Rocky Enterprise Linux 9.2 Manual Pages on command 'keep-one-running.1'

$ man keep-one-running.1

run-one(1)   run-one   run-one(1)

NAME

   run-one - run just one instance at a time of some command and unique set of arguments

   (useful for cronjobs, eg)

SYNOPSIS

   run-one COMMAND [ARGS]

   run-this-one COMMAND [ARGS]

   run-one-constantly COMMAND [ARGS]

   keep-one-running COMMAND [ARGS]

   run-one-until-success COMMAND [ARGS]

   run-one-until-failure COMMAND [ARGS]

DESCRIPTION

   run-one is a wrapper script that runs no more than one unique instance of some command

   with a unique set of arguments.

   This is often useful with cronjobs, when you want no more than one copy running at a time.

   run-this-one is exactly like run-one, except that it will use pgrep(1) and kill(1) to find

   and kill any running processes owned by the user and matching the target commands and ar?

   guments. Note that run-this-one will block while trying to kill matching processes, until

   all matching processes are dead.

   run-one-constantly operates exactly like run-one except that it respawns "COMMAND [ARGS]"

   any time COMMAND exits (zero or non-zero).

   keep-one-running is an alias for run-one-constantly.

   run-one-until-success operates exactly like run-one-constantly except that it respawns

"COMMAND [ARGS]" until COMMAND exits successfully (ie, exits zero).

run-one-until-failure operates exactly like run-one-constantly except that it respawns

"COMMAND [ARGS]" until COMMAND exits with failure (ie, exits non-zero).

EXAMPLE

In one shell:

```
$ run-one rsync -azP $HOME $USER@example.com:/srv/backup
foouser/
foouser/.bash_history
    40298 100%   37.13MB/s    0:00:00 (xfer#1, to-check=3509/3516)
foouser/.viminfo
    20352 100%   98.39kB/s    0:00:00 (xfer#3, to-check=3478/3516)
...
sent 746228 bytes  received 413059 bytes  36802.76 bytes/sec
total size is 3732890955  speedup is 3219.99
```

In another shell, while the first is still running:

```
$ run-one rsync -azP $HOME $USER@example.com:/srv/backup
$ echo $?
1
```

Another example... In one shell:

```
$ run-one top
```

In another shell:

```
$ run-one top
$ echo $?
1
$ run-this-one top
top - 17:15:36 up  1:43,  3 users,  load average: 1.05, 1.04, 1.00
Tasks: 170 total,   1 running, 169 sleeping,   0 stopped,   0 zombie
...
```

And note that the process in the first shell was killed.

You might want to keep one ssh connection up and running all the time, with:

```
$  run-one-constantly ssh -N -C -L 3128:localhost:3128 -L 7778:localhost:7778 example.com
&
```

SEE ALSO

flock(1), kill(1), pgrep(1)

AUTHOR

This manpage and the utility was written by Dustin Kirkland <kirkland@ubuntu.com> for
Ubuntu systems (but may be used by others). Permission is granted to copy, distribute
and/or modify this document under the terms of the GNU General Public License, Version 3
published by the Free Software Foundation.

On Debian systems, the complete text of the GNU General Public License can be found in
/usr/share/common-licenses/GPL, or on the web at http://www.gnu.org/licenses/gpl.txt.

run-one   9 Jan 2011   run-one(1)