all distros are bound to those new rules perpetually; which he did

i wanted that to be absolutely clear from the onset, specifically to remove the doubt that you just expressed - at the time, that rule was invoked as the justification to convince pureos to remove chromium, which they did, closing a long standing freedom bug report on their bug tracker

because of that rule, it is impossible for any new distro to pass the evaluation, if it distributes chromium without a satisfying liberation treatment - at the time, the FSDG work-group was aware of the pureos freedom bug report, and was actively trying to find a liberation treatment, to replace the current recommendation - the liberation treatment (later) chosen by guix was considered; and it was decided to be necessary but insufficient - that all happened long before guix was even interested in that program - to this day, no one has offered a satisfying liberation treatment, which would change the recommendation

i would agree with any rules which are applied fairly to all; otherwise, i would disassociate myself from the organization, rather than imposing rules, with which i disagree, upon others, or breaking the rules myself - i strongly disagree with any rationale, which would permit some distros to distribute programs, which others distros are not permitted to distribute - but that is precisely the FSDG we have today; and i will not be a party to unfairness - we are hoping to correct that unfairness; not to encourage and perpetuate it

if distros decide for themselves, which rules to follow and which rules to ignore, then the endorsement is meaningless - i refuse to carry a meaningless banner - so if we keep guix and chromium, then we should resign from the FSDG, rather than joining those who have made a mockery of it

Exactly, I'll remove rubygems as soon as I find the time.

and i will remove guix - sounds like a plan - i think we both agree, that it is time to stop debating, and start *doing*

#88 - 2022-08-04 05:47 PM - gap

This isn't going to end well.

The brute-force solution of just blacklisting the offending TPPMs is going to have a knock-on effect on the packages which depend on the TPPMs, which will either become uninstalleable or have to be blacklisted in turn, which is a lot of work.

If this discussion has caused a healthy surge of motivation, then I suggest we redirect that towards cooperating with the other FSDG-compliant distros, or if we really want to take immediate action, to the temporary solution of patching out the upstream repo URLs and packaging the TPPMs in the libre repo.

Personally I use cargo (Rust) and cabal and stack (Haskell), amongst the rare use of other TPPMs, by doing the best I can to make sure all the packages in the dependency graph for what I am trying to install are libre.

Without them in the Parabola repos, I would have to install them from elsewhere because I need to use them anyway, and all that would achieve is being extremely inconvenient.

With them kept in Parabola but neutered, this makes the fact that the users of the TPPMs have to manually input the URLs to the repos they use and go to extra lengths to retain their freedom explicit, which is a better temporary solution than having to go completely out of your way to even install and keep them updated.

I use Parabola in part so that I don't have to use the AUR or other repos for pacman, or manually install and update everything like on the dreaded proprietary OS named after a transparent looking-glass.

#89 - 2022-08-04 10:27 PM - GNUtoo

the list is authoritative; because the FSF licensing officer made it so - the FSF licensing officer is the only authority on the matter - if anyone disagree with that rationale, then someone should convince the FSF licensing officer to change that rule; but no one should ignore it

What I meant is that while the list and the problems are authoritative (I trust you on that), there is only a **Recommended Fix**.

So if I understood right:

- The Recommended fix is a valid fix
- Not fixing is not an option.

So if I understood well it can also be fixed in another way.

the FSDG endorsement process requires all distros to follow those recommendations - i believe that i have demonstrated this, beyond any doubt - after those (then) new rules were codified, i prodded donald to confirm publicly, that all distros are bound to those new rules perpetually; which he did

That makes no sense to me, it would mean that we can only fix by using the recommendation?

It would not be a recommendation then.

i wanted that to be absolutely clear from the onset, specifically to remove the doubt that you just expressed - at the time, that rule was invoked as the justification to convince pureos to remove chromium, which they did, closing a long standing freedom bug report on their bug tracker

I'm very open to various courses of actions but I badly need information here.

If Guix is really not compliant I need the info about it so I can make the case about it to Guix (in a bug report) and get that fixed in Guix.

I cannot simply say to Guix, the FSF or the GNU Linux libre that bill auger said so, so it must be true.

I need facts that I can use to get that issue solved, and just removing Guix in Parabola isn't going to solve the issue at all.

Guix is also used in Replicant for instance, so if it's not compliant we also need to get rid of it.

And having Guix considered non-compliant in Parabola and compliant in Replicant, that would make no sense.

If that's the case we should also stop advocating for Guix, etc.

And right now I only have this bug report to show to Guix and the FSF so it's not going to work.

So where do I find about that?

Do I need to ask publically on the mailing list and to the FSF and GNU Linux libre if Guix's chromium is FSDG compliant?

If so I can do it and then we'd remove Guix as it's removed from the list of FSDG compliant distributions.

because of that rule, it is impossible for any new distro to pass the evaluation, if it distributes chromium without a satisfying liberation treatment - at the time, the FSDG work-group was aware of the pureos freedom bug report, and was actively trying to find a liberation treatment, to replace the current recommendation - the liberation treatment (later) chosen by guix was considered; and it was decided to be necessary but insufficient - that all happened long before guix was even interested in that program - to this day, no one has offered a satisfying liberation treatment, which would change the recommendation

Do you have links about that?

Was a liberation treatment similar to the actual one considered as well?

i strongly disagree with any rationale, which would permit some distros to distribute programs, which others distros are not permitted to distribute -

Same here

but that is precisely the FSDG we have today; and i will not be a party to unfairness - we are hoping to correct that unfairness; not to encourage and perpetuate it

Do you have something that I can verify?

if distros decide for themselves, which rules to follow and which rules to ignore, then the endorsement is meaningless - i refuse to carry a meaningless banner - so if we keep guix and chromium, then we should resign from the FSDG, rather than joining those who have made a mockery of it

I also agree with that.

i will remove guix

Under what rationale?

The issue here is that we have a big consistency issue that we need to resolve. And we cannot limit that issue to Parabola.

Guix is still listed in the list of FSDG compliant distributions and so far you didn't give me any information that would enable me to take care of the problem with various upstream projects.

Maybe that information is not public, in that case I could write

a mail to the GNU Linux libre mailing list to find out and try as hard as I can to get back some consistency.

We cannot have GNU consider Guix compliant, and Parabola considering it not compliant. That doesn't work.

For instance if you consider Guix compliant and that I don't we'd end up in reverting each other's commit because for me it's compliant on <https://www.gnu.org/distros/> and for you it's not compliant because of chromium.

So just removing it without any other rationale does not work.

So in practice you could probably manage to somehow revoke my commit access but I think the better course of action would be to get that fixed upstream. That is absolutely crucial.

If you want to exclude it now without waiting for upstream, you would need to find a good excuse for that, and not being FSDG compliant doesn't work because it has a consistency issue here.

We also have the option to put rules to the discussion on the Parabola mailing list. A rule that could be put to discussion there right now could be "Should Parabola forbid all third party package managers".

That rule would have no consistency issue and people could debate about its pros and the cons and we could reach a collective decision.

An argument in favor of that rule could be to avoid a war in Parabola. Though we also need to solve the underlying issue as for instance we cannot have the FSF say that Guix is not compliant (I've no idea if they thought or said that, I'm trying to find out asking questions here) while also having GNU list Guix as compliant.

edit1: Fix "Not fixing is not an option"

edit2: fix "So if I understood well it can also be fixed in another way."

edit3: Replaced last paragraph that was unclear with

"We also have the option to put rules to the discussion on the Parabola mailing list. [...]"

Denis.

