

.Debian

銀河系唯一のDebian専門誌

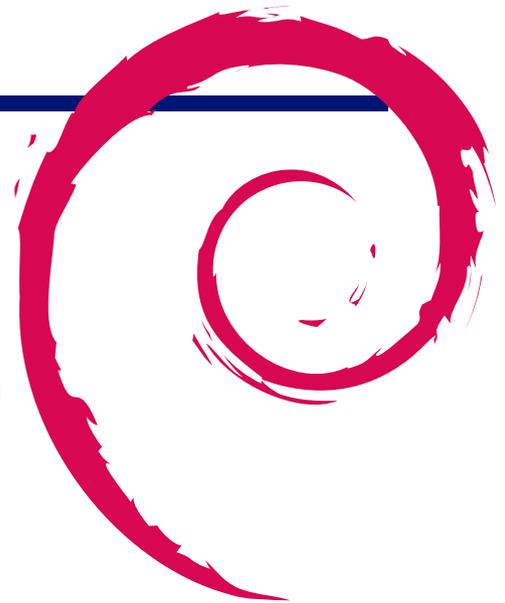
2021年7月17日

OSSGate&piuparts



1 最近の Debian 関連のミーティング報告

杉本 典充



1.1 2021 年 6 月度 東京エリア・関西合同 Debian 勉強会

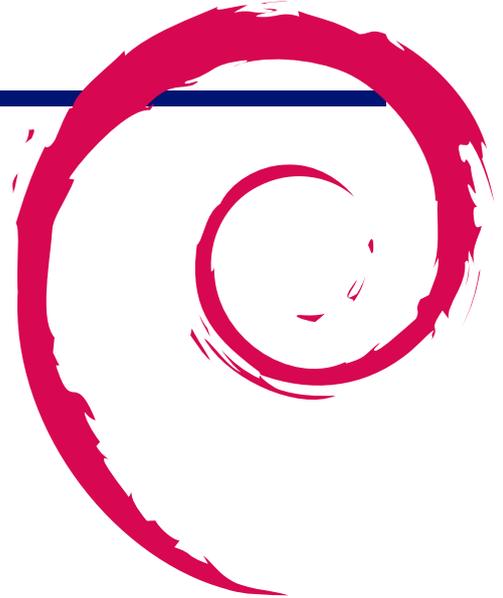
2021 年 6 月 19 日 (土) に東京エリア Debian 勉強会と関西 Debian 勉強会の合同でオンラインによる Debian 勉強会を開催しました。参加者は 6 名でした。

勉強会では、「bullseye のリリースノート日本語版の翻訳レビュー」を行いました。参加者からは様々な改善提案をいただきました。以下 URL のファイルにレビュー記録をまとめています。

https://tokyodebian-team.pages.debian.net/2021-06_tokyodebian_bof.txt

3 piuparts について調べてみた

杉本 典充



3.1 はじめに

Debian パッケージの作り方を学ぶことは多かったのですが、作ったパッケージをテストする方法の知識がありませんでした。

今回は作成した Debian パッケージのインストールテストを行うツールである piuparts について調べてみました。

3.2 piuparts と piuparts.debian.org

piuparts とは、Debian パッケージがインストール、アップグレード、アンインストールを正しく処理することをテストするツールです^{*1}。2021 年 7 月 11 日現在、Debian unstable において piuparts-1.1.3 の Debian パッケージを配布しています^{*2}。

piuparts パッケージの説明には「chroot 環境に最小の Debian をインストールした環境を作り、その chroot 環境でパッケージのインストールとアップグレードとアンインストールを行い、インストール前とアンインストール後のディレクトリの状態を比較して、パッケージの処理に問題があるか判定する」と記述があります。

Debian デベロッパー レファレンスの Appendix 「1 Debian メンテナツールの概要」に piuparts が紹介されており^{*3}、Debian パッケージを扱う場合には piuparts を知っておくとよいと記載されています。

また、Debian Project ではパッケージを自動でテストするシステム <https://piuparts.debian.org/> を運用しています。このサイトでは日々アップロードされる Debian パッケージを piuparts を使ってテストを実行し続けています。テスト結果は <https://piuparts.debian.org/> の web サイトを見ることができます。

3.3 piuparts を使ったパッケージのテスト

3.3.1 テストする hello パッケージのビルド

ここでは、unstable にある hello パッケージ <https://packages.debian.org/sid/hello> を参考にしてみます。まずは hello のソースパッケージをダウンロードして、debian パッケージを手元でビルドしてみます。

