

# ขั้นตอนการตั้งเวลาเพื่อใช้งาน File transfer

## macOS

### Overview

1. Set Passwordless โดย ssh-keygen สำหรับ ssh
2. เตรียม Script สำหรับโอนไฟล์
3. ตั้งเวลารันงาน โดยใช้คำสั่ง crontab

### Requirement

1. Terminal
2. NSTDA VPN or NSTDA Network
3. TARA user account

## 1. Set Passwordless โดยใช้ ssh-keygen

ดำเนินการทำ Passwordless เพื่อที่เวลาโอนข้อมูลจะไม่มีคำถาม Password ในขณะที่ตั้งเวลา ผ่านทาง ssh-keygen เพราะหากไม่ทำจะไม่สามารถโอนไฟล์ได้ โดยมีขั้นตอนดังนี้

1. ทำการสร้าง Key สำหรับ SSH เข้าระบบ TARA โดยใช้คำสั่ง

```
$ ssh-keygen -t rsa -b 4096 -C "@tara.nstda.or.th"
```

2. ไม่ต้องระบุข้อมูลในข้อที่ขึ้นมาถาม โดยที่สามารถกด Enter ผ่านในส่วนที่ขึ้นในด้านล่าง

```
Enter file in which to save the key (/home/best/.ssh/id_rsa): <ไม่ต้องระบุข้อมูล>
Enter passphrase (empty for no passphrase): <ไม่ต้องระบุข้อมูล>
Enter same passphrase again: <ไม่ต้องระบุข้อมูล>
```

3. เมื่อผ่านข้อที่ 2 จะทำการสร้าง Key ขึ้นมาแบบดังรูป

```
$ ssh-keygen -t rsa -b 4096 -C "@tara.nstda.or.th"

Generating public/private rsa key pair.
Enter file in which to save the key (/home/best/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:

Your identification has been saved in /home/best/.ssh/id_rsa.
Your public key has been saved in /home/best/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:yilC3rKweLNbiupcOgavyewERZK17PntdFclMXIPI0Y @tara.nstda.or.th
The key's randomart image is:
+---[RSA 4096]---+
|.oo          .E * |
|.+.         . + * |
| +          o . |
| o .        . . |
|. =      S   .   |
|+= o o o   .    |
|=o=.+ * . .    |
|B**= + . .    |
|XX+o   .      |
+----[SHA256]-----+
```

4. ตรวจสอบว่ามีไฟล์การสร้างไฟล์ id\_rsa.pub ที่เป็น Key หรือไม่ โดยใช้คำสั่ง

```
$ ls -l .ssh/
total 8
-rw----- 1 test test 3243 May 21 17:39 id_rsa
-rw-r--r-- 1 test test 743 May 21 17:39 id_rsa.pub
```

5. ทำการ Copy Key ไปยังระบบ TARA โดยใช้คำสั่ง

```
$ ssh-keygen -t rsa -b 4096 -C "@tara.nstda.or.th"
```

```
$ ssh-copy-id -i .ssh/id_rsa.pub test@tara.nstda.or.th

/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed:
".ssh/id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s),
to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you
are prompted now it is to install the new keys

Password: <ระบุ Password ของระบบ TARA>

Number of key(s) added: 1

Now try logging into the machine, with:  "ssh
'tsuwan@tara.nstda.or.th'"

and check to make sure that only the key(s) you wanted were added.
```

6. ทดสอบ SSH เข้าระบบ TARA หาก SSH เข้าระบบได้โดยไม่ต้องใส่ Password แสดงว่าสำเร็จ

## 2. เตรียม Script สำหรับโอนข้อมูล (ทำสองแบบ โฟลเดอร์และไฟล์)

1. เริ่มจากการเตรียม Script สำหรับใช้โอนข้อมูล

- 1.1 กรณีที่ต้องการ Upload ไปยัง TARA ระบุไฟล์

```
$ vi upload.sh
```

ภายใน Script ใส่คำสั่งที่ต้องการโอนข้อมูลขึ้นระบบ TARA ลงไปดังนี้

```
#!/bin/bash
scp /path-on-laptop/yourfile.txt yourusername@tara.nstda.or.th:
```