#90 - 2022-08-04 11:46 PM - GNUtoo

no, i meant literally: "chromium is explicitly forbidden, by name" - i assumed that you were aware of that - donald made it so in 2018, and the only distro which distributed chromium (pureos) removed it shortly afterward - i wrote about that on the parabola dev list, just a few months ago

I've found [a mail by Donald](#) which gives more information on the usage of [List_of_software_that_does_not_respect_the_Free_System_Distribution_Guidelines](#)

He said that:

Something being on the list

[List_of_software_that_does_not_respect_the_Free_System_Distribution_Guidelines] isn't an automatic ban for a distro wanting to include it. You can view it as an initial nonfree report that needs to be addressed.

But if at the end of the day, something is determined to not meet the guidelines, then that will have to be removed, or the distro can't maintain it's promise. So no endorsed distros will be able to make a different decision, though obviously as members of the list they are involved in making that determination.

So the argument that Guix is not compliant because it has Chromium and that Chromium is in the [List_of_software_that_does_not_respect_the_Free_System_Distribution_Guidelines](#) becomes more complicated to make, but that doesn't mean it's

invalidated either.

Now if you want to go this route because you think it's important to remove Guix only with the rationale that it's not FSDG compliant (and not because of other rationales), then the following could work: Someone (you or me if I stumble upon it) could find the proof that it was not compliant at some point, and we'd show that the diff is small enough between before and now and we'd bug report that to Guix to see what happens. If they refuse we could take it to the GNU Linux Libre mailing list again and finally get that resolved for good (this way I could go back at fixing other FSDG compliance issues and reviewing patches to get more people help with maintaining Parabola).

If that can help somehow, I've some patches in Guix so I am able to make small patches to fix things if needed, add packages, etc.

edit1: Removed "I've no idea of the context, but" in I've no idea of the context, but I've found [...] because I read a bit of the thread now.

edit2: Replaced last sentence (was: So we need to understand if the actual Guix's chromium meet the guidelines.) by what's after the quote of Something being on the list".

Denis.

#91 - 2022-08-05 01:49 AM - GNUtoo

Another interesting thing to look into would be to understand if Chromium has incompatible licenses by actually looking for that.

After a quick look I found a potential issue that we'd need to confirm.

Guix's Chromium bundles speech dispatcher.

The official speech dispatcher source code has GPLv2+ only code (in src/server/output.c, src/server/sem_functions.c, etc). And Guix's Chromium also has code under the APSL 2.0 license which is "incompatible with the GPL"[1].

We cannot be sure yet that there is an issue because:

- The APSL page doesn't mention the GPL version, and the APSL 2.0 was released in 2003 and the GPLv3 in 2007. So a question remains: if the GPLv2+ is upgraded to the AGPLv3+ or to the GPLv3+, would it be compatible with the APSL 2.0?
- We also need to prove that these GPLv2+ files are actually used. Chromium might only use LGPLv2.1+ files.

References:

¹<https://www.gnu.org/philosophy/apsl.html>

Denis.

#92 - 2022-08-05 02:27 AM - GNUtoo

gap wrote:

This isn't going to end well.

The brute-force solution of just blacklisting the offending TPPMs is going to have a knock-on effect on the packages which depend on the TPPMs, which will either become uninstalleable or have to be blacklisted in turn, which is a lot of work.

Do you have example of that? I assumed that precisely because they are meant to install third party packages, no Parabola packages depended on them. Though I didn't verify my assumption.

In any case you have a good point as some could also be used to build software as well and so end up in the build dependencies of some packages.

If the default maven (parabola has its own fork) uses maven-central, it would be a good example of that.

If this discussion has caused a healthy surge of motivation, then I suggest we redirect that towards cooperating with the other FSDG-compliant distros, or if we really want to take immediate action, to the temporary solution of patching out the upstream repo URLs and packaging the TPPMs in the libre repo.

We do need to do that at some point indeed. Though I cannot fix what is in Parabola by contributing to Guix and vice versa for instance. But we can start adding documentation on the Libreplanet wiki at least. And if we work on repos, we can try to do something that is as reusable as possible by other distributions.

But at the end we also need to modify (Parabola) packages.

Personally I use cargo (Rust) and cabal and stack (Haskell), amongst the rare use of other TPPMs, by doing the best I can to make sure all the packages in the dependency graph for what I am trying to install are libre.

What I'm concerned about first is to be able to take decisions that are consistent and rationale. This is to make sure that this doesn't end up in some commit revert war and that people don't end up burning out or becoming crazy, and that we could go back to normal. Though the normal situation is already not that great as there is too much work to be done for too few contributors.

Once we have the proper procedure in place to decide something, we could then start taking decisions in a more calm way.

And ideally I'd prefer not to remove third party package manager that are not incompatible with the FSDG, but my top priority is to create and/or restore a good context to be able to take decisions.

Without them in the Parabola repos, I would have to install them from elsewhere because I need to use them anyway, and all that would achieve is being extremely inconvenient.

With them kept in Parabola but neutered, this makes the fact that the users of the TPPMs have to manually input the URLs to

Here you assume that it would be easier to just add a repository URL than installing a programming language package manager on top of Parabola. I've no idea if it's actually the case or even if it depends on the specific programming language package manager or not.

But it's definitely an option. And this option is compliant with the FSDG so I've nothing against it.

It's just that removing known problematic packages is faster. The fact that the package is removed also doesn't mean that it has to stay that way. Parabola needs a lot of help for fixing freedom bugs, so people that want to could also help with research to try to see if there is a way to remove the repository URL and that it is still easy for users to do some configuration to add (potentially other) repository URLs.

For users In some cases it could be easier to just install other package managers on top of Parabola. Some can at least be installed through some curl <https://some-address.sh> | bash, though that trusts the certificate authorities, so some users might not like that at all. Other can probably be checked with gpg before running the installation script.

Personally I installed the tor-browser in this way and for me it was not very hard (especially because it's well documented). Of course I made sure not to install any nonfree addons on top (and in fact installing any addons on that browser is strongly discouraged because it can make it lose its anonymity feature). There might also have been a lot of work made in their documentation, though can that also be reused for other things.

edit1: Changed "Of course make sure not to install" into "Of course I made sure not to install"

Denis.

#93 - 2022-08-05 04:52 AM - bill-auger

note, i did not read any of today's comments yet - i believe that i demonstrated what i intended to though, and it had nothing to do with guix specifically

from my perspective, this is what i see

gnutoo is going to a great length to retain that one example, guix - while oaken-source feels just as strongly about retaining pip, which gnutoo was very easy to discard

that is the real reason why i pushed this button so vigorously - i wanted to highlight that the disagreement is only a matter of favoritism, and i have no favorites - i am disagreeing only about treating *any* one of them differently than the others

i am indifferent to all of them - IMHO all TPPMs are equally ugly, regardless of which software they install - we, not anyone else, should be packaging software for parabola users - we do not really need any of these; so any time that i spend discussing it, is time better spent on something else

if we get rid of pip, oaken-source will object - if we get rid of guix, gnutoo will object - i will object, only if we get rid of some, but not all others

so i say, lets just do the simplest thing, and get rid of them all now (including any which are FSDG-fit already) - anyone who feels so strongly about recovering any of them, that person can patch it, and argue for the FSDG-fitness of the treatment - or just wait to see if anyone else complains about any of them

im sure that i could patch guix to suppress chromium, with very little effort - if i wanted to use that program, i would simply patch it, rather than to waste a single moment defending it - of all TPPMs, that treatment would probably be the easiest one of the lot - just do it, or drop the subject please

we will get absolutely nowhere with the FSDG work-group - that would be "putting the cart before the horse" - unfortunately, that horse died 3 years ago - (not coincidentally) it was this problem with guix specifically, which killed it - the FSDG work-group has 4 serious ailments, which would need to be cured, in order to revive it - resolving this chromium problem, is one of those - the process will not work in the normal way; because the machinery is currently broken, and the captain is asleep at the wheel

#94 - 2022-08-05 05:49 AM - GNUtoo

bill-auger wrote:

note, i did not read any of today's comments yet - i believe that i demonstrated what i intended to though, and it had nothing to do with guix specifically

from my perspective, this is what i see

gnutoo is going to a great length to retain that one example, guix - while oaken-source feels just as strongly about retaining pip, which gnutoo was very easy to discard

Here you're really missinterpenetrating what I said. I didn't defend Guix per se, I was just trying to make sure that whatever decision we take, that the decision would be consistent and rationale.