^{*1} <https://packages.debian.org/ja/sid/piuparts>

^{*2} Debian 10 buster では python2.7 で動くツールでしたが、Debian 11 bullseye では python3 に対応しました。

^{*3} <https://www.debian.org/doc/manuals/developers-reference/tools.ja.html#piuparts>

```
# apt-get update
# apt-get install build-essential
# apt-get build-dep hello
# apt-get source hello
# cd hello-2.10/
# dch
-> バージョンを 1 つ上げます
# debuild -us -us
# ls -l .. | grep deb
-rw-r--r-- 1 norimitu norimitu 36356 7月 11 04:53 hello-dbgSYM_2.10-2.1_amd64.deb
-rw-r--r-- 1 norimitu norimitu 6224 7月 11 04:53 hello_2.10-2.1.debian.tar.xz
-rw-r--r-- 1 norimitu norimitu 56488 7月 11 04:53 hello_2.10-2.1_amd64.deb
```

hello.2.10-2.1.amd64.deb パッケージができました。これ以降の記事では、piuparts コマンドを使ってこの Debian パッケージをテストしてみます。

3.3.2 piuparts の実行とログ

piuparts コマンドを使うには、piuparts パッケージをインストールします。

```
# apt-get install piuparts
# which piuparts
/usr/sbin/piuparts
```

piuparts コマンドを実行して hello パッケージのテストを実行します。"-l" オプションは、piuparts を実行したときに出力する標準出力をログファイルに追記するオプションです。

```
# piuparts hello_2.10-2.1_amd64.deb -l /tmp/piuparts_hello.log
```

最初はテストするパッケージのメタ情報を dpkg -info コマンドで出力します。

```
Om0.0s INFO: -----
Om0.0s INFO: To quickly glance what went wrong, scroll down to the bottom of this logfile.
Om0.0s INFO: FAQ available at https://wiki.debian.org/piuparts/FAQ
Om0.0s INFO: The FAQ also explains how to contact us in case you think piuparts is wrong.
Om0.0s INFO: -----
Om0.0s INFO: piuparts version 1.1.3 starting up.
Om0.0s INFO: Command line arguments: /usr/sbin/piuparts hello_2.10-2.1_amd64.deb -l /tmp/piuparts.log
Om0.0s INFO: Running on: Linux sid1 4.19.0-17-amd64 #1 SMP Debian 4.19.194-1 (2021-06-10) x86_64
Om0.0s DEBUG: Starting command: ['dpkg', '--info', 'hello_2.10-2.1_amd64.deb']
Om0.0s DUMP:
new Debian package, version 2.0.
size 56488 bytes: control archive=1872 bytes.
 758 bytes, 20 lines control
3601 bytes, 49 lines md5sums
Package: hello
Version: 2.10-2.1
Architecture: amd64
Maintainer: Santiago Vila <sanvila@debian.org>
Installed-Size: 280
Depends: libc6 (>= 2.14)
Conflicts: hello-traditional
Breaks: hello-debhelper (<< 2.9)
Replaces: hello-debhelper (<< 2.9), hello-traditional
Section: devel
Priority: optional
Homepage: http://www.gnu.org/software/hello/
Description: example package based on GNU hello
The GNU hello program produces a familiar, friendly greeting. It
allows non-programmers to use a classic computer science tool which
would otherwise be unavailable to them.
.
Seriously, though: this is an example of how to do a Debian package.
It is the Debian version of the GNU Project's 'hello world' program
(which is itself an example for the GNU Project).
Om0.0s DEBUG: Command ok: ['dpkg', '--info', 'hello_2.10-2.1_amd64.deb']
```

