

(一)啟動

1. 啟動三個節點，如下圖

啟動 node1

在 start_node1.sh 的資料夾下，輸入指令：「./start_node1.sh」

```
ntut01@node1:~$ ./start_node1.sh
```

啟動 node2

在 start_node2.sh 的資料夾下，輸入指令：「./start_node2.sh」

```
ntut02@node2:~$ ./start_node2.sh
```

啟動 node3

在 start_node3.sh 的資料夾下，輸入指令：「./start_node3.sh」

```
ntut03@node3:~$ ./start_node3.sh
```

2. 啟動 flask

在 flaskRun.sh 的資料夾下，輸入指令：「./flaskRun.sh」(紅色框框的內容會自動跳出來)

```
ntut01@node1:~/Blockchain-main$ ./flaskRun.sh
ntut01@node1:~/Blockchain-main$ nohup: appending output to 'nohup.out'
```

3. 啟動每日排程

在 dailyUpload.sh 的資料夾下，輸入指令：「./dailyUpload.sh」(紅色框框的內容會自動跳出來)

```
ntut01@node1:~/Blockchain-main$ ./dailyUpload.sh
ntut01@node1:~/Blockchain-main$ nohup: appending output to 'nohup.out'
```

可以確認上述三個步驟都有正確執行

1. 確認有啟動 指令：「ps aux | grep "geth"」

節點三台 server 都必須要輸入(各自檢查)

2. 確認有啟動 flask(網頁)：「ps aux | grep "flask"」

flask(網頁)只有 node1 要輸入

3. 確認有啟動 dailyUpload(每日排程)：「ps aux | grep "dailyUpload"」

dailyUpload(每日排程)只有 node1 要輸入

只要輸入都有看到紅框處的內容就代表有啟動成功

```
ntut01@node1:~$ ps aux | grep "dailyUpload"
ntut01  4566  0.5  0.1  78648 51432 ?        S   16:47   0:03 python3 dailyUpload.py 16:41
ntut01  5383  0.0  0.0  18172  2688 pts/1    S+  16:59   0:00 grep --color=auto dailyUpload
ntut01@node1:~$ ps aux | grep "geth"
ntut01  4788 23.3  0.5 2847520 167352 ?        Sl   16:52   2:00 geth --datadir node1 --networkid 12345 --nat extip:10.0.129.2 --port 5021 --syncmode full --cache 4096 --nodiscover --http --http.port 5002 --http.addr 0.0.0.0 --http.corsdomain * --http.api personal,eth,net,web3,txpool,miner --authrpc.addr 0.0.0.0 --authrpc.port 6002 --allow-insecure-unlock --unlock 0xBd367d52531889EbCBA1F53B47FB75ED334262b0 --password node1/passwd.txt --verbosity 3
ntut01  5387  0.0  0.0  18172  2432 pts/1    S+  17:00   0:00 grep --color=auto geth
ntut01@node1:~$ ps aux | grep "flask"
ntut01  4742  3.4  0.1 233504 59352 ?        S   16:50   0:23 /usr/bin/python3 /home/ntut01/.local/bin/flask run
ntut01  5391  0.0  0.0  18172  2688 pts/1    R+  17:01   0:00 grep --color=auto flask
```

(二)關閉

1. 暫停節點連線

停止 node1

在 stop_node1.sh 的資料夾下，輸入指令: 「./ stop_node1.sh」

停止 node2

在 stop_node2.sh 的資料夾下，輸入指令: 「./ stop_node2.sh」

停止 node3

在 stop_node3.sh 的資料夾下，輸入指令: 「./ stop_node3.sh」

2. flask(網頁) 和 dailyUpload (每日排程上傳)

看到每個功能啟動的 pid 後，刪除 pid

指令: 「kill pid」

綠框處為各功能的 pid

```
ntut01@node1:~$ ps aux | grep "dailyUpload"
ntut01 4566 0.5 0.1 78648 51432 ? S 16:47 0:03 python3 dailyUpload.py 16:41
ntut01 5383 0.0 0.0 18172 2688 pts/1 S+ 16:59 0:00 grep --color=auto dailyUpload
ntut01@node1:~$ ps aux | grep "geth"
ntut01 4788 23.3 0.5 2847520 167352 ? Sl 16:52 2:00 geth --datadir node1 --networkid 12345 --nat extip:10
.0.129.2 --port 5021 --syncmode full --cache 4096 --nodiscover --http --http.port 5002 --http.addr 0.0.0.0 --http.corsde
main * --http.api personal,eth.net,web3,txpool,miner --authrpc.addr 0.0.0.0 --authrpc.port 6002 --allow-insecure-unlock
--unlock 0xBd367d52531889EbCBA1F53B47FB75ED334262b0 --password node1/password.txt --verbosity 3
ntut01 5387 0.0 0.0 18172 2432 pts/1 S+ 17:00 0:00 grep --color=auto geth
ntut01@node1:~$ ps aux | grep "flask"
ntut01 4742 3.4 0.1 233504 59352 ? S 16:50 0:23 /usr/bin/python3 /home/ntut01/.local/bin/flask run
ntut01 5391 0.0 0.0 18172 2688 pts/1 R+ 17:01 0:00 grep --color=auto flask
```

以上圖舉例：

1. 關閉 geth(三個節點都要做) → 節點 pid 為 4788

指令: 「kill 4788」

2. 關閉 flask(網頁) → flask pid 為 4742

指令: 「kill 4742」

2. 關閉 dailyUpload (每日排程上傳) → dailyUpload pid 為 4566

指令: 「kill 4566」