So I made serious propositions to go forward where the outcome is that Guix is removed, but in a way that is consistent and rationale.

I even tried to do licenses review to actually find an excuse to get chromium removed in Guix so this mess would end.

And if I let the situation become worse we would have something called horizontal hostility which is extremely dangerous for any communities.

So I was looking for any way out really, where the community of FSDG compliant distribution would not explode in a war, and the people would be kept sane.

that is the real reason why i pushed this button so vigorously - i wanted to highlight that the disagreement is only a matter of favoritism, and i have no favorites - i am disagreeing only about treating *any* one of them differently than the others

That only made me misinterpret what you said.

I thought that you were completely irrational and that you were trying to get some vengeance against Guix because you could not stand that they shipped Chromium while we had to patch many programs not to use Chromium.

This kind of reaction is normal with certain situations. For instance if you have a group of people that assemble and that decide to form a community, and that stealing is considered wrong, and that the people who steal benefit from it at the expense of the ones that don't, then if too much people steal, it would start dislocating the group.

The fact that you were trying to get vengeance made me sad and angry because we're all in the same camp and I really don't want a war. I've already too much things to care about.

And note that pushing people already under big pressure can be very dangerous (more on that below).

i am indifferent to all of them - IMHO all TPPMs are equally ugly, regardless of which software they install - we, not anyone else, should be packaging software for parabola users - we do not really need any of these; so any time that i spend discussing it, is time better spent on something else

if we get rid of pip, oaken-source will object - if we get rid of guix, gnutoo will object - i will object, only if we get rid of some, but not all others

I will not object if you can give me a good reason to do it.

So far the reason you gave me had huge repercussions. It meant that Parabola would not have the same culture than the rest of the FSDG compliant distributions, GNU and the FSF combined.

Like Parabola would be in a parallel world with different facts. And weather or not these facts are true is not the question here. The question is to be on the same universe and discuss the facts and come to a common conclusion where the decision is the same everywhere.

We cannot have downstream project unilaterally decide that another distribution is not FSDG compliant. This needs to be done upstream (so in the GNU Linux Libre mailing list and the page with the distributions needs to be updated on the gnu.org website).

Very recently, I had to deal with a very similar issue where someone was trying to force a French community ISP to take actions against the FSFE. Here too a community ISP is not the place where to decide something like that. If the FSFE has issue, there is an article about the FSFE on the English and the French Wikipedia, so people can discuss about facts there. People can also organize to make pressure so the issues are fixed. But a tiny French ISP is not the right place for that, especially because, the FSFE is European and the ISP is French. The discussions would at least need to happen in English for instance (and potentially be translated locally).

This way again it fixes the problem upstream and for everybody.

I also can't stand horizontal hostility: basically people that should fight together (here against nonfree software) end up fighting each other.

so i say, lets just do the simplest thing, and get rid of them all now (including any which are FSDG-fit already) - anyone who feels so strongly about recovering any of them, that person can patch it, and argue for the FSDG-fitness of the treatment - or just wait to see if anyone else complains about any of them

Just give me a good excuse for that (in the commit message for instance) and I'd be ok with it.

Denis.

#95 - 2022-08-05 05:53 AM - GNUtoo

Another thing that what made me think that you were not rationale was that I made clear proposition and you kept talking about the fact that Guix was not FSDG compliant. That is not rationale.

#96 - 2022-08-05 06:17 AM - GNUtoo

Also, that question was not answered and was probably lost in the stream:

"init-freedom" and "free-culture" are more strict than the FSDG

I'm aware that we also have non-systemd inits like openRC, but I didn't hear of a free culture policy in Parabola. I'll need to find infos on that.

Are they optional for contributors? Or do we need some policy about them? Or do we just need more awareness about them ?

I really need to know about it as I often add packages (and at least once the package had nonfree data).

Denis.

#97 - 2022-08-05 06:29 AM - bill-auger

And if I let the situation become worse

it is not going to get worse; because that already happened 3 years ago, and nothing has changed since - the critical time to "not let it get worse" was december 2018

but there is no "war" that i am aware of - there is simply no cross-distro cooperation happening - the FSDG work-group has been effectively defunct since then

gnutoo is going to a great length to retain that one example

Here you're really missinterpenetrating what I said.

you have written about a megabyte to this ticket in the past 2 days, specifically defending the value of guix - i was not interpreting *what* you wrote; but the sheer volume of it

i have no desire for vengeance - removing guix from parabola would not satisfy that anyway - my position is entirely rational - chromium is forbidden by the FSDG; so we must take steps to avoid making it available - thats all i care to discuss - guix only happens to be the specific impediment; where pip is a generic impediment

but, just to avoid any confusion WRT my opinion:

!!! GUIX IS NOT FSDG-COMPLIANT !!!

- the presence of chromium in guix is in direct conflict with an explicit FSDG requirement
- any new distro would be rejected for that same reason
- chromium was added by guix core maintainers, in full knowledge that it was forbidden
- when asked to help the work-group to determine if their chosen treatment was FSDG-fit, the guix maintainers declined to participate

now which part of that do you want to defend?

- the hypocrisy of the FSDG?
- or the blatant disregard of the guidelines by the guix maintainers?
- or their refusal to redeem themselves, by helping to resolve the conflict?

if you are not trying to defend either of those ugly facts, then let us not discuss guix or the FSDG any further - we have work to do

that discussion would be relevant, only if we were trying to rescue the reputation of the FSDG and the work-group - i believe that ship has sailed long ago, unfortunately - i can only protect the reputation of parabola now

#98 - 2022-08-05 06:42 AM - GNUtoo

And if I let the situation become worse

it is not going to get worse; because that already happened 3 years ago, and nothing has changed since - the critical time to "not let it get worse" was december 2018

I was not talking about the Chromium situation here but simply hostility.

but there is no "war" that i am aware of - there is simply no cross-distro cooperation happening - the FSDG work-group has been effectively defunct since then

you have written about a megabyte to this ticket in the past 2 days, specifically defending the value of guix - i was not interpreting what you

wrote; but the sheer volume of it

Please do read that megabyte and maybe you'll understand what has been going on.

but, just to avoid any confusion WRT my opinion:

!!! GUIX IS NOT FSDG-COMPLIANT !!!

To be clear, I'm only pointing that this should be solved **upstream**. Parabola can't decide that unilaterally. But we can make upstream accept that fact.

now which part of that do you want to defend?

None. Please actually read what I wrote. Maybe you'll understand.

the hypocrisy of the FSDG?
or the blatant disregard of the guidelines by the guix maintainers?
or their refusal to redeem themselves, by helping to resolve the conflict?

None of that.

if you are not trying to defend either of those ugly facts, then let us not discuss guix or the FSDG any further - i have work to do

Then I resign.

DO NOT PUSH PEOPLE LIKE THAT!!!!

