

My little Reddit “bot” - in Python

Steve Brorens (@snori74)

LinuxUpskillChallenge.org

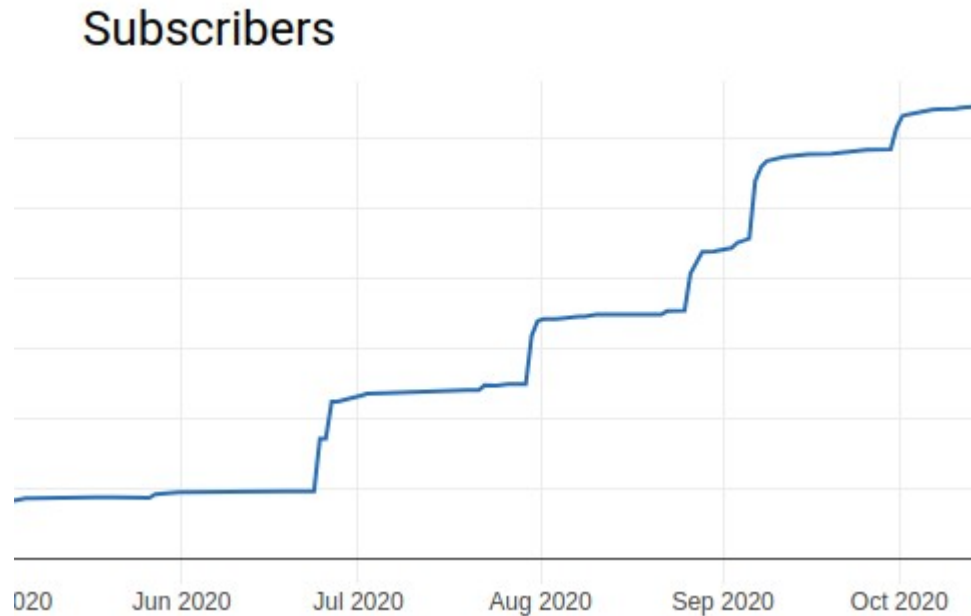
- *A month-long course aimed at those who aspire to get Linux-related jobs in industry - junior Linux sysadmin, devops-related work and similar.*
- *Server focussed and commandline, designed to be well-suited to a Windows or Apple person as well as those that already using Linux in some form.*

LinuxUpskillChallenge.org

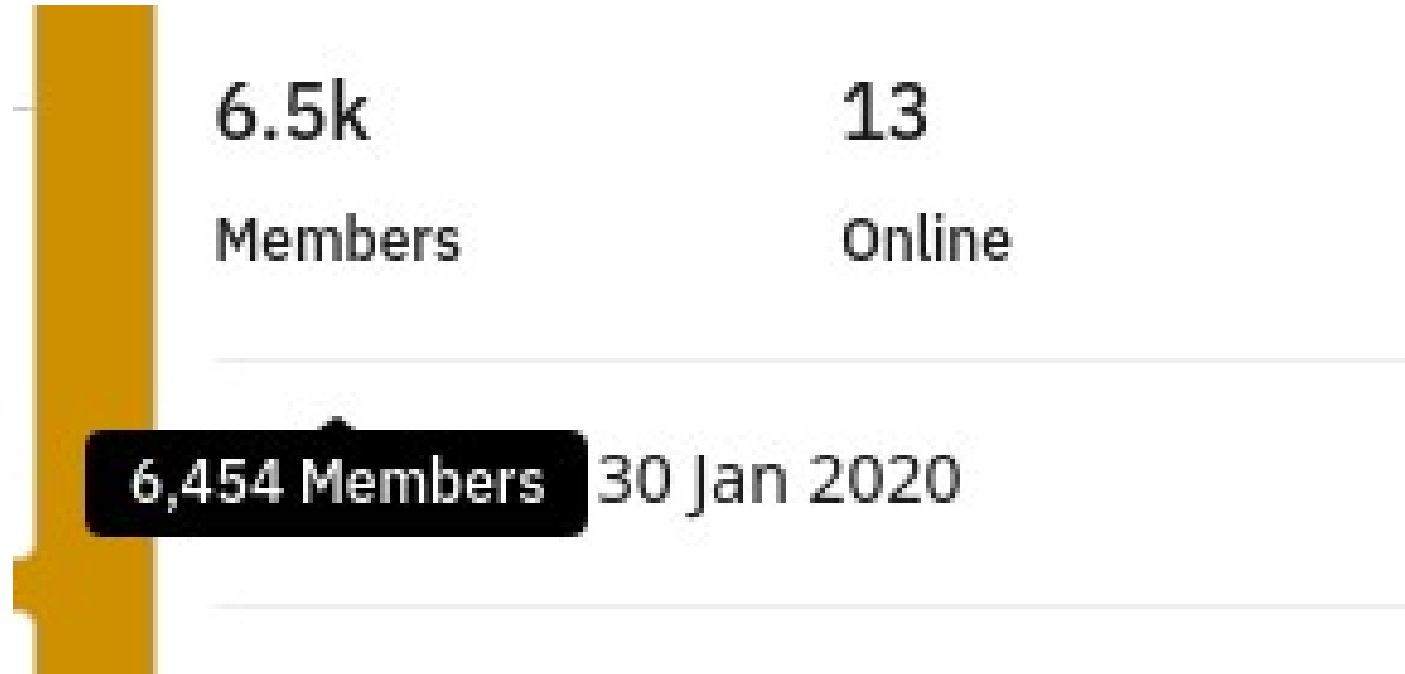
- *Repeats each month*
- *Starts on the first Monday of the month*
- *Twenty “bite-sized” lessons, one each workday*
- *Just the last four lessons are available*

