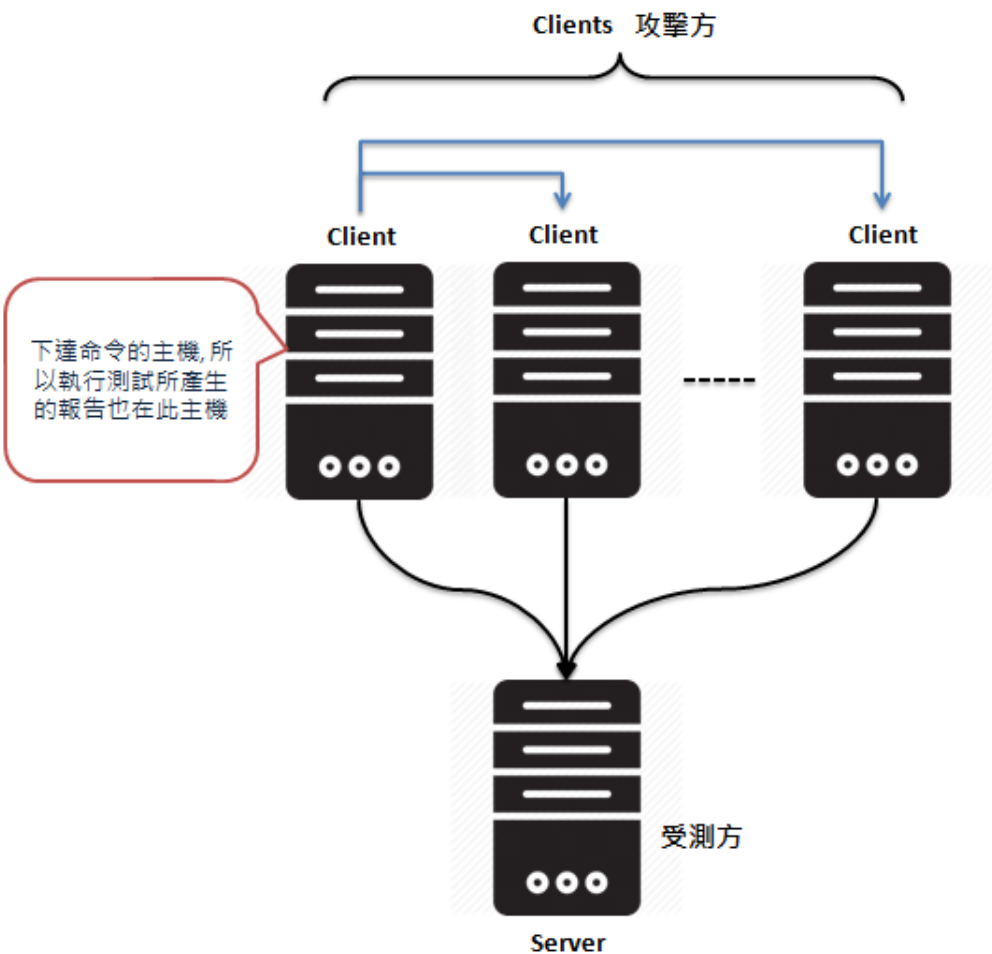


Tsung

Script

Tsung Configuration

參見 [tsung.xml](#)



Environment Setting

參見 [Tsung架設](#)

Client	54.168.154.216 (controller) 54.95.199.37 54.249.204.224
Server	54.150.230.6

SSH KEY

參見 [Tsung installation and Cluster Configuration](#)

```
ssh-keygen -t dsa
chmod 600 ~/.ssh/id_dsa
cd ~/.ssh
cat id_dsa.pub >> authorized_keys
ssh-agent sh -c 'ssh-add < /dev/null && bash'
```

把建立的private key及public key複製到其他的client

```
scp -i ~/.ssh/jp2tvnowcom.pem id_dsa ubuntu@54.95.199.37:/home/ubuntu/.ssh/
scp -i ~/.ssh/jp2tvnowcom.pem id_dsa.pub ubuntu@54.95.199.37:/home/ubuntu/.ssh/
ssh -i ~/.ssh/jp2tvnowcom.pem ubuntu@54.95.199.37
cd .ssh
cat id_dsa.pub >> authorized_keys
```

HOST NAME

```
sudo vim /etc/hosts
把clients加入
54.168.154.216      tsung0
54.95.199.37       tsung3
54.249.204.224     tsung4
```

SSH VERIFICATION

每個 client 都要執行

```
ssh tsung0
ssh tsung3
ssh tsung4
```

應該要可以不輸入password就可ssh過去才行

Test Result

Issue

```
ubuntu@ip-10-185-134-66:~$ tsung status
{"init terminating in do_boot",{timeout,{gen_server,call,[{global,ts_config_server},{status}]}}}

Crash dump was written to: erl_crash.dump
init terminating in do_boot ()
```

或是利用 /usr/lib/tsung/bin/tsung_stats.pl 產生報表時出現
size_rcb is equal to 0 !
size_sent is equal to 0 !