次に、debootstrap ('-no-merged-usr' オプションを指定) を実行してテストに使う chroot 環境を作ります。このとき、/proc や /dev などを bind mount しています。

```

Om0.0s DEBUG: Created temporary directory /tmp/tmpbcw125x_
Om0.0s DEBUG: Setting up minimal chroot for sid at /tmp/tmpbcw125x_
Om0.0s DEBUG: Starting command: ['debootstrap', '--variant=minbase',
'--keyring=/usr/share/keyrings/debian-archive-keyring.gpg', '--include=eatmydata',
'--no-merged-usr', '--components=main', 'sid', '/tmp/tmpbcw125x_', 'http://deb.debian.org/debian']
(debootstrap の標準出力は省略)
Om41.2s DEBUG: Command ok: ['debootstrap', '--variant=minbase',
'--keyring=/usr/share/keyrings/debian-archive-keyring.gpg', '--include=eatmydata', '--no-merged-usr',
'--components=main', 'sid', '/tmp/tmpbcw125x_', 'http://deb.debian.org/debian']
Om41.2s DEBUG: Starting command: ['mount', '-t', 'proc', 'proc', '/tmp/tmpbcw125x_/proc']
Om41.2s DEBUG: Command ok: ['mount', '-t', 'proc', 'proc', '/tmp/tmpbcw125x_/proc']
Om41.2s DEBUG: Starting command: ['mount', '-t', 'devpts',
'-o', 'newinstance,noexec,nosuid,gid=5,mode=0620,ptmxmode=0666', 'devpts', '/tmp/tmpbcw125x_/dev/pts']
Om41.2s DEBUG: Command ok: ['mount', '-t', 'devpts',
'-o', 'newinstance,noexec,nosuid,gid=5,mode=0620,ptmxmode=0666', 'devpts', '/tmp/tmpbcw125x_/dev/pts']
Om41.2s DEBUG: Starting command: ['mount', '-o', 'bind', '/tmp/tmpbcw125x_/dev/pts/ptmx',
'/tmp/tmpbcw125x_/dev/ptmx']
Om41.2s DEBUG: Command ok: ['mount', '-o', 'bind', '/tmp/tmpbcw125x_/dev/pts/ptmx', '/tmp/tmpbcw125x_/dev/ptmx']
Om41.2s DEBUG: Starting command: ['mount', '-o', 'bind', '/dev/pts/4', '/tmp/tmpbcw125x_/dev/console']
Om41.2s DEBUG: Command ok: ['mount', '-o', 'bind', '/dev/pts/4', '/tmp/tmpbcw125x_/dev/console']
Om41.2s DEBUG: Starting command: ['mount', '-t', 'tmpfs', '-o', 'size=65536k', 'tmpfs', '/tmp/tmpbcw125x_/dev/shm']
Om41.2s DEBUG: Command ok: ['mount', '-t', 'tmpfs', '-o', 'size=65536k', 'tmpfs', '/tmp/tmpbcw125x_/dev/shm']
Om41.2s DEBUG: sources.list:
deb http://deb.debian.org/debian sid main
Om41.2s DEBUG: Created policy-rc.d and chmodded it.
Om41.2s DEBUG: Created resolv.conf.

```

debootstrap で作成したディレクトリに chroot で入り、apt-get update を実行して試験対象のリポジトリのパッケージ情報を取得します。また、このタイミングで chroot のテスト環境のディレクトリ状態を保存しています。

```

Om41.2s DEBUG: Starting command: ['chroot', '/tmp/tmpbcw125x_', 'eatmydata', 'apt-get', 'update']
Om44.4s DUMP:
Hit:1 http://deb.debian.org/debian sid InRelease
Get:2 http://deb.debian.org/debian sid/main Translation-en [6509 kB]
Fetched 6509 kB in 2s (2650 kB/s)
Reading package lists...
Om44.4s DEBUG: Command ok: ['chroot', '/tmp/tmpbcw125x_', 'eatmydata', 'apt-get', 'update']
Om44.4s DEBUG: Starting command: ['chroot', '/tmp/tmpbcw125x_', 'eatmydata', 'sh', '-c', 'apt-cache dumpavail | md5sum']
Om44.5s DUMP:
c0117488ce119ec145559966f8a0cef8 -
Om44.5s DEBUG: Command ok: ['chroot', '/tmp/tmpbcw125x_', 'eatmydata', 'sh', '-c', 'apt-cache dumpavail | md5sum']
Om44.5s DEBUG: Starting command: ['chroot', '/tmp/tmpbcw125x_', 'eatmydata', 'apt-get', 'clean']
Om44.5s DEBUG: Command ok: ['chroot', '/tmp/tmpbcw125x_', 'eatmydata', 'apt-get', 'clean']
Om44.5s DEBUG: Recording chroot state