เมื่อใส่คำสั่งเสร็จเรียบร้อยแล้ว กด :wq เพื่อทำการบันทึกคำสั่ง

- 1.2 กรณีที่ต้องการ Download ข้อมูลจาก TARA ระบุไฟล์

```
$ vi download.sh
```

ภายใน Script ใส่คำสั่งที่ต้องการโอนข้อมูลจากระบบ TARA มาที่เครื่องตัวเอง ดังนี้

```
#!/bin/bash
scp yourusername@tara.nstda.or.th:~/path-on-server/yourfile.txt path-on-
laptop/yourfile.txt
```

เมื่อใส่คำสั่งเสร็จเรียบร้อย กด :wq เพื่อทำการบันทึกคำสั่ง

### 1.3 กรณีที่ต้องการ Upload ไปยัง TARA ทั้ง Path

```
$ vi upload.sh
```

ภายใน Script ใส่คำสั่งที่ต้องการโอนข้อมูลขึ้นระบบ TARA ลงไปดังนี้

```
#!/bin/bash  
scp -r /path-on-laptop yourusername@tara.nstda.or.th:
```

เมื่อใส่คำสั่งเสร็จเรียบร้อย กด :wq เพื่อทำการบันทึกคำสั่ง

### 1.4 กรณีที่ต้องการ Download ข้อมูลจาก TARA ทั้ง Path

```
$ vi download.sh
```

ภายใน Script ใส่คำสั่งที่ต้องการโอนข้อมูลขึ้นระบบ TARA ลงไปดังนี้

```
#!/bin/bash  
scp -r yourusername@tara.nstda.or.th:~/path-on-server /path-on-laptop
```

เมื่อใส่คำสั่งเสร็จเรียบร้อย กด :wq เพื่อทำการบันทึกคำสั่ง

### 1.3 ทดสอบ Script ก่อนว่าสามารถทำงานได้จริงหรือไม่ โดยใช้คำสั่ง

```
$ sh upload.sh
```

หากสำเร็จจะขึ้นตัวอย่าง ดังรูป

```
$ sh upload.sh  
yourfile.txt 100% 8362 420.6KB/s
```

## 3. ตั้งเวลาดำเนินงาน โดยใช้คำสั่ง crontab

1. ทำการเปิด crontab โดยใช้คำสั่ง

```
$ crontab -e
```

2. ตั้งเวลาที่ต้องการจะ Upload หรือ Download ข้อมูล ตัวอย่างคือ ต้องการทำการ Upload ข้อมูล เวลา 9 โมงเช้า

```
0 9 * * * sh /path/upload.sh
```

```

# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow  command
0 9 * * * /path/upload.sh

```

อีกตัวอย่าง ทำการระบุวันและเวลาอย่างละเอียด เพื่อทำการตั้งเวลา วันที่ 21 เดือน 10 ปี 2020 เวลา 14.30

```

# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow  command
30 14 21 10 * /path/upload.sh

```

เมื่อใส่คำสั่งเสร็จเรียบร้อยแล้ว กด :wq เพื่อทำการบันทึกคำสั่ง

- ตรวจสอบดูว่ามีงานของที่ตั้งอยู่ใน schedule หรือไม่ โดยใช้คำสั่ง

```
$ crontab -l
```

```

$ crontab -l

# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow   command

0 9 * * * /path/upload.sh
30 14 21 10 * /path/upload.sh

```

4. เมื่อถึงกำหนดเวลาที่ได้ทำการตั้งไว้ จะทำการส่งคำสั่งสำหรับการโอนไฟล์โดยอัตโนมัติ
5. หากใช้งานคำสั่งเสร็จสิ้น ต้องทำการลบคำสั่งที่ใส่ไป มิเช่นนั้น 9 โมงเช้า คำสั่งนี้จะทำอีกครั้งในเวลาที่ได้กำหนดไว้
6. หากต้องการลบหรือยกเลิก Job ใช้คำสั่ง

```
$ crontab -e
```

กต :wq เพื่อทำการบันทึก เมื่อตรวจสอบดูคำสั่งที่อยู่ใน schedule จะพบว่าคำสั่งถูกลบออกแล้ว

```

$ crontab -l

# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow   command

30 14 21 10 * /path/upload.sh

```

### **อ่านข้อมูลเพิ่มเติม**

crontab command:

<http://manpages.ubuntu.com/manpages/cosmic/man5/crontab.5.html>

crontab calculator:

<https://crontab.guru/>

ssh-keygen:

<http://manpages.ubuntu.com/manpages/trusty/man1/ssh-keygen.1.html>