#99 - 2022-08-05 06:52 AM - GNUtoo

I've now resigned: <https://git.parabola.nu/hackers.git/commit/?id=71059ce4e93cb772baec40be7d90852b95e709f7>

I do not have commit access anymore.

Denis.

#100 - 2022-08-05 06:54 AM - bill-auger

i hold no hostility towards anyone - i just dont like wasting time - parabola is a demanding mistress - it needs essential technical work every day - if i spent so much time discussing the FSDG, or tending to freedom bugs, parabola would cease to exist in a short time - i would irresponsible to let that happen; and there would no longer be any distro to discuss - we definitely need more help to address these issues - i can not be so much involved in these long discussions, as a simple matter of priorities

#101 - 2022-08-05 06:55 AM - GNUtoo

i have no hostility - i just dont like wasting time - parabola is a demanding mistress - it needs constant technical work to keep it running at all - if i spent so much time discussing the FSDG, or tending to freedom bugs, parabola would cease to exist in a short time - we definitely need more help to address these issues - i would be irresponsible to spend any significant amount of time on them

It's not about the FSDG anymore here, it's about understanding that pushing people is not a good thing.

If that's something you understand, then I need that commit reverted somehow, and I want to write the commit message though.

Denis.

#102 - 2022-08-05 06:59 AM - GNUtoo

And to summarize what happened here, that I kept discussing that issue so long precisely because the health of the free software community is important.

What we could do for the commit you will not push people again like that is that you'd revert and I'd force push changing the reverted commit.

Denis.

#103 - 2022-08-05 07:04 AM - GNUtoo

*What we could do for the commit is that if you will not push people again like that, is that you'd revert it and I'd force push changing the reverted commit.

#104 - 2022-08-05 07:25 AM - GNUtoo

I finally managed to really talk to Bill Auger again on IRC and it turned out that because he didn't read what I wrote here (which is long) we got misunderstandings on top of misunderstanding.

Denis.

#105 - 2022-08-05 07:55 AM - nona

I have been watching this discussion. I think that

- GNUtoo wants the links (or means of communication) to show that Guix is not FSDG-compliant
- bill-auger wants equity among software (remove pip only if all the rest is also removed)

It seems that removing pip is not going to solve that situation. It also seems as if we can keep the situation as it is for a couple of days, when people can go back, have a pleasant time and come back with a fresh mind.

Does everyone agree?

#106 - 2022-08-05 08:21 AM - GNUtoo

What I want is more subtle than that. I don't want downstream FSDG distributions to decide that other FSDG distributions listed on the <https://www.gnu.org/distros/free-distros.html> page are not FSDG compliant.

If something like that is to be decided, it should involve some process that is done upstream, either through the GNU Linux Libre mailing list, or through GNU, or the FSF etc. Since that GNU Linux Libre mailing list is public, any Parabola user or contributor can also raise the issue there. The opposite is also true (anyone can comment on this bug too) but bringing the discussion here would be really weird when it should instead happen in the GNU Linux Libre mailing list.

I even had a mail ready for asking the question there but I waited to see if we could resolve this here before.

I had bill-auger and the gnu linux libre mailing list in the To fields.

Subject: Is (the current) Guix('s Chromium) FSDG compliant?

Date: Fri, 5 Aug 2022 02:06:11 +0200

Hi,

In Parabola we started to look into third party package managers.

We found that at least python-pip and rubygems were incompatible with the FSDG because their repositories had no commitments to only have free software[1]. I've also started to make a Libreplanet article about the bigger issue of external repositories[2].

During that discussion, Bill Auger argued that the actual Guix's chromium is not FSDG compliant because Guix's liberation procedure for Chromium is not good enough.

And I cannot make my own opinion on that topic because I lack information. And we badly need to resolve that issue because it's now a cause of big tensions in Parabola.

And having different views on whether Guix is compliant or not is not a good idea: If that becomes the case then contributors contributor A could remove packages enabling to install or use Guix on the rationale that the liberation procedure is not good enough, and contributor B (like me for instance) could add it back because Guix is fine as it's listed on the <https://www.gnu.org/distros/free-distros.html> page.

So we need to resolve that issue in some ways.

And I looked in the source code of the Guix Chromium package, and the third party code still used by Guix's chromium is listed[5] but I've no idea how to check if that list is complete myself, so it doesn't help me much here.

And since I was not involved much in the discussions, finding response to the questions below myself would be really time consuming, and I might not even find the response at all in this mailing list.

Questions:

- Has the actual liberation procedure been considered as not good enough by the FSF? Is there any public information on that? At what time did the review happen[4]?

- Is Guix still FSDG compliant or not?

References:

[1]<https://labs.parabola.nu/issues/1035>

[2]<https://libreplanet.org/wiki/Group:Software/research/ExternalRepositories>

[3]<https://www.gnu.org/distros/free-distros.html>

[4]If it happened in the past, I could still look if it's very different or not with git.

[5]<https://git.savannah.gnu.org/cgit/guix.git/tree/gnu/packages/chromium.scm?id=a5a88b0248f3ca70db30605c0c2aa93bb360d8ad#n76>

Denis.

That discrepancy between Parabola and GNU would also means that other FSDG distributions that do use Guix would also automatically end up in the same situation. One contributor could either trust Parabola or GNU to add or remove support for Guix.

So in that case it is preferable to avoid that situation completely and directly bring the question (again) to the GNU Linux Libre mailing list instead of taking a decision like that.

PS: Everyone is free to take the mail and modify it or not and post it to the GNU Linux Libre mailing list if that helps.

Denis.

#107 - 2022-08-05 08:22 AM - GNUtoo

As for the tension, it's now resolved as me and bill-auger cleared at least our most important misunderstandings.

#108 - 2022-08-05 09:52 AM - nona

(spell-checking):

could even be made optional where users would choose whether or not to use it.
And having different views on whether

whether (related to meteorology) should be whether (grammatical element)

contributors contributor A

should be "contributor A"

hope it helps :) .

#109 - 2022-08-05 03:18 PM - bill-auger

Do I need to ask publically on the mailing list and to the FSF and GNU Linux libre if Guix's chromium is FSDG compliant?

yes; but you will not get an answer - i have asked that question repeatedly over the years

the first time i asked, donald promised an answer - that was march 2018 - no answer ever came

<https://lists.nongnu.org/archive/html/gnu-linux-libre/2018-03/msg00080.html>

i asked again in 2019 - no answer

<https://lists.nongnu.org/archive/html/gnu-linux-libre/2019-10/msg00012.html>

since then, there has accumulated four major problems with the FSDG - i and others have asked about them, multiple times - no answer was given to any of those questions either

take the most recent one for example - in early 2021, donald asked distros to freeze nmap, ending the message with:

"and let me know if you have any questions."

a month later, i asked about resolving the nmap issue - no answer

<https://lists.nongnu.org/archive/html/gnu-linux-libre/2021-03/msg00000.html>

then i asked again - no answer

<https://lists.nongnu.org/archive/html/gnu-linux-libre/2021-04/msg00042.html>

then i asked again - no answer

<https://lists.nongnu.org/archive/html/gnu-linux-libre/2021-07/msg00000.html>

that time, i also opened a formal ticket with the licensing office - no answer

then i complained about all of these things again a few months later - no answer

<https://lists.nongnu.org/archive/html/gnu-linux-libre/2021-10/msg00012.html>

i still have not gotten so much as one single word of acknowledgement, that i ever asked about nmap

i discussed this with the new FSF licensing officer, a few weeks ago in IRC, and i literally begged him to acknowledge the nmap ticket, even if he had no good answer to give, but simply to acknowledge the fact that the question was asked - he has not done so

so forgive me, if i seem pessimistic about asking the FSF again - lets see if you get an answer about that apple license, and use that as a guage, to determine if there is any hope of involving the FSF in this larger issue

Do you have links about that?