....and completely free!

500+ sign ups each month



Sign-ups – in 8 months



Course content kept in GitHub

The screenshot shows the GitHub repository page for `snori74/linuxupskillchallenge`. The browser address bar shows the URL `github.com/snori74/linuxupskillchallenge`. The repository has 118 forks, 2.8k stars, and 288 forks. The main navigation bar includes links for Code, Issues (10), Pull requests, Actions, Projects, and Security. The repository is currently on the `master` branch. A recent commit by `snori74` is visible, titled "Thanks u/heitorPB for suggesting this!", dated 5 days ago, with 592 likes. Below the commit, there are two files listed: `00-AWS-Free-Ti...` and `00-Digital-Ocea...`, both with a note "Nixed reference to using LTS 18.04" and a date of "last month". The right sidebar contains the "About" section, which describes the repository as a guide for learning Linux skills required to sysadmin a remote Linux server from the commandline. A "Readme" link is also present.

github.com/snori74/linuxupskillchallenge

Dashboard W F A HOME HP M W RDP P m How to Transf... T » Other bookmarks

Search or jump to... Pulls Issues Marketplace Explore

snori74 / linuxupskillchallenge

Unwatch 118 Star 2.8k Fork 288

<> Code ! Issues 10 Pull requests Actions Projects Security ...

master Go to file Add file Code

snori74 Thanks u/heitorPB for suggesting this! 5 days ago 592


00-AWS-Free-Ti...	Nixed reference to using LTS 18.04	last month
00-Digital-Ocea...	Nixed reference to using LTS 18.04	last month

About

Learn the skills required to sysadmin a remote Linux server from the commandline.

Readme

...and posted to Reddit



linuxupskillchallenge

r/linuxupskillchallenge

JOINED

START CHATTING

Create Post

Hot New Top ...

PINNED BY MODERATORS

16 ↑
↓

Posted by u/snori74 **Linux Guru** 10 days ago ✓

HOW THIS WORKS...

0 Comments Share Approved Remove Spam ...

8 ↑
↓

Posted by u/snori74 **Linux Guru** 13 hours ago ✓

Day 9 - Ports, open and closed

0 Comments Share Approved Remove Spam ...

25 ↑
↓

Posted by u/snori74 **Linux Guru** 8 hours ago

I got a postcard today...

About Community MOD TOOLS ...

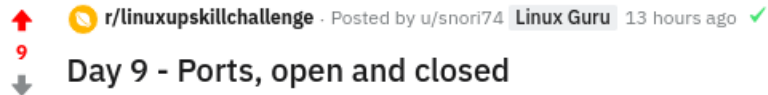
Support and discussion for those following the month-long "Linux Upskill Challenge" course material. Aimed at those who aspire to get Linux-related jobs in industry - junior Linux sysadmin, devops-related work and similar. Discord server (<https://discordapp.com/invite/wd4Zqyk>) run by u/cobaltrune - feel free to chat about the course there too. ✍

6.5k	14
Members	Online

Created 30 Jan 2020

0 Coins

...like this:



INTRO

The two services your server is now running are *sshd* for remote login, and *apache2* for web access. These are both "open to the world" via the TCP/IP "ports" - 22 and 80.

