

# Operating System Project#3

0510532 楊上萱

## 1. 程式說明：

Tcp server 的部分是用 python 的 socketserver 來實作的。

程式分成兩部分－EchoHandler 和 main。

### ● EchoHandle：

Echo Handler 會接收來自 client 的 message，並回傳

msg+seed+hash\_value 給 client。

- i. 當接收到 msg 時，先 decode 再把行尾換行(\n)拿掉。然後交給 generate\_echo\_msg 來產生要回傳的訊息。

```
43     def handle(self):
44         """ Main Tcp Server Handler """
45         print('Got connection from', self.client_address)
46         while True:
47             msg = self.request.recv(8192)
48             if not msg:
49                 break
50             msg = msg.decode('utf-8').rstrip()
51             print('Recv msg: ', msg)
52             self.request.send(self.generate_echo_msg(msg).encode('utf-8'))
```

- ii. generate\_echo\_msg 會產生一直產生 seed 和 hash\_value 直到 hash\_value 符合 PoW 難度。最後回傳 msg+seed+hash\_value。

```

11     def randomStringDigits(self, stringLength):
12         """ Generate a random string of letters and digits """
13         lettersAndDigits = string.ascii_letters + string.digits
14         return ''.join(
15             random.choice(lettersAndDigits) for i in range(stringLength))
16
17     def hash_sha256(self, data):
18         """ Generate sha256 hash """
19         hash_value = hashlib.sha256(data.encode('utf-8')).hexdigest()
20         return hash_value
21
22     def checkPoW(self, hash_val):
23         """ Check PoW correctness """
24         PoWDifficulty = 1
25         for i in range(PoWDifficulty):
26             if hash_val[i] != '0':
27                 return False
28         return True
29
30     def generate_echo_msg(self, msg):
31         while True:
32             """ Generate echo message: msg+seed+hash_value """
33             seed = self.randomStringDigits(5)
34             hash_value = self.hash_sha256(seed + msg)
35             # print('hash value***: ', hash_value)
36             # check PoW
37             if (self.checkPoW(hash_value)):
38                 break
39             echo_msg = msg + ',' + seed + ',' + hash_value + "\n"
40             print(echo_msg)
41             return echo_msg
42

```

- Main:

建立 Tcp server 在 ip = localhost, port=20000，並等待來自 client 的訊息。

```

55 if __name__ == '__main__':
56     serv = ThreadingTCPServer(('', 20000), EchoHandler)
57     print('Listening on 127.0.0.1:8000...')
58     serv.serve_forever()
59

```

## 2. 過程中遇到的問題：

- 從 client 收到的 message 是 byte literal，必須先 decode 成 string 才能做 string 相關運算。
- echo message 要加'\n'才能讓 client 分別不同訊息。

- 一開始忘記 server 也要檢查 PoW，導致 client 那邊一直抱錯。之後改回來就好了。

### 3. 效能：

PoW 難度	0	1	2	3	4	5	6
回覆數/秒	867.550	779.383	190.783	10.050	0.700	0.033	0.000