<https://lists.nongnu.org/archive/html/gnu-linux-libre/2016-12/msg00001.html>

IIRC, that was luke from hyperbola, who had asked those chromium forks, about any licensing treatments they applied; and none of them did

Was a liberation treatment similar to the actual one considered as well?

yes, as far as i know it is exactly the same treatment - it is "ungoogled" chromium - as far the FSDG work-group is concerned, that treatment does no liberation whatsoever - it addresses only privacy concerns - the "ungoogling" was deemed to be necessary (because of the "no anti-features" requirement) but insufficient (because it's maintainers admittedly did nothing to address licensing concerns)

He said that:

But if at the end of the day, something is determined to not meet the guidelines, then that will have to be removed, or the distro can't maintain it's promise. So no endorsed distros will be able to make a different decision, though obviously as members of the list they are involved in making that determination.

ok, but here is the subtle problem with that statement - how long is "the day"?

if the code-base takes 10 years to audit, and no one ever starts doing so, then that "day" will never end - the work-group could never declare: "sorry, but the end of the day has arrived, and you failed to liberate that program satisfactorily - so *now* you must remove it" - that is effectively giving a free pass to any enormous code-bases like chromium, indefinitely; because "the end of the day" will never arrive

i think the key point donald made there, is that no single distro can decide what an acceptable liberation treatment is - they must be "involved in making that determination" as "members of the list" - the guix maintainers made that determination only for themselves, while their own users were objecting strongly, and declined to be involved with the work-group to form a consensus - IMHO, that was "the end of the day"; and no liberation procedure was submitted or accepted

Now if you want to go this route because you think it's important to remove Guix only with the rationale that it's not FSDG compliant

we do not need even to invoke the FSDG directly - chromium and webengine are already on the parabola blacklist - there are many items which are on the blacklist, only because "depends on blacklisted webengine" or "recommends whatever" - the blacklist reason could simply be "guix distributes the blacklisted chromium and webengine", with the chromium ticket as the bug reference, instead of this one - that would be enough to justify removal from the blacklist, if and when chromium and webengine get removed from the blacklist, or if chromium and webengine get removed from the guix

#110 - 2022-08-05 03:38 PM - GNUtoo

Now that the misunderstanding has been cleared, we can go on with the discussions.

For the record we have support for installing many more distributions that are in the list of FSDG compliant distributions:

Package(s)	Distribution(s)	Distribution support status
libre/hyperbola-archlinux-keyring libre/hyperbola-keyring libre/hyperbola-mirrorlist libre/hyperbola-pacstrap-config	Hyperbola	OK, won't support HyperbolaBSD
libre/debootstrap	Gnewsense (3.0, 4.0)	Gnewsense 3.0 and 4.0 probably unmaintained

	PureOS (amber, awen, green, landing)	OK
	Trisquel (2.0 -> 9.0)	Trisquel 9 still supported, need to add Trisquel 10
pcr/guix	Guix 1.1	Package should be either removed or updated
pcr-guix-installer	Guix 1.3	OK

I also need to disclose a part of my work in Parabola has actually been to work on these packages.

The goal was not only to enable to install any FSDG compliant distribution from any (FSDG compliant) distribution, but it was also to make sure that the signatures are checked automatically along the way.

So I cannot not care at all about the ability to bootstrap other distributions since I actually spent time to do the work in an organized way.

But I also care way more about the whole community of people working on the FSDG compliant distributions than these packages.

Also, that feature is not Guix specific. And currently I care way more about debootstrap than Guix¹ because I need Trisquel to be bootstrapable from other distributions for Replicant. I even sent patches for that upstream² to enable people to more easily debootstrap Trisquel.

The status of this work is documented here: <https://libreplanet.org/wiki/Group:Software/research/CrossDistroBootstrap> though it only mention distributions that are in the list of FSDG compliant distributions.

I also have work in progress for adding support in Guix for adding pacstrap with Parabola and Hyperbola support.

So if all that has to go, I would at least want to have a rationale discussion about it, looking at pros and cons and at possible alternatives that would work for me and other users that do want or need to install other distributions from Parabola.

And for me the two mailing lists looks a good place for that (the mail could put both mailing lists in copy).

And note that what I care about is the feature (make it easy to install other distributions in the list of FSDG compliant distributions in a way that is secure), not the actual packages. The packages is just one way to achieve that. There might be other ways to achieve the same thing. Having discussions about that could also help me finding these ways.

¹ If someone can review [my patches for debootstrap](#) it would be great.

#111 - 2022-08-05 03:56 PM - GNUtoo

bill-auger wrote:

Do I need to ask publically on the mailing list and to the FSF and GNU Linux libre if Guix's chromium is FSDG compliant?

yes; but you will not get an answer - i have asked that question repeatedly over the years

In this case it might be more fruitful to pursue a different strategy and try to get the problem fixed by proving that there is conflicting licenses in Chromium.

We could also try to apply that strategy directly on the upstream project, but we'd need a broader coalition for that.

We would first need to identify where the real upstream is (maybe it's chrome?).

Then we could try to see if there is some way for us to get it under the GPLv2 and prove that there is a GPLv2 violation and never reinstate their right to distribute the code whose license has been violated. We would need help from people who have significant contributions in the projects that have this GPLv2 code though.

The advantage here is that it would solve completely unrelated issues that are important. If Google is not able anymore to make its own browser, it would probably have less control over the web standards. The collateral damage would obviously be the free software projects that somehow depend on Chrome but the benefits would probably outweigh the damage in several order of magnitude.

The question here would also be to try to predict other collateral damage. If it ends up with a huge war of license violation lawsuits it would probably not be good either.

Denis.

#112 - 2022-08-05 03:56 PM - bill-auger

gap wrote:

The brute-force solution of just blacklisting the offending TPPMs is going to have a knock-on effect on the packages which depend on the TPPMs, which will either become uninstalleable or have to be blacklisted in turn, which is a lot of work.

it is true - most of them would not be very disruptive to remove but pip is - pip alone, has 10 runtime dependents and 84 build dependents - so at least 94 other packages need to be blacklisted, and all of their dependents, and just hope that none in the chain are part of the core system

#113 - 2022-08-05 04:19 PM - GNUtoo

How can we find that information?

On a computer that needs some updates, I tried to find the info with python-pip:

```
$ pacman -Q -i python-pip
```

So if it's installed I do get the information in Required By : python-pipenv, but that is not very useful because it only shows which installed packages depend on python-pip and I probably can't install all packages on my system (some might conflict I guess).

And if I look with -S instead that information is completely gone:

```
$ pacman -S -i python-pip
```

Denis.