```

dpkg-query、dpkg-divert -list を実行してパッケージの情報や中身を調べています。

```

Om44.6s DEBUG: Starting command: ['chroot', '/tmp/tmpbcw125x_', 'eatmydata', 'dpkg-query', '-W',
'-f', '${Status}\t${binary:Package}\t${Package}\t${Version}\n']
(dpkg-query の結果は省略)
Om44.6s DEBUG: Command ok: ['chroot', '/tmp/tmpbcw125x_', 'eatmydata', 'dpkg-query', '-W',
'-f', '${Status}\t${binary:Package}\t${Package}\t${Version}\n']
Om44.6s DEBUG: Starting command: ['chroot', '/tmp/tmpbcw125x_', 'eatmydata', 'dpkg-divert', '--list']
Om44.6s DUMP:
diversion of /usr/share/man/man1/sh.1.gz to /usr/share/man/man1/sh.distrib.1.gz by dash
diversion of /bin/sh to /bin/sh.distrib by dash
Om44.6s DEBUG: Command ok: ['chroot', '/tmp/tmpbcw125x_', 'eatmydata', 'dpkg-divert', '--list']
Om44.6s INFO: apt-cache does not know about any of the requested packages
Om44.6s DEBUG: Starting command: ['lsdf', '-w', '+D', '/tmp/tmpbcw125x_']
Om44.8s DEBUG: Command failed (status=1), but ignoring error: ['lsdf', '-w', '+D', '/tmp/tmpbcw125x_']
Om45.3s DEBUG: No broken symlinks as far as we can find.
Om45.3s DEBUG: Starting command: ['lsdf', '-w', '+D', '/tmp/tmpbcw125x_']
Om45.4s DEBUG: Command failed (status=1), but ignoring error: ['lsdf', '-w', '+D', '/tmp/tmpbcw125x_']
Om45.9s DEBUG: No broken symlinks as far as we can find.
Om45.9s DEBUG: Starting command: ['chroot', '/tmp/tmpbcw125x_', 'eatmydata', 'dpkg-query', '-f', '${Version}\n', '-W', 'apt']
Om45.9s DUMP:
2.2.4
Om45.9s DEBUG: Command ok: ['chroot', '/tmp/tmpbcw125x_', 'eatmydata', 'dpkg-query', '-f', '${Version}\n', '-W', 'apt']

```

piupart コマンドの引数に指定した Debian パッケージを chroot のテスト環境内にコピーし、apt-get install コマンドを実行してコピーした Debian パッケージをインストールします。

```

Om45.9s DEBUG: Copying hello_2.10-2.1_amd64.deb to /tmp/tmpcwcw125x_/tmp
Om45.9s DEBUG: Starting command: ['chroot', '/tmp/tmpcwcw125x_', 'eatmydata', 'apt-get', '-y',
'--allow-downgrades', 'install', './tmp/hello_2.10-2.1_amd64.deb']
Om47.1s DUMP:
  Reading package lists...
  Building dependency tree...
  The following NEW packages will be installed:
  hello
  0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
  Need to get 0 B/56.5 kB of archives.
  After this operation, 287 kB of additional disk space will be used.
  Get:1 /tmp/hello_2.10-2.1_amd64.deb hello amd64 2.10-2.1 [56.5 kB]
  debconf: delaying package configuration, since apt-utils is not installed
  Selecting previously unselected package hello.

  Preparing to unpack /tmp/hello_2.10-2.1_amd64.deb ...
  Unpacking hello (2.10-2.1) ...
  Setting up hello (2.10-2.1) ...
Om47.1s DEBUG: Command ok: ['chroot', '/tmp/tmpcwcw125x_', 'eatmydata', 'apt-get', '-y',
'--allow-downgrades', 'install', './tmp/hello_2.10-2.1_amd64.deb']
Om47.1s DEBUG: Starting command: ['chroot', '/tmp/tmpcwcw125x_', 'eatmydata', 'dpkg-query',
'-f', '${Package} ${Status}\n', '-W', 'hello']
Om47.1s DUMP:
  hello install ok installed
Om47.1s DEBUG: Command ok: ['chroot', '/tmp/tmpcwcw125x_', 'eatmydata', 'dpkg-query',
'-f', '${Package} ${Status}\n', '-W', 'hello']
Om47.1s INFO: Installation of ['./tmp/hello_2.10-2.1_amd64.deb'] ok