As a sysadmin, you need to understand what ports you have open on your servers because each open port is also a potential focus of attacks. You need to be able to put in place appropriate monitoring and controls.

INSTRUCTIONS

First we'll look at a couple of ways of determining what ports are open on your server:

- `ss` - this, "socket status", is a standard utility - replacing the older `netstat`
- `nmap` - this "port scanner" won't normally be installed by default

There are a wide range of options that can be used with `ss`, but first try: `ss -ltp`

The output lines show which ports are open on which interfaces:


```
ss -ltp
```






The “bot” code is also on GitHub:

snori74 / **auto-luc**

<> **Code** ⓘ Issues 2 🔗 Pull requests ▶ Actions 📁 Projects 📖 Wiki ⓘ Sec

🔗 master ▼ 🧑 1 branch 🏷 0 tags

 **snori74** Merge branch 'master' of <https://github.com/snori74/auto-luc>

 INSTALL.md	Sp
 README.md	Update the README text
 bot_script.py	tested
 functions.py	tested

The Python code:

```
#!/usr/bin/env python
```

```
"""
```

```
bot-script.py - Pulls the lessons for the Linux Upskill Challenge from Github into the subreddit
```

Note 1: Don't run more than once a day - you'll risk multiple posting

Note 2: This is basically a "Steve simulator":

- Steve, in NZ, gets out of bed on a Monday morning
- Feeds the cats, and makes himself a miso soup
- Then posts Monday's lesson - (it's about 8 or 9am by this time)
- However, because this script runs "in the cloud", the server will be running (almost always) on UTC - so the task must be scheduled for 8PM. But, on Monday morning Steve's time, while it will be 8pm in Greenwich and it will be SUNDAY
- So, the script will run and say "No lesson on a weekend" - not good
- This is why the "time_bump" variable exists...

Note 3: This also means that the script can't be run just Monday-Friday (which might seem sensible), but instead should be scheduled EVERY day

...and more

```
# Which day of the course are we on?
day_num, month_name, next_month = check_today(today_date)

if day_num == 1:
    # Post and pin the standard "Day 1" lesson - and the ditto the "short vid
    # Repost "HOW THIS WORKS" text. Don't pin, cos only two can be at a time
    clear_all_pinned(subreddit)
    get_post_pin_day(subreddit, day_num)
    get_post_pin_file(subreddit, "day1-short-video.md")
    # delete the old "HOW..."
    delete_title(subreddit, "HOW THIS WORKS")
    # then pull in the current one and pin...
    get_post_pin_file(subreddit, "how-this-works.md")
    # clear last few of last month's lessons...
    delete_day(subreddit, 20)
    delete_day(subreddit, 19)
    delete_day(subreddit, 18)
    delete_day(subreddit, 17)
    delete_day(subreddit, 16)

elif day_num == 18:
```

How you *think* it works...



...but sometime you get this!



..including some “corner cases”...

- Starts as late as the 7th of the month
- Last lessons of <MONTH> are in <MONTH+1>
- A whole week's gap at the end

Document the “corner cases”

```
def check_today(thisdate):  
    """  
    The course is described as:  
  
    "...runs from the first Monday of the month,  
    and lasts for four weeks..."  
  
    Simple? Yes, but there are some surprising corner  
    cases, e.g.:  
    - sometimes the course doesn't start until the  
      7th of the month (e.g. September 2020)  
    - the last day or two of <MONTH>'s course end up  
      being in <MONTH+1> (e.g. September 2020)  
    - there's sometimes a whole week's gap at the end  
      of a course (e.g. June 2020)  
  
    """  
    delta = datetime.timedelta(days=1)
```

Testing

```
if len(sys.argv) < 2:
    sys.exit(
        "\n Usage: bot-script.py LIVE|TEST [<date>]"
        "\n "
        "\n e.g      bot-script LIVE           " #
        "\n      bot-script TEST           " #
        "\n      bot-script TEST 2020-11-01  " #
        "\n "
    )
```


Testing

```
# Tests by running through the whole of year
# of 2020 - with all output redirected to:
#     test.out
#     test.err
#
# ...to allow checking.
#
# Note: You can watch and continuously refresh
# r/LinuxUpSkillBotTest/ to follow along.
#
start_date = datetime.date(2020, 1, 1)
end_date = datetime.date(2021, 1, 1)
```

Summary

- [LinuxUpskillChallenge.org](https://linuxupskillchallenge.org)
 - github.com/snori74/auto-luc
 - r/linuxupskillchallenge
- Steve Brorens (@snori74)