#114 - 2022-08-05 04:28 PM - bill-auger

the website shows it https://www.parabola.nu/packages/extra/x86_64/python-pip/

you can get the dependents with `pacman -Sii` - i dont think pacman can show the build-time dependents though

#115 - 2022-08-05 04:45 PM - bill-auger

why do you suppose that those bootstrap installers would need to go? - pureos would be the only other with chromium and webengine - im pretty sure that hyperbola and trisquel never had those

anyway, the bootstrap installer would not install that GUI stuff - i assume those are clean of anything dubious at the first order; and that we can ignore what may happen inside the chroot at the second order

in other words, arch-installer is probably FSDG-fit; because everything in an arch basic system is libre - i dont think we need to be concerned, that inside the arch chroot, its pacman or AUR helper may be used to install something non-free, as long as that is done intentionally by the user

#116 - 2022-08-05 06:05 PM - bill-auger

some notes on that letter

We found that at least python-pip and rubygems were incompatible with the FSDG

we found 5 already with dubious or no licensing requirements: CPAN, PECL, python, ruby, rust (comment # 3)

During that discussion, Bill Auger argued that the actual Guix's chromium is not FSDG compliant because Guix's liberation procedure for Chromium is not good enough.

that makes it sound like my personal opinion - i was explaining the consensus among the work-group - this would be more accurate:

During that discussion, Bill Auger argued that Guix's chromium is not FSDG compliant, because the variant of Chromium packaged by guix, was reviewed by the FSDG work-group, a year previously, and the privacy-related changes made by that variant, were considered to be necessary but insufficient for FSDG-fitness; and that Guix's never demonstrated any liberation procedure to supplement the upstream fork's modifications. For that reason, the only known liberation procedure for Chromium recognized and recommended by the FSDG, is still "Remove program. Use Icecat".

Guix's actual implementation was not reviewed; because guix never submitted it for review - i dont know what the guix procedure actually is - they told us, that if we wanted to know exactly what they did, we would need to read their guile script - what little they did explain, did not include anything convincing - it only confirmed what had already been done among the work group previously

but AFAIK, there probably is no "Guix's liberation procedure", as you wrote that - im pretty sure they simply build "ungoogled chromium" as-is - so "the procedure", is whatever the ungoogled team does; and that is what the work-group reviewed

#117 - 2022-08-06 12:59 PM - gap

The brute-force solution of just blacklisting the offending TPPMs is going to have a knock-on effect on the packages which depend on the TPPMs, which will either become uninstalleable or have to be blacklisted in turn, which is a lot of work.

To clarify this statement, this isn't the entire issue per se.

The problem is that this solution is just temporary, so all the work being done will have to be undone as soon as we switch to a more sustainable temporary solution, which as I mentioned previously, I think should be to keep the TPPMs, but patch out the URLs to the offending repos and move them to the libre repo.

When someone inevitably says something like, "Parabola is painful to use for development because it has no TPPMs, and you have to download and install them manually by wget-ing an install script and then piping it into bash, which is inconvenient and insecure," the response will have to be something like, "Yes, we made it that way instead of implementing a better solution."

#118 - 2022-08-06 05:39 PM - wael

gap: what would the "better solution" look like though?

This is a fundamental issue with TPPMs, which gets worse when applied to an FSDG repo like Parabola.

Other than the purely technical aspects which have been discussed already, I think there is also the user trust and social contract Parabola has with said users.

Distro maintainers serve not only to package software and test it, but also to vet it for working properly and not containing malware...etc

TPPMs do nothing of that at times, and with the attacks on TPPM like PIP lately I think if someone really needs it for development reasons they can download it and work with it outside the context of the official repos, otherwise we'd be putting a majority of users at increased risk for little benefit.

#119 - 2022-08-06 10:13 PM - gap

See comment # 64.

As for the other issues you raise, I agree the lack of vetting of third-party repos is a flaw with them, and the two "proper" solutions I list in comment # 64 should fix that.

In general, I agree TPPMs are a bad idea, and that we should all use a universal standard package format, but for the time being, TPPMs are the only practical way to develop software in most programming languages, and a blanket removal will only drive devs to manually installing them, which is inconvenient and insecure.

I think most devs are just going to give up at that point and switch back to Arch; many people value their freedom and work around the inconvenience, such as myself, but this issue might just push fence-sitters back to Arch.

This issue has already caused disagreement amongst people committed to freedom.

Imagine how fence-sitters are going to perceive a blanket removal of all TPPMs.

We should do our best to show all the benefits of freedom, and one way we can do that is by exercising all four freedoms in order to modify these TPPMs and publish our modified versions in order to implement the solution of patching out upstream URLs.

That would only be slightly less convenient than TPPMs with no modifications, but it would be FSDG-compliant and make it explicit that these TPRs are not recommended or endorsed by Parabola.

#120 - 2022-08-07 03:48 AM - bill-auger

"Parabola is painful to use for development because it has no TPPMs, and you have to download and install them manually by wget-ing an install script and then piping it into bash, which is inconvenient and insecure,"

firstly, convenience is not among Parabola's stated goals - it is primarily a do-it-yourself distro catering to "power-users" - the goal is to provide total software freedom; and no one should expect any sort of freedom to be convenient - freedom requires responsibility and diligence; and that is often a "painful" concession

distros exist to shoulder most of that responsibility and diligence on behalf of users; and LTS distros do that fundamentally better than rolling distros - but that is not why TPPMs exist - most TPPMs exist, only to help people ignore responsibility and diligence

secondly, no one would suggest that `wget` is a good way to install software - to install third-party software responsibly, one would get the source-ball *and* its signature, verify the signature, un-tar it, verify its licensing, and compile it, all manually - that is definitely less convenient than `pip install what-ever and i-dont-care-what-else` - but that is why distros exist - a well-maintained PKGBUILD does all those chores for the user, with a simple `makepkg -sri`; and the distro devs DO care about licensing and minimizing the 'whats' and 'what-elses' - that is why you will find very few java or golang programs in distros

lastly, anyone who would complain that their favorite TPPM was missing from parabola, would very likely complain the same, if the TPPM was patched to filter out improperly-licensed packages - probably most python, ruby, and javascript programs depend on at least one dubious package

TPPMs are the only practical way to develop software in most programming languages

that is not generally true - java and golang software is even worse - their build systems download all those 'what-ers' from github or 'where-ever', often un-versioned

the only reason that is most practical for some languages, is because of the absurd number of dependencies their devs pile on, like penny candy from a candy shop - people who develop native software are not nearly so glib about adding dependencies - most often, the only non-standard dependencies are from a single well-licensed toolkit like QT - so, that is not an argument for using TPPMs - it is an argument for lazy engineering and a lack of concern for licensing

libre-minded people should not be developing software which requires hundreds of un-vetted dependencies (possibly un-versioned too), with unknown licenses, packaged by hundreds of people (most of whom are anonymous); then encouraging all of their users to have the same lack of concern for licensing and supply chain trust - it is irresponsible, and it works against software freedom - that doesnt mean that parabola must prevent it; but it would be a very low priority

Imagine how fence-sitters are going to perceive a blanket removal of all TPPMs.

i would like to imagine:

- they would realize that TPPMs encourage bad engineering and licensing practices
- and that no one is accountable for the software in those repos
- and that TPPMs prevent parabola from keeping its promise to ensure software freedom
- and that TPPMs exist only because windows and mac lack proper package management
- they would conclude that we can write more efficient and trustable software; because we have system-level package management, and native compiled languages
- they would decide to write humble and stable KISS software, which lightens the burden of ensuring software freedom, for both the developer and users who want that

#121 - 2022-08-07 03:56 PM - gap

My point is that we don't always have to accept the choice between freedom and convenience; we can have both if we put in the work. We don't have to reject TPPMs entirely; Hackage and Stackage (at least I think as far as we have checked(?)) are GNU FSDG-compliant TPRs, and there's no reason we couldn't convince other TPRs to be compliant too, thus liberating all uses of the TPPM with the TPR, whether it is used on Parabola or any other system.

I agree with you, and I'd like to imagine that scenario happens too, but for the time being, we essentially have the option between wholeheartedly rejecting TPPMs, thus driving fence-sitters away, or embracing TPPMs, and finding a solution to make them work. If we reject TPPMs, we cripple Parabola by default for the development of the languages which are intimately tied to them.