```

chroot のテスト環境内のディレクトリを調べたり、インストールしているパッケージのバージョンを調べています。

```

Om47.1s DEBUG: Removing /tmp/tmpcwcw125x_/./tmp/hello_2.10-2.1_amd64.deb
Om47.1s DEBUG: Starting command: ['ls', '-w', '+D', '/tmp/tmpcwcw125x_']
Om47.3s DEBUG: Command failed (status=1), but ignoring error: ['ls', '-w', '+D', '/tmp/tmpcwcw125x_']
Om47.7s DEBUG: No broken symlinks as far as we can find.
Om48.0s DEBUG: Starting command: ['debsums', '--root', '/tmp/tmpcwcw125x_', '-ac', '--ignore-obsolete']
Om48.5s DEBUG: Command ok: ['debsums', '--root', '/tmp/tmpcwcw125x_', '-ac', '--ignore-obsolete']
Om48.5s DEBUG: Starting command: ['dpkg-query', '-f', '${Version}\n', '-W', 'adequate']
Om48.5s DUMP:
  0.15.6
Om48.5s DEBUG: Command ok: ['dpkg-query', '-f', '${Version}\n', '-W', 'adequate']
Om48.5s INFO: Running adequate version 0.15.6 now.
Om48.5s DEBUG: Starting command: ['adequate', '--root', '/tmp/tmpcwcw125x_', 'hello']
Om48.6s DEBUG: Command ok: ['adequate', '--root', '/tmp/tmpcwcw125x_', 'hello']
Om48.6s DEBUG: Starting command: ['chroot', '/tmp/tmpcwcw125x_', 'eatmydata', 'dpkg-query', '-W',
'-f', '${Status}\t${binary:Package}\t${Package}\t${Version}\n']
Om48.6s DEBUG: Command ok: ['chroot', '/tmp/tmpcwcw125x_', 'eatmydata', 'dpkg-query', '-W',
'-f', '${Status}\t${binary:Package}\t${Package}\t${Version}\n']

```

chroot のテスト環境内でインストールしたパッケージを apt-get remove を実行してアンインストールします。これで Debian パッケージのインストールからアンインストールするテストは終了です。

```

Om48.6s DEBUG: Starting command: ['chroot', '/tmp/tmpcwcw125x_', 'eatmydata', 'apt-get',
'remove', 'hello']
(省略)
Om49.1s DEBUG: Command ok: ['chroot', '/tmp/tmpcwcw125x_', 'eatmydata', 'apt-get', 'remove', 'hello']
Om49.1s DEBUG: Starting command: ['chroot', '/tmp/tmpcwcw125x_', 'eatmydata', 'dpkg', '--purge', 'hello']
Om49.2s DUMP:
  dpkg: warning: ignoring request to remove hello which isn't installed
Om49.2s DEBUG: Command ok: ['chroot', '/tmp/tmpcwcw125x_', 'eatmydata', 'dpkg', '--purge', 'hello']
Om49.2s DEBUG: Starting command: ['chroot', '/tmp/tmpcwcw125x_', 'eatmydata', 'dpkg', '--purge', '--pending']
Om49.2s DEBUG: Command ok: ['chroot', '/tmp/tmpcwcw125x_', 'eatmydata', 'dpkg', '--purge', '--pending']
Om49.2s DEBUG: Starting command: ['chroot', '/tmp/tmpcwcw125x_', 'eatmydata', 'dpkg', '--remove', '--pending']
Om49.2s DEBUG: Command ok: ['chroot', '/tmp/tmpcwcw125x_', 'eatmydata', 'dpkg', '--remove', '--pending']
Om49.2s DEBUG: Starting command: ['ls', '-w', '+D', '/tmp/tmpcwcw125x_']
Om49.4s DEBUG: Command failed (status=1), but ignoring error: ['ls', '-w', '+D', '/tmp/tmpcwcw125x_']
Om49.8s DEBUG: No broken symlinks as far as we can find.
Om49.8s DEBUG: Starting command: ['chroot', '/tmp/tmpcwcw125x_', 'eatmydata', 'dpkg-divert', '--list']
Om49.8s DUMP:
  diversion of /usr/share/man/man1/sh.1.gz to /usr/share/man/man1/sh.distrib.1.gz by dash
  diversion of /bin/sh to /bin/sh.distrib by dash
Om49.8s DEBUG: Command ok: ['chroot', '/tmp/tmpcwcw125x_', 'eatmydata', 'dpkg-divert', '--list']
Om49.8s DEBUG: Starting command: ['chroot', '/tmp/tmpcwcw125x_', 'eatmydata', 'apt-get', 'clean']
Om49.9s DEBUG: Command ok: ['chroot', '/tmp/tmpcwcw125x_', 'eatmydata', 'apt-get', 'clean']
Om49.9s DEBUG: Recording chroot state
Om50.0s INFO: PASS: Installation and purging test.