或是看報表產出的log中
=ERROR REPORT==
failed to connect to 之字眼

SOLUTION

確定所使用的Clients內的 Tsung, erlang版本皆相同。
確定所使用的Clients內的 public key, private key皆有設定。

tsung.xml內的client，不用設定 ip 值。
因為此值應為 ifconfig 所得的 address，與對外address不同。
利用 /etc/hosts 所設定的即可，不用去管對內 ip address 為何。

測試結果(持續累積中)

參見 http://203.74.1.177/tsung_log/

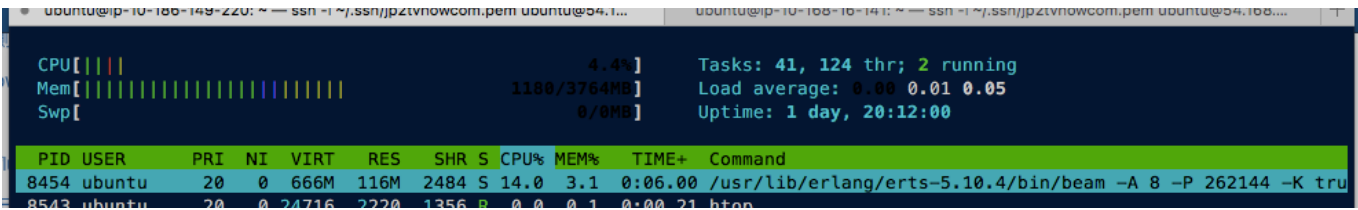
下列的table列出主要使用多台client測試1server且成功的結果。
其餘為失敗或為1client-1server 的結果

Folder	Description
20160323-1636	使用 activityInfo > updateUser 來測試 (1 session , 2 requests) 其中 1秒200 users, 持續1分，共 12000 session 2個request間的時間 thinktime 是隨機的，平均為2s 皆為GET，所以沒有error

