today's howtos and programming bits

By *Roy Schestowitz*
Created *21/07/2019 - 9:08pm*
Submitted by Roy Schestowitz on Sunday 21st of July 2019 09:08:14 PM Filed under Development [1] HowTos [2]

- **How to fix Ubuntu live USB not booting [3]**
- **How to Create a User Account Without useradd Command in Linux? [4]**
- **Container use cases explained in depth [5]**
- **Containerization and orchestration concepts explained [6]**
- **Set_env.py [7]**

> A good practice when writing complicated software is to put in lots of debugging code. This might be extra logging, or special modes that tweak the behavior to be more understandable, or switches to turn off some aspect of your test suite so you can focus on the part you care about at the moment.
>
> But how do you control that debugging code? Where are the on/off switches? You don?t want to clutter your real UI with controls. A convenient option is environment variables: you can

access them simply in the code, your shell has ways to turn them on and off at a variety of scopes, and they are invisible to your users.

Though if they are invisible to your users, they are also invisible to you! How do you remember what exotic options you?ve coded into your program, and how do you easily see what is set, and change what is set?

- **RPushbullet 0.3.2** **[8]**

  A new release 0.3.2 of the RPushbullet package is now on CRAN. RPushbullet is interfacing the neat Pushbullet service for inter-device messaging, communication, and more. It lets you easily send alerts like the one to the left to your browser, phone, tablet, ? ? or all at once.

  This is the first new release in almost 2 1/2 years, and it once again benefits greatly from contributed pull requests by Colin (twice !) and Chan-Yub ? see below for details.

- **A Makefile for your Go project (2019)** **[9]**

  My most loathed feature of Go was the mandatory use of GOPATH: I do not want to put my own code next to its dependencies. I was not alone and people devised tools or crafted their own Makefile to avoid organizing their code around GOPATH.

- **Writing sustainable Python scripts** **[10]**

  Python is a great language to write a standalone script. Getting to the result can be a matter of a dozen to a few hundred lines of code and, moments later, you can forget about it and focus on your next task.

  Six months later, a co-worker asks you why the script fails and you don?t have a clue: no documentation, hard-coded parameters, nothing logged during the execution and no sensible tests to figure out what may go wrong.

  Turning a ?quick-and-dirty? Python script into a sustainable version, which will be easy to use, understand and support by your co-workers and your future self, only takes some moderate effort.

- **Notes to self when using genRSS.py** **[11]**

Development HowTos

Source URL: http://www.tuxmachines.org/node/126126

Links:
[1] http://www.tuxmachines.org/taxonomy/term/145
[2] http://www.tuxmachines.org/taxonomy/term/98
[3] https://www.addictivetips.com/ubuntu-linux-tips/fix-ubuntu-live-usb-not-booting/
[4] https://www.2daygeek.com/linux-user-account-creation-in-manual-method/
[5] https://www.linuxnix.com/container-use-cases-explained-in-depth/
[6] https://www.linuxnix.com/containerization-and-orchestration-concepts-explained/
[7] https://nedbatchelder.com//blog/201907/set_envpy.html
[8] http://dirk.eddelbuettel.com/blog/2019/07/21#rpushbullet_0.3.2
[9] https://vincent.bernat.ch/en/blog/2019-makefile-build-golang
[10] https://vincent.bernat.ch/en/blog/2019-sustainable-python-script
[11] https://janusworx.com/notes-to-self-when-using-genrsspy.html