```

引き続き、インストール、アップグレード、アンインストールのテストを実行します。まずはリポジトリにある同名の hello パッケージを apt-get install でインストールします。

```

Om50.0s DEBUG: Starting command: ['chroot', '/tmp/tmpchcw125x_', 'eatmydata', 'apt-cache', 'show',
'-no-all-versions', 'hello']
Om50.9s DEBUG: Command ok: ['chroot', '/tmp/tmpchcw125x_', 'eatmydata', 'apt-cache', 'policy', 'hello']
Om50.9s DEBUG: Starting command: ['chroot', '/tmp/tmpchcw125x_', 'eatmydata', 'apt-get', '-y', 'install', 'hello']
Om51.5s DUMP:
  Reading package lists...
  Building dependency tree...
  Reading state information...
  The following NEW packages will be installed:
  hello
  0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Om51.5s DEBUG: Command ok: ['chroot', '/tmp/tmpchcw125x_', 'eatmydata', 'apt-get', '-y', 'install', 'hello']
Om51.5s DEBUG: Starting command: ['lsdf', '-w', '+D', '/tmp/tmpchcw125x_']
Om51.7s DEBUG: Command failed (status=1), but ignoring error: ['lsdf', '-w', '+D', '/tmp/tmpchcw125x_']
Om52.2s DEBUG: No broken symlinks as far as we can find.
Om52.2s DEBUG: Starting command: ['chroot', '/tmp/tmpchcw125x_', 'eatmydata', 'dpkg-query', '-f', '${Version}\n', '-W', 'apt']
Om52.2s DUMP:
  2.2.4
Om52.2s DEBUG: Command ok: ['chroot', '/tmp/tmpchcw125x_', 'eatmydata', 'dpkg-query', '-f', '${Version}\n', '-W', 'apt']

```

次に piuparts コマンドの引数に指定した Debian パッケージを chroot のテスト環境内にコピーし、apt-get install コマンドを実行してコピーしたパッケージでアップグレードします。

```

Om52.2s DEBUG: Copying hello_2.10-2.1_amd64.deb to /tmp/tmpchcw125x_/tmp
Om52.2s DEBUG: Starting command: ['chroot', '/tmp/tmpchcw125x_', 'eatmydata', 'apt-get', '-y',
'--allow-downgrades', 'install', './tmp/hello_2.10-2.1_amd64.deb']
Om52.8s DUMP:
  Reading package lists...
  Building dependency tree...
  Reading state information...
  The following packages will be upgraded:
  hello
  1 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
Om52.8s DEBUG: Command ok: ['chroot', '/tmp/tmpchcw125x_', 'eatmydata', 'apt-get', '-y',
'--allow-downgrades', 'install', './tmp/hello_2.10-2.1_amd64.deb']
Om52.8s DEBUG: Starting command: ['chroot', '/tmp/tmpchcw125x_', 'eatmydata', 'dpkg-query',
'-f', '${Package} ${Status}\n', '-W', 'hello']
Om52.8s DUMP:
  hello install ok installed
Om52.8s DEBUG: Command ok: ['chroot', '/tmp/tmpchcw125x_', 'eatmydata', 'dpkg-query',
'-f', '${Package} ${Status}\n', '-W', 'hello']
Om52.8s INFO: Installation of ['./tmp/hello_2.10-2.1_amd64.deb'] ok