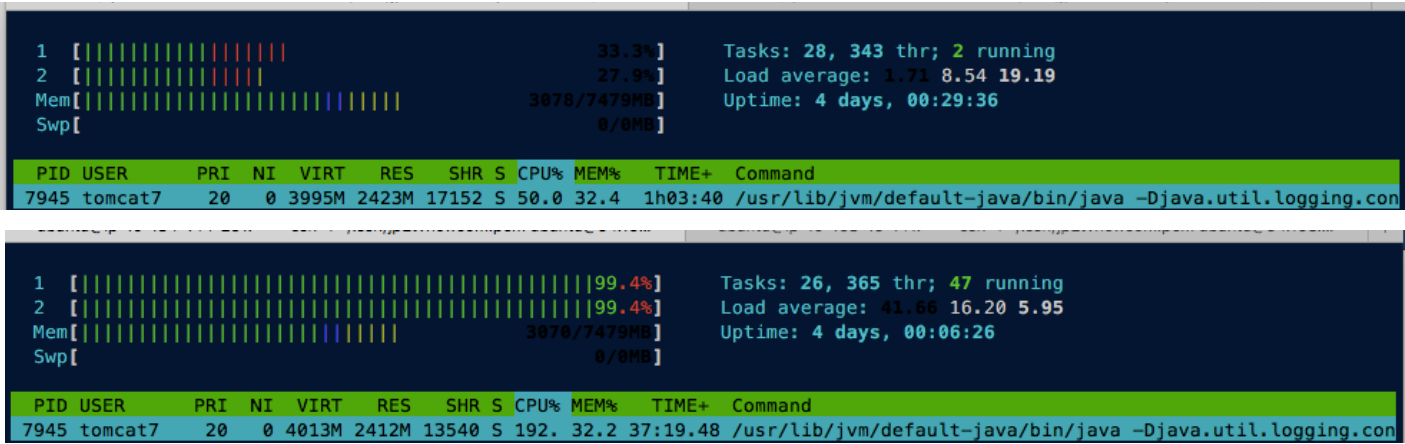
Folder	Description
20160323-1734	<p>使用『使用者開啟並打卡』的劇情 activityInfo > updateUser > checkInUser > sendEventResult 其中 fb_id參數值為動態變數(random) 1秒 200 users，持續2分，共 24000 session request間的時間thinktime是隨機的，平均為2s 只有activityInfo的 Type為GET，其餘皆為POST</p>
20160324-1538 (tsung 1.5.1) 20160325-1112 (tsung 1.6.0)	<p>分解『使用者開啟並打卡』的劇情 取其中 activityInfo > updateUser (1 session) 其中 updateUser中的fb_id為random動態變數 1秒 200 users，持續3分，共 36000 session request 間的時間 thinktime 是隨機的，平均為2s</p>
20160324-1629 (tsung 1.5.1) 20160325-1049 (tsung 1.6.0)	<p>分解『使用者開啟並打卡』的劇情 取其中 activityInfo > updateUser > checkInUser 其中 updateUser 及 checkInUser 中的fb_id為random動態變數 1秒 200 users，持續3分，共執行 36000 session request 間的時間 thinktime 是隨機的，平均為2s</p>
20160329-1025	<p>Static Page Test (1 session, 1 request) 每秒建立 200 users 持續 30 分，共執行 360000 session</p>
20160329-1452	<p>Static Page Test (concurrent user 300000) 利用每秒增加 500 users 並持續 10 分 及 限制最大值 而執行 session 時，利用 for 迴圈，讓 user 不停地執行 request 而達到 concurrent 的現象 因為 tsung will end when all started users have finished their session</p>
20160329-1757	<p>Static Page Test (concurrent user 20000) 利用每秒增加 200 users 並持續 10 分及限制最大值</p>
20160330-0953	Static Page Test (concurrent user 15000)
20160330-1016	Static Page Test (concurrent user 12000)

測試主機效能

CLIENT 端 – 攻擊方

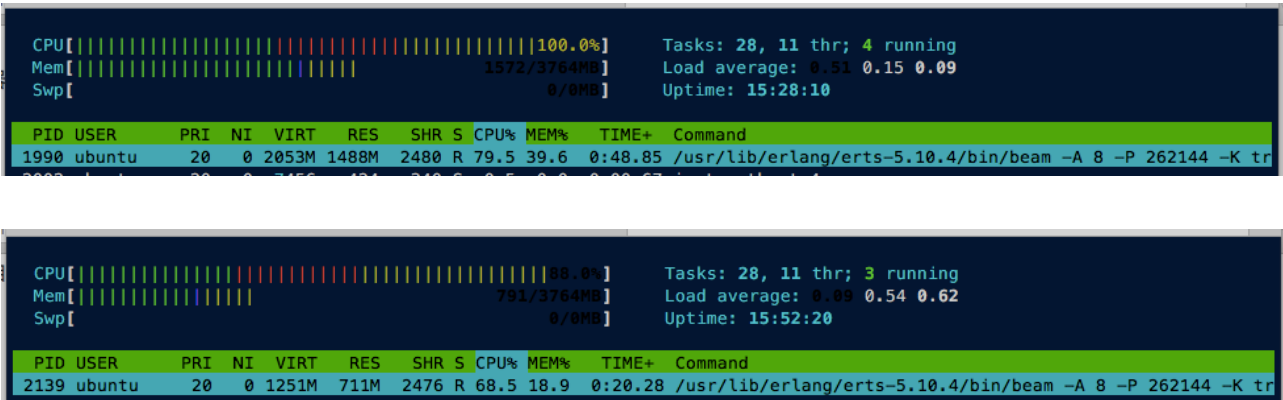


SERVER 端 – 受測方



上圖為 20160325-1112 的負載截圖
下圖為 20160325-1049 的負載截圖

CONCURRENT USERS TEST



上圖為 20160330-0953 的負載截圖
下圖為 20160330-1016 的負載截圖

Tsung 討論

由之前的測試及結果圖分析，可發現在 concurrent users 的測試中：

3 台 clients (AWS m3.medium, 攻擊端) + 1 台 server (AWS m3.medium, 受測端) 的架構大約達 12000 人左右時負載就會達 100%，若人數再多可能造成的 error rate 就會提高