Data Engineering
Crontab Schedule

Taught by Pichaya Tandayya

PRINCE OF SONGKLA UNIVERSITY

Crontab vs Cronjob

- A crontab is a file which contains the schedule of cronjob entries to be run at specified times.
  - Crontab is short for cron table. You can think of a crontab as a configuration file that specifies shell commands to run periodically on a given schedule.
- A cronjob is basically instructions to run a command at a prescribed time.

Source: https://tonyteaches.tech/schedule-python-script/

Cron Job

CRON JOB

cron expression

* * * * * username command to be executed

- Day of Week (0=Sun . . 6=Sat)
- Month (1 . .12)
- Day of Month (1 . .31)
- Hour (0. .23)
- Minute (0. .59)

```
# every Mon midnight
0 0 * * 1  command

# everday 05:04 AM
4 5 * * *  command

# every Sun 12:05 PM
5 12 * * 0 command

# every year
@yearly    command
@annually  command

# every week
@weekly    command
0 0 * * 0  command

# every hour
@hourly    command
0 * * * *  command

# every month
@monthly command
0 0 1 * *  command

# every midnight
@midnight command
# same as daily

# every day
@daily   command
0 0 * * *  command

# every reboot
@reboot  command

# every 6 hours
0 */6 * * * command

# every 5minutes
*/5 * * * * command
```

Operators

*	All potential values in a field
,	List numerios values
-	A range of values
/	Specify a step value for a field
L	Last value, Only used in month and day-of-week field
W	Get the closest weekday from a given time
#	Only used in day-of-week field, followed by a number between 1 and 5.
?	No specific value. Used in day-of-month and day-of-week fields

Crontab

$ crontab -e	Edit or create a crontab file if it doesn't already exist
$ crontab	Display crontab file
$ crontab -r	Remove a crontab
$ crontab -u username -l	Display another user's crontab file
$ crontab -u username -e	Edit another user's crontab file
$ echo "username" > /etc/cron.allow	Allow specific user to use to crontab
$ echo "username" > /etc/cron.deny	Deny specific user to use to crontab
$ crotab -v	Display the last time you edited your crontab file

sysxplore.com

Crontab Syntax

```
┌──────────── minute (0 - 59)
│ ┌────────── hour (0 - 23)
│ │ ┌──────── day of the month (1 - 31)
│ │ │ ┌────── month (1 - 12)
│ │ │ │ ┌──── day of the week (0 - 6) (Sunday to Saturday;
│ │ │ │ │                7 is also Sunday on some systems)
│ │ │ │ │
│ │ │ │ │
* * * * * command to execute from left to right
```

- Each of the 5 asterisks represents minute, hour, day of month, month, and day of week.
- Finally on the very right is the actual command to execute.

Source: https://tonyteaches.tech/schedule-python-script/

More Examples of Cronjobs

Description	Crontab Syntax
Every day at midnight	0 0 * * *
Every day at 3:30 PM	30 15 * * *
Each hour	0 * * * *
Monthly at midnight on the first day of the month	0 0 1 * *
Midnight on the 1st ,10th, and 15th of every month	0 0 1,10,15 * *
Every weekday at 8:01 PM	1 20 * * 1-5
Midnight on the 15th of March, June, September, and December	0 0 15 3,6,9,12 *
Every year at midnight on April 25th	0 0 25 4 *

Source: https://tonyteaches.tech/schedule-python-script/

watch

- A Linux command used to execute a program periodically, showing output in full screen.
- By default, the specified command will run every 2 seconds and watch will run until interrupted

Syntax

watch [options] command

Examples

watch –d free –m

watch –n 1 free –m

Watch options

-b, --beep beep if command has a non-zero exit
-c, --color interpret ANSI color and style sequences
-d, --differences[=<permanent>] highlight changes between updates
-e, --errexit exit if command has a non-zero exit
-g, --chgexit exit when output from command changes
-n, --interval <secs> seconds to wait between updates
-p, --precise attempt run command in precise intervals
-t, --no-title turn off header
-w, --no-wrap turn off line wrapping
-x, --exec pass command to exec instead of "sh -c"
-h, --help display this help and exit
-v, --version output version information and exit