```

次に apt-get remove コマンドを実行してパッケージをアンインストールします。これで Debian パッケージのインストールからアップグレード、アンインストールするテストは終了です。

```

Om54.3s DEBUG: Starting command: ['chroot', '/tmp/tmpchcw125x_', 'eatmydata', 'apt-get', 'remove', 'hello']
Om54.9s DUMP:
  Reading package lists...
  Building dependency tree...
  Reading state information...
  The following packages will be REMOVED:
  hello
  0 upgraded, 0 newly installed, 1 to remove and 0 not upgraded.
Om54.9s DEBUG: Command ok: ['chroot', '/tmp/tmpchcw125x_', 'eatmydata', 'apt-get', 'remove', 'hello']
Om54.9s DEBUG: Starting command: ['chroot', '/tmp/tmpchcw125x_', 'eatmydata', 'dpkg', '--purge', 'hello']
Om54.9s DUMP:
  dpkg: warning: ignoring request to remove hello which isn't installed
Om54.9s DEBUG: Command ok: ['chroot', '/tmp/tmpchcw125x_', 'eatmydata', 'dpkg', '--purge', 'hello']
Om54.9s DEBUG: Starting command: ['chroot', '/tmp/tmpchcw125x_', 'eatmydata', 'dpkg', '--purge', '--pending']
Om54.9s DEBUG: Command ok: ['chroot', '/tmp/tmpchcw125x_', 'eatmydata', 'dpkg', '--purge', '--pending']
Om54.9s DEBUG: Starting command: ['chroot', '/tmp/tmpchcw125x_', 'eatmydata', 'dpkg', '--remove', '--pending']
Om54.9s DEBUG: Command ok: ['chroot', '/tmp/tmpchcw125x_', 'eatmydata', 'dpkg', '--remove', '--pending']
Om54.9s DEBUG: Starting command: ['lsdf', '-w', '+D', '/tmp/tmpchcw125x_']
Om55.1s DEBUG: Command failed (status=1), but ignoring error: ['lsdf', '-w', '+D', '/tmp/tmpchcw125x_']
Om55.5s DEBUG: No broken symlinks as far as we can find.
Om55.5s DEBUG: Starting command: ['chroot', '/tmp/tmpchcw125x_', 'eatmydata', 'dpkg-divert', '--list']
Om55.5s DUMP:
  diversion of /usr/share/man/man1/sh.1.gz to /usr/share/man/man1/sh.distrib.1.gz by dash
  diversion of /bin/sh to /bin/sh.distrib by dash
Om55.5s DEBUG: Command ok: ['chroot', '/tmp/tmpchcw125x_', 'eatmydata', 'dpkg-divert', '--list']
Om55.5s DEBUG: Starting command: ['chroot', '/tmp/tmpchcw125x_', 'eatmydata', 'apt-get', 'clean']
Om55.5s DEBUG: Command ok: ['chroot', '/tmp/tmpchcw125x_', 'eatmydata', 'apt-get', 'clean']
Om55.5s DEBUG: Recording chroot state
Om55.7s INFO: PASS: Installation, upgrade and purging tests.

```

最後に debootstrap で構築した chroot 環境をクリーンアップします。

```

0m55.9s DEBUG: Starting command: ['umount', '/tmp/tmpkwcw125x_/dev/shm']
0m55.9s DEBUG: Command ok: ['umount', '/tmp/tmpkwcw125x_/dev/shm']
0m55.9s DEBUG: Starting command: ['umount', '/tmp/tmpkwcw125x_/dev/console']
0m55.9s DEBUG: Command ok: ['umount', '/tmp/tmpkwcw125x_/dev/console']
0m55.9s DEBUG: Starting command: ['umount', '/tmp/tmpkwcw125x_/dev/ptmx']
0m55.9s DEBUG: Command ok: ['umount', '/tmp/tmpkwcw125x_/dev/ptmx']
0m55.9s DEBUG: Starting command: ['umount', '/tmp/tmpkwcw125x_/dev/pts']
0m56.0s DEBUG: Command ok: ['umount', '/tmp/tmpkwcw125x_/dev/pts']
0m56.0s DEBUG: Starting command: ['umount', '/tmp/tmpkwcw125x_/proc']
0m56.0s DEBUG: Command ok: ['umount', '/tmp/tmpkwcw125x_/proc']
0m56.0s DEBUG: Starting command: ['rm', '-rf', '--one-file-system', '/tmp/tmpkwcw125x_']
0m56.1s DEBUG: Command ok: ['rm', '-rf', '--one-file-system', '/tmp/tmpkwcw125x_']
0m56.1s DEBUG: Removed directory tree at /tmp/tmpkwcw125x_
0m56.1s INFO: PASS: All tests.
0m56.1s INFO: piuparts run ends.

```

3.4 piuparts コマンドのオプションの紹介

3.4.1 -d オプション：バージョンを指定してテスト

piuparts に "-d _distro_" オプションを指定すると、debootstrap する環境のバージョンを指定してテストを実行します。"-d" オプションは省略でき、省略した場合は "-d sid" を指定したときと同じ動きをします。

```
# piuparts hello_2.10-2.1_amd64.deb -d sid -l /tmp/piuparts.log
```

この "-d" オプションは複数連続して指定することができます。その場合、最初に指定した "-d" オプションのバージョンで debootstrap してからパッケージをインストールし、それ以降に指定した "-d" のバージョンへ順番に apt-get dist-upgrade を行い、最後まで dist-upgrade が完了した状態で piuparts コマンドに指定したパッケージのインストールテストを実行します。

以下の例では、buster の環境を debootstrap で構築して hello パッケージをインストールし、bullseye へ apt-get dist-upgrade し、sid へ apt-get dist-upgrade した後に、hello_2.10-2.1_amd64.deb のアップグレードとアンインストールのテストを実行します。

```
# piuparts hello_2.10-2.1_amd64.deb -d buster -d bullseye -d sid -l /tmp/piuparts.log
```