Also, many TPPM dependencies are not huge frameworks like Qt, but are tiny libraries which do a very specific job, as languages nowadays are designed with very minimalistic standard libraries, which rely on external libraries to do basic tasks. The downsides we all know, but the upsides are that the language need not be burdened with an aging standard library, and more innovation can come from competition between libraries. Therefore, comparing native, system-wide packages to TPPM libraries isn't always an apples-to-apples comparison, because they accomplish different tasks.

Overhauling the way we write software, and package programs for different systems, won't happen overnight, and I think a wholehearted rejection of TPPMs would be perceived more as a way of giving up than an attempt to help improve the situation or invent and implement a better solution.

#122 - 2022-08-08 11:39 AM - bill-auger

the options considered should not be seen as prescriptions or rules - they are only plans for what to do now - regardless of which plan is chosen now, there is always the possibility of doing something else later

My point is that we don't always have to accept the choice between freedom and convenience; we can have both if we put in the work.

i agree - "we" (that is everyone) do not need to compromise always; but someone always does - that liberation work is an inconvenience to the person who does the work

freedom and convenience are not mutually exclusive - you could have one or the other, or both, or neither - but they each have their costs - at some point, someone must pay those costs - with commercial software, people pay the cost for convenience in the form of the money; but they usually are not paying for freedom - so, they pay a hidden cost in the form of that sacrifice - even with freeware and freebie web services, users pay the price with their privacy or by having their CPUs hijacked, or by reading ads

with free software there are no contracts or market forces at work, no offers, no expectations, no obligations, and there is no natural progression toward improvement or desire for universal customer satisfaction - developers generally pay all of the costs for everyone else, purely of their own volition and at their own pace

so, just because something is possible or is a good idea, that is

not enough to make it actually happen - even if some people want very much for it to happen, that still is not enough - if someone wants to do something, it is probably possible - if no one wants to do that something, it will not be done - its really that simple

so WRT this issue, during the six years this ticket has been open, no one has yet volunteered to patch any of these programs, or to curate libre-only repos, or to try convincing the upstreams to adopt libre licensing policies - that does not mean those are bad ideas or that they should not be done - it only means that no one who believes that these things should be done, actually wants to do them

by deleting these packages, it is essentially forcing parabola users to become aware of this bug report - if that is not enough to motivate volunteers, then maybe it never was such a big problem after all - it would be great if all FSDG distros would do it at the same time - in that way, the problem would benefit from a larger pool of potential contributors who may be compelled to volunteer

#123 - 2022-08-14 10:08 AM - GNUtoo

bill-auger wrote:

why do you suppose that those bootstrap installers would need to go? - pureos would be the only other with chromium and webengine - im pretty sure that hyperbola and trisquel never had those

I'm a bit lost here. I assumed that there was some will to remove all third party package managers and that people would be able to find good arguments for that (which would be then discussed on the mailing list).

If the criteria is to ban repositories with "webengine", then we'd also need a good argumentation for it because we could as well ban repositories with known freedom bugs even if they're FSDG compliant. A Replicant installer would also need to be banned for instance.

And a way to do it would be to ban repositories with policies that are different from the ones in place in Parabola, on the basis that users might expect the exact same policies in other distributions but I don't find this argument compelling enough because it might be better to inform users of that in https://wiki.parabola.nu/How_does_Parabola_protects_users_against_nonfree_software (especially because some of the issues mentioned there are way worse than this one).

So we have several choices:

What to ban	Examples of banned software/repos	Comments
third party package managers	Guix, debootstrap, npm, pip, etc	Can work
programming languages repositories	npm, pip, etc	Issue: Why are programming language special?
non-FSDG compliant repositories	npm, pip, debian main, ubuntu, etc	
Repositories that contain nonfree software	npm, pip, etc	Mandatory as per the FSDG
<include additional ideas here>		

In any of these cases we need to have good reasons to ban theses.

in other words, arch-installer is probably FSDG-fit; because everything in an arch basic system is libre

Arch Linux probably contains nonfree software in at least their Linux packages. Debian has found the following files that are not DFSG compliant:

- Documentation/netlabel/draft-ietf-cipso-ipsecurity-01.txt
- arch/powerpc/platforms/8xx/micropatch.c
- drivers/media/usb/dvb-usb/af9005-script.h
- drivers/media/i2c/vs6624.c
- drivers/net/appletalk/cops*
- drivers/video/fbdev/nvidia
- drivers/video/fbdev/riva

In arch/powerpc/platforms/8xx/micropatch.c there is some data that really looks like code:

```
static uint patch_2000[] __initdata = {
0XXXXXXXXXX, 0XXXXXXXXXX, 0XXXXXXXXXX, 0XXXXXXXXXX,
};
```

And here we lack the corresponding source code of that code.

If Debian based distributions also reuse the work of Debian to exclude non DFSG compliant files, they're more likely to only have free software.

#124 - 2022-08-14 04:06 PM - bill-auger

Issue #1035 has been updated by GNUtoo.
bill-auger wrote:

why do you suppose that those bootstrap installers would need to go?

I assumed that there was some will to remove all third party package managers and that people would be able to find good arguments for that

not all necessarily - the ones which have libre licensing policies could stay - haskell and debian are the only ones that come to mind though - as long as debootstrap will install packages only from debian main, it is probably acceptable - the ubuntu target probably would not be though, because the kernel is not de-blobbed

if debootstrap is not restricted debian main by default, then debootstrap would need patching - it could pull from pureos and trisquel instead; so that one would probably be rather easy to rescue

i am only wanting to hold them all to a single standard - if there are packages in some repo that is known to be a FSDG issue, and the maintainers have no intention of removing them, then that is enough reason

chromium and webengine are only well-known examples - those on the parabola blacklist; because we believe that they have FSDG problems - i would not want anything on the blacklist for any other reason - thats why there are multiple blacklists - the others are optional, for non-FSDG issues such as extra privacy

If the criteria is to ban repositories with "webengine", then we'd also need a good argumentation for it because we could as well ban repositories with known freedom bugs even if they're FSDG compliant. A Replicant installer would also need to be banned for instance.

i would not use "known freedom bugs" as the criteria - parabola has plenty of those too - i would just consider if someone were to report an issue to the repo maintainers, which the FSDG would consider to be a freedom bug, would the maintainers out-right refuse to comply

if the replicant repos have known freedom bugs, which are retained intentionally, then i would conclude that it is a non-free repo - there is not likely any grey area in that criteria - some repos will refuse to remove unlicensed packages; because their policy (or lack of) permits them

So we have several choices:

What to ban	Examples of banned software/repos	Comments
third party package managers	Guix, debootstrap, npm, pip, etc	Can work
programming languages repositories	npm, pip, etc	Issue: Why are programming language special?
non-FSDG compliant repositories	npm, pip, debian main, ubuntu, etc	
Repositories that contain nonfree software	npm, pip, etc	Mandatory as per the FSDG
<include additional ideas here>		

In any of these cases we need to have good reasons to ban theses.

i dont see any real distinction in any of those rows - the reason is the same for all

programming language specific repos are not special - those are third party package managers - the only problem with any of those, is that they pull packages from "non-FSDG compliant repositories", "Repositories that contain nonfree software" so they all roll into one and the same concern

we can not ban the repos - the only "recommended" interface to those repos are their package managers - all we can do is to patch the package managers in every case in that table

in other words, arch-installer is probably FSDG-fit; because everything in an arch basic system is libre