A Very Simple Cronjob Example

```
crontab -e
```

-e here means *edit*

```
* * * * * date > /tmp/test.txt
```

```
watch cat /tmp/test.txt
```

```
Mon Oct 7 12:21:00 CEST 2018
```

```
Mon Oct 7 12:22:00 CEST 2018
```

Source: https://tonyteaches.tech/schedule-python-script/

Schedule a Python Script with Crontab

```
import random
from datetime import datetime
now = datetime.now()
num = random.randint(1, 101)
with open('/tmp/rand.txt', 'a') as f:
        f.write('{} - Your random number is {}\n'.format(now, num))
```

Source: https://tonyteaches.tech/schedule-python-script/

Test

```
python rand.py

cat /tmp/rand.txt

2018-10-07 12:33:21.211066 - Your random number is 65
```

Source: https://tonyteaches.tech/schedule-python-script/

Schedule the Python script to execute every minute

```
crontab -e
* * * * * /usr/bin/python3 /home/pichaya/rand.py
watch cat /tmp/rand.txt
```

2018-10-01 13:57:31.158516 - Your random number is 27

2018-10-01 14:01:00.175556 - Your random number is 23

2018-10-01 14:02:00.267484 - Your random number is 81

2018-10-01 14:03:00.386802 - Your random number is 85

2018-10-01 14:04:00.504855 - Your random number is 22

2018-10-01 14:05:00.613324 - Your random number is 94

2018-10-01 14:06:00.706200 - Your random number is 45

Source: https://tonyteaches.tech/schedule-python-script/

Clear out all of cronjobs

```
crontab -r
```

Source: https://tonyteaches.tech/schedule-python-script/

Data Backup

- เทคนิคการสำเนาข้อมูลจากแหล่งแรกไปยังแหล่งที่สอง
- ทำเพื่อคุ้มครองกรณีเกิดภัยพิบัติ อุบัติเหตุ หรือการกระทำ
- จากผู้อื่นที่มุ่งร้าย
- ป้องกันการขัดจังหวะการดำเนินธุรกิจในองค์กรที่ใช้ข้อมูลใน
- การขับเคลื่อน

Source: Thanathip Limna, Cloud Storage and Backup: Review for Data Engineer, 21 August 2021

ตัวเลือกในการทำสำเนา

- Removable Media – USB flash drive
- Redundancy
- External Hard Drive
- Hardware Appliances
- Backup Software
- Cloud Backup Service - Backup as a Service (BaaS)

Source: Thanathip Limna, Cloud Storage and Backup: Review for Data Engineer, 21 August 2021

3-2-1 Backup Strategy

- สำเนาข้อมูล 3 ชุด – 1 ชุดเดิม 2 ชุดใหม่
- สำเนาข้อมูลไว้ในหน่วยเก็บข้อมูล 2 ชนิดที่แตกต่างกัน
- สำเนาข้อมูลไปไว้ที่อื่น 1 แห่ง

Source: Thanathip Limna, Cloud Storage and Backup: Review for Data Engineer, 21 August 2021

cp Command Syntax

```
$ cp SOURCE DEST
$ cp SOURCE DIRECTORY
$ cp SOURCE1 SOURCE2 SOURCE3 SOURCEn DIRECTORY
$ cp [OPTION] SOURCE DEST
$ cp [OPTION] SOURCE DIRECTORY
```

Where,

- In the first and second syntax you copy SOURCE file to DEST file or DIRECTORY.
- In the third syntax you copy multiple SOURCE(s) (files) to DIRECTORY.

Linux Copy File Examples

```
$ cp file.doc newfile.doc
$ ls -l *.doc

$ cp main.c demo.h lib.c backup

$ cp main.c demo.h lib.c /home/project/backup/

$ cp * /home/tom/backup          # Copying all files
$ cp -R * /home/tom/backup       # Recursive copy, including all its files and subdirectories
$ cp -i foo bar                  # Interactive option, get prompt before overwriting file
```

เอกสารอ้างอิง

- https://tonyteaches.tech/schedule-python-script/
- How to Schedule a Python Script with a Cron Job https://www.youtube.com/watch?v=EgrpfvBc7ks
- Linux/Mac Tutorial: Cron Jobs - How to Schedule Commands with crontab https://www.youtube.com/watch?v=QZJ1drMQz1A


PRINCE OF
SONGKLA
UNIVERSITY