3.4.2 インストール済みパッケージのテスト

これまでの紹介では Debian パッケージ自体を piuparts の引数に指定してきました。テストするパッケージは .deb ファイルではなく、"-apt" オプションとパッケージ名を指定することで実行している PC にインストールしている Debian パッケージのテストを行うことができます。

```
# piuparts hello --apt -l /tmp/piuparts.log
```

3.4.3 -p オプション：pbuilder のイメージを使う

Debian パッケージをクリーン環境でビルドするツールに pbuilder^{*4} があります。

以下のように pbuilder コマンドを実行して debootstrap したクリーン環境で Debian パッケージのビルドを行うことができます。

```
# pbuilder create
# pbuilder update
# pbuilder build hello_2.10-2.1.dsc
```

pbuilder create を実行したとき、または pbuilder update を実行したときに pbuilder が利用する debootstrap のイメージを「/var/cache/pbuilder/base.tgz」に保存します。

piuparts に "-p" オプションを指定すると、debootstrap を実行する代わりに pbuilder が保存している「/var/cache/pbuilder/base.tgz」をテスト環境に利用します。

debootstrap を実行するとインターネットから毎回約 100 MByte 程度のファイルをダウンロードするため "-p" オ

^{*4} <https://packages.debian.org/ja/sid/pbuilder>

プションを使うことでネットワーク帯域やダウンロード時間を節約できます。

似たようなオプションに `"-basetgz=file"` があり、`base.tgz` ファイルを任意のパスで指定することもできます。

3.4.4 `-docker-image` : docker イメージを使ってテストする

`piuparts` に `"-docker-image=IMAGE"` オプションを指定すると、テストに使う環境に `debootstrap` で構築したイメージを使わず、`docker` イメージを使ってテストを実行します。

実際試してみると、`chroot` コマンドを使って `chroot` のテスト環境内でコマンドを実行するところが `docker run`、`docker inspect`、`docker top`、`docker exec` などを使ってテストを実行する動きに変わります。

3.4.5 `-merged-usr` : /usr マージした環境で `debootstrap` する

`piuparts` に `"-merged-usr"` オプションを指定すると、`debootstrap` するときに `"-merged-usr"` オプションを指定して実行します (オプションなしの場合は `"-no-merged-usr"` を指定しています)。

これで `/usr` マージした環境で Debian パッケージのインストールテストを実行することができます。

3.4.6 `-arch` : アーキテクチャを指定して `debootstrap` する

`piuparts` に `"-arch=ARCH"` オプションを指定すると、`debootstrap` するときに `"-arch=ARCH"` を指定して実行します。

つまり、異なるアーキテクチャの `chroot` 環境を構築して Debian パッケージのインストールテストを実行することができます。ただ、テストする PC が `-arch` オプションに指定したアーキテクチャで動作する環境であること、Debian パッケージも `-arch` オプションに指定したアーキテクチャ向けにビルドしてある必要があります。

3.5 終わりに

`piuparts` を使ってパッケージのテストを試してみました。今回試した範囲では Debian Developer が作成したテストが合格するパッケージのみでした。今後パッケージを自分で作成するときには様々な不具合に遭遇としますので、`piuparts` を使ってうまくテストできるようになるとよいと思います。

なお、`piuparts` コマンドはネットワーク帯域を多く使うため、回線の帯域や通信量の残量に注意してご利用ください。^{*5}

3.6 参考資料

- Debian Wiki - `piuparts` <https://wiki.debian.org/piuparts>
- 岩松 信洋、「`piuparts` の使い方」、*あんどきゅめんとっど でびあん* 2010 年夏号
<https://tokyodebian-team.pages.debian.net/pdf2010/debianmeetingresume201004.pdf>
- Holger Levsen、*DebConf19* 「i'm (a bit) sick of maintaining piuparts.debian.org (mostly) alone, please help」^{*6}

^{*5} https://piuparts.debian.org/doc/README_server.txt に `piuparts` のサーバでは 100 Mbps の帯域を簡単に使い切ってしまうから気をつけてください、と注意書きがあります。

^{*6} <https://debconf19.debconf.org/talks/103-im-a-bit-sick-of-maintaining-piupartsdebianorg-mostly-alone-please-help/>



Debian 勉強会資料

2021年7月17日 初版第1刷発行

東京エリア Debian 勉強会（編集・印刷・発行）
