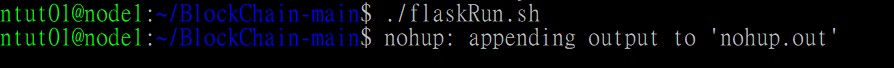
(一)啟動

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」  
   一張含有 文字, 螢幕擷取畫面, 黑色, 字型 的圖片

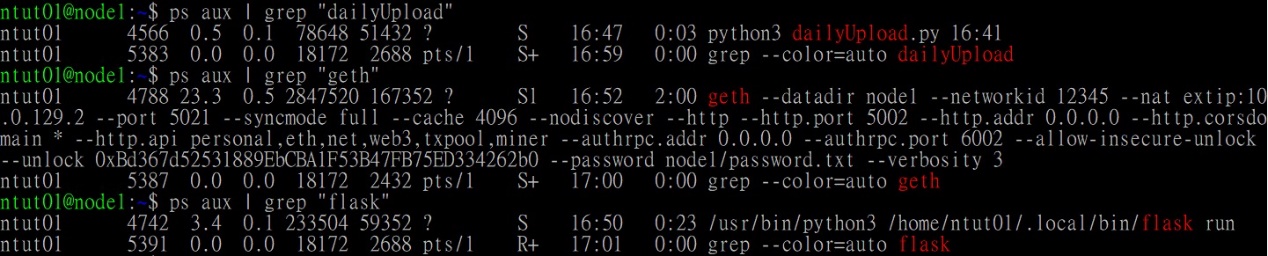
   自動產生的描述  
   啟動node2  
   在start\_node2.sh的資料夾下，輸入指令:「./start\_node2.sh」  
     
   啟動node3  
   在start\_node3.sh的資料夾下，輸入指令:「./start\_node3.sh」  
   一張含有 文字, 螢幕擷取畫面, 字型, 黑色 的圖片

   自動產生的描述
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)  
   
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)  
   

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

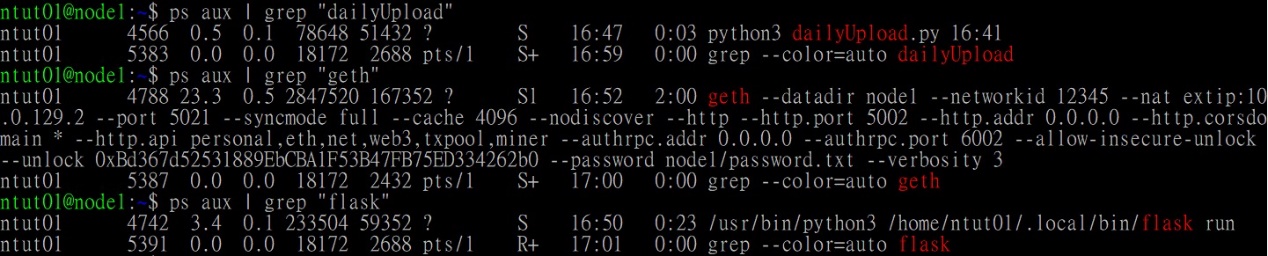
3. 確認有啟動dailyUpload(每日排程):「ps aux | grep "dailyUpload"」  
dailyUpload(每日排程)只有node1要輸入  
只要輸入都有看到紅框處的內容就代表有啟動成功



(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



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」