

Packages - Freedom Issue #3422

Freedom Issue # 1035 (in progress): [your-system-sanity]: Non-Free Software From Third-party Package Managers (TPPM)

Docker configured to use a nonfree repository

2023-01-09 06:41 PM - GNUtoo

Status: confirmed	% Done: 0%
Priority: freedom issue	
Assignee:	
Category:	
Description	

History

#1 - 2023-01-09 07:00 PM - GNUtoo

- Parent task set to #1035

If I run the following command

```
docker run -it --rm archlinux bash -c "echo hello world"
```

It will contact the registry-1.docker.io. I've confirmed that with Wireshark.

The documentation also mentions that in several places. In the docker package source we have:
In `cli/man/src/image/pull.md`:

```
If you do not specify a `REGISTRY_HOST`, the command uses Docker's public registry located at `registry-1.docker.io` by default.
```

In `src/cli/docs/reference/commandline/tag.md`:

```
[...] If a hostname is present, it may optionally be followed by a port number in the format `:8080`. If not present, the command uses Docker's public registry located at `registry-1.docker.io` by default.
```

There were various attempt to upstream support to add a configuration to change the default repository. For instance the [Bug #7203](#).

This was not done due to several concerns. Some of the concern are described [in documentation about podman](#). Basically it would break many dockerfiles, and this way the default repository is also tried first, so there are security concerns as well.

So if we want to fix docker (or reintroduce a fixed version after having removed it) we could:

- (1) Remove the default repository in the source code.
- (2) Optionally fix the documentation too
- (3) Inform users that Parabola is not responsible for dockerfiles found on the Internet and inform that full path is needed for these dockerfiles to work. So with small modifications docker files will still work but Parabola won't be responsible for reviewing the ones that are FSDG compliant and the ones that aren't.
- (4) Inform that there is some research on Libreplanet, in the [Group:Software/research/ExternalRepositories](#) about finding fully free repositories, or at least FSDG compliant repositories
- (5) We could also inform users about ways to generate FSDG compliant docker images like in the [Group:Software/research/DistroExecutionEnvironments](#) article.

There might also be some docker images that are FSDG compliant in the default repository (like some images that don't use distributions like the busybox image) but that might be more complicated to review: if a repository is created with them it would probably need some work and maintenance. At worst we might be able to somehow create some wiki pages for the reviews though in case people want to review them.

edit1: Move the latest sentence in the list of things to do.

#2 - 2023-01-09 07:31 PM - GNUtoo

I've started looking into it, and I started working on a function to fix the issue:

```
prepare() {  
    # Remove the default repository (contains nonfree software)
```

```

sed 's/registry-1\.docker\.io//g' -i \
  cli/vendor/github.com/docker/docker/registry/config.go \
  moby/registry/config.go \
  ${sentinel}

sed 's/"registry-1\.docker\.io"/ns/g' -i \
  moby/vendor/github.com/containerd/containerd/remotes/docker/resolver.go \
  ${sentinel}

# Seems to only use the registry-1.docker.io repository
rm -f moby/contrib/download-frozen-image-v2.sh

# TODO: remove special cases
# moby/vendor/github.com/moby/buildkit/util/resolver/resolver.go

# TODO: Find an FSDG compliant repository and Use another example:
# moby/vendor/github.com/docker/distribution/registry/api/v2/descriptors.go
# moby/vendor/github.com/docker/distribution/registry/client/auth/session.go

# TODO: Documentation
# cli/docs/reference/commandline/tag.md
# cli/man/src/image/pull.md
# cli/man/src/image/tag.md
}

```

However even without modification the PKGBUILD doesn't build:

```

--> Making bundle: dynbinary (in bundles/dynbinary)
Building: bundles/dynbinary-daemon/dockerd-20.10.22
GOOS="" GOARCH="" GOARM=""
# github.com/docker/docker/daemon/graphdriver/btrfs
daemon/graphdriver/btrfs/btrfs.go:437:11: args.lim.max_referenced undefined (type _Ctype_struct_btrfs_qgroup_l
imit has no field or method max_referenced)
==> ERROR: A failure occurred in build().
Aborting...

```

So I've no idea what to do here.

Also if at some point it works again or that someone knows how to fix that compilation issue, I wonder if it's better to have a patch or use sed. A patch might require more maintenance but it's also easier to share with other FSDG compliant distributions. Guix might require a patch for that for instance.

edit1: Also ask about sed vs patch

#3 - 2023-01-09 07:45 PM - GNUtoo

- Priority changed from bug to freedom issue
- Tracker changed from Bug to Freedom Issue

Moved to freedom bug

#4 - 2023-01-09 07:48 PM - GNUtoo

- Status changed from unconfirmed to confirmed

Since:

- it uses index.docker.io
- index.docker.io has nonfree images: https://index.docker.io/_/microsoft-windows-base-os-images
- The nonfree images seems not to be an accident or a bug but be there on purpose

Then the bug is confirmed.