Arch Linux probably contains nonfree software in at least their Linux packages. Debian has found the following files that are not DFSG compliant:

yer right - i really was not thinking clearly - the arch linux would disqualify arch-installer

hopefully, debian will remove anything that does not meet the DFSG though

#125 - 2022-09-12 01:04 AM - bill-auger

WRT pip, it looks like it's absence would not be very problematic for parabola - it is required by only one package - it is a non-essential package - it appears to be used only to parse a manifest of python requirements from the source code, to be used to populate the PKGBUILD depends array

```
$ sudo parabola-dependents -v python-pip
updating database ....
querying database ....
searching abslibre ....
compiling results for (13) dependents ....
```

```
makedependents dependents:
  libre/parabolaweb-utils
```

```
community/home-assistant      (ignored)
community/httpie              (ignored)
community/python-ansi2html    (ignored)
community/filebin             (ignored)
community/python-hatch        (ignored)
community/python-pip-api       (ignored)
community/python-pip-run       (ignored)
community/python-pip-shims     (ignored)
community/python-pipenv        (ignored)
community/python-pipenv-to-requirements (ignored)
community/python-pwntools      (ignored)
community/python-requirementslib (ignored)
```

#126 - 2022-09-13 03:30 AM - bill-auger

- Related to Bug #3338: [freecad]: depends on blacklisted 'qt5-webengine' and blacklisted 'python-pip' added

#127 - 2022-09-13 07:25 AM - bill-auger

just to add, that i un-webengine'd 'freecad' today, and noticed that webengine is apparently used, only for a runtime "addons" downloader, which i also removed

so it deserves an special mention, as the first to cross both [#1167](#) and [#1035](#), and remarkably, both per the same feature

#128 - 2022-09-24 10:48 AM - bill-auger

i was hoping that oaken-source would add this - we discussed it briefly on IRC; and he made at least one good point, which is worth considering (and tossed 'virtualenv' on the barbecue)

```
<bill-auger> i think it is either doing too little, or too much - i am mostly indifferent to which strategy is decided, but whichever that is, i want to apply the same strategy uniformly to all of them
<oaken-source> actually, you'd have to remove virtualenv as well, because virtualenv will pull a recent pip into the environment
<oaken-source> but please don't do that, because thats an incredibly useful tool that I use daily for work
<bill-auger> well the problem now is that the change is in blacklist.git - it will either need to be reverted or accepted before any new your-freedom is made
<bill-auger> my point is that we would need to remove  npm, rubygens, guix, nuget, cpan, pecl, flatpack, appimage, docker, and a dozen others also, if the rationale is to be defended sincerely
<oaken-source> and curl, wget and probably even pacman
```

<oaken-source> maybe there's a way to draw a rational line in the sand, and come up with measurable, objective criteria, but I don't think we're quite there yet
<bill-auger> thats different - curl and pacman will not locate non-free software for you
<oaken-source> so any change to that regard we're doing right now can only be blind activism
<oaken-source> pip won't either
<oaken-source> the search function was removed
<bill-auger> look at the OP of [#1035](#) - i made a table of all the options
<bill-auger> i was not aware of that - if the search function was removed, there may be no problem
<bill-auger> the search feature is the only FSDG problem IMHO
<bill-auger> without the search feature , i think any of those could be exempt from the [#1035](#) issue
<bill-auger> OTOH, it would still pull in non-free deps without warning - so that is still technically not FSDG-fit
<oaken-source> \$ pip search nonfree
<oaken-source> ERROR: XMLRPC request failed [code: -32500]
<oaken-source> RuntimeError: PyPI's XMLRPC API is currently disabled due to unmanageable load and will be deprecated in the near future. See <https://status.python.org/> for more information.
<oaken-source> that's maybe worth another point in your table
<oaken-source> 'make sure that no non-free dependencies can be pulled in without user confirmation'
<bill-auger> well, the way the FSDG is worded, we must not direct users to any repo which does not have a libre licensing policy
<oaken-source> this is the least well defined rule of the fsdg, and it's been bugging me for a long time
<bill-auger> im thinking maybe the FSDG could be re-worded more leniently - strictly worded, it does not allow client-side filtering of such a repo
<oaken-source> and arguably, it's not our job.
<oaken-source> I think our job ends where the user uses a program other than the package manager.
<bill-auger> well, unfortunately, *arguably* the FSDG workgroup is broken, so if its not our job, its no one's job
<oaken-source> I just think everything beyond that is an overly intrusive power-grab.
<oaken-source> it takes away choice from the user, which isn't what this should be about

#129 - 2022-09-29 01:48 PM - nona

a converter for PERL packages:

<https://github.com/jordansissel/fpm> (referenced on <https://github.com/anntzer/pypi2pkgbuild>)

#130 - 2022-12-22 05:55 AM - GNUtoo

Hi again,

I've (finally) asked on the dev mailing list specifically about the Parabola policies and its interaction with third party repositories¹.

This way it might reach more Parabola contributors, and it might also be more suited for policy decisions as this bug report also has lots of technical details on the work being done to fix this bug.

¹<https://lists.parabola.nu/pipermail/dev/2022-December/008402.html>

Denis.

#131 - 2023-01-08 09:50 PM - bill-auger

- Description updated

#132 - 2023-01-09 06:35 PM - GNUtoo

Now that this is clarified, I've started creating resources for cross distro collaboration on the Libreplanet wiki:

- [ExternalRepositories](#) : This page points to issues and bugreports in various FSDG compliant distributions.
- [Group:Software/FSDG_distributions](#) : This page lists distribution policies, to see if there are differences that results in situation that is a bug in one FSDG distribution and not in the other one.

#133 - 2023-02-01 03:45 AM - bill-auger

i tried to package a rust program today - AFAICT, rust programs typically expect to be online at build time, to fetch dependencies - there is a way around that, if you package all of its dependencies ahead of time, such as what the 'cargo-pkgbuild' would do - however, 'cargo-pkgbuild' also requires internet access at build time, to fetch it's dependencies (about 10 of them); and in the end, the program always crashes with a "file not found" error, which does not bother to mention which file was not found; so i had enough of that for one day

the dependencies appear to be source code though, and were compiled in the build() stage; so maybe this is not terrible - they would need to be moved into mksource() or the source=() array if possible

we could use it perhaps (if it ever can run without crashing) - but even with a packaging helper, it appears that this could be the routine PITA procedure, and maintaining any rust program properly may be a huge chore - hopefully there are not many of them

i must say though, that we could be backing ourselves into an uncomfortable corner by removing these TPPMs - some are so integrated with their language build system, that it may already be impossible to build all parabola packages on a parabola system - if it turns out that way, the result is only trading one FSDG conflict (unvetted repos) for another (not self-hosting) - its not obvious that this is the way to progress - some of these TPPMs may need to be liberated, not simply evicted

#134 - 2023-02-02 11:22 PM - bill-auger

- Related to Bug #3395: [ruby] Requires rubygems, which conflicts with your-freedom added

#135 - 2023-05-30 09:52 PM - bill-auger

- Description updated

as an update, ive seen it on the grapevine, that flatpack now has an option to exclude poorly-licensed packages - im not sure exactly to what degree (search results, a conflict mechanism, what are the criteria for exclusion, etc), or if the filter can be compiled (hard-coded) in, or if it is simply a CLI option or trivial checkbox in some GUI (ie: easily over-ridden, probably not the default behavior, etc) - but maybe there is something hopeful in that - at least those folks are considering this to be a significant issue now - thats more than can be said of most of them

#136 - 2023-06-22 04:17 PM - bill-auger

- Description updated

#137 - 2023-06-22 06:40 PM - bill-auger

- Description updated

#138 - 2023-06-22 06:42 PM - bill-auger

- Description updated

#139 - 2023-06-22 06:47 PM - bill-auger

- Description updated

#140 - 2023-06-22 08:43 PM - bill-auger

- Description updated

#141 - 2023-07-13 07:29 AM - bill-auger

- Description updated