案例篇：服務吞吐量下降很厲害，怎麼分析？

動態追蹤

服務在運作的同時，收集各種資訊來判斷問題

ftrace

perf

eBPF/BCC

SystemTap

火焰圖

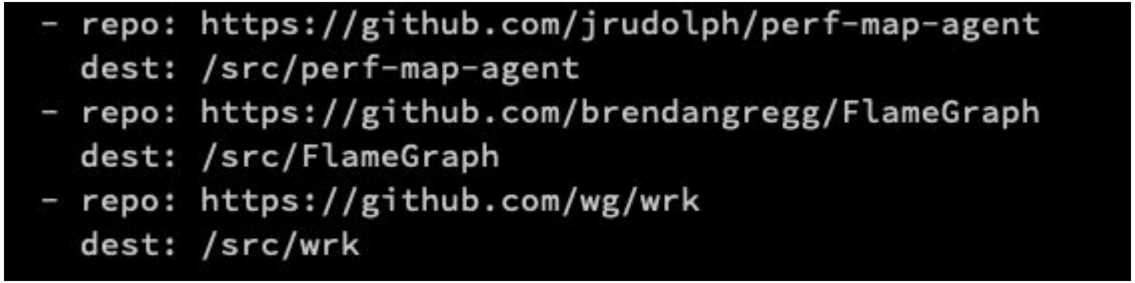
環境

VM：2 cpu + 8G memory

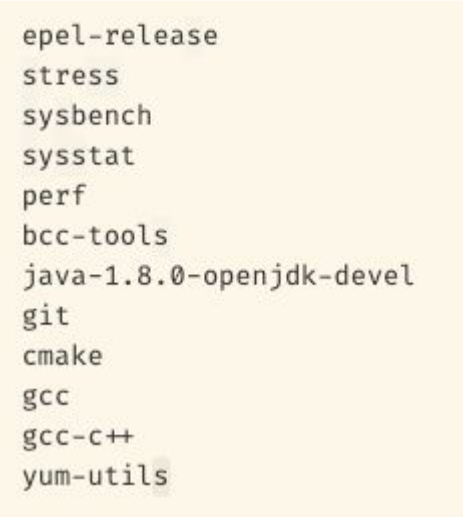
Docker, curl, wrk, perf, FlameGraph

server: nginx + php web

client: curl, wrk



```
- repo: https://github.com/jrudolph/perf-map-agent
  dest: /src/perf-map-agent
- repo: https://github.com/brendangregg/FlameGraph
  dest: /src/FlameGraph
- repo: https://github.com/wg/wrk
  dest: /src/wrk
```



```
epel-release
stress
sysbench
sysstat
perf
bcc-tools
java-1.8.0-openjdk-devel
git
cmake
gcc
gcc-c++
yum-utils
```

分析吞吐量瓶頸的順序

1. 看host的連線狀況:
   a. ss -s (看到estb太少，closed & timewait太多)
   b. netstat -s (看到socket dropped)
   c. dmesg (看到nf_conntrack: table full, dropping packet, TCP: request_sock_TCP: Possible SYN flooding on port 9000. Sending cookies. Check SNMP counters)
2. 看ap server的log
   a. nginx (http 499 & [crit] 99: Cannot assign requested address & 1024 worker_connections are not enough)
   b. php (max_children)
3. 從log看到異常狀況，檢查對應的設定、系統限制
4. 一次修改一個地方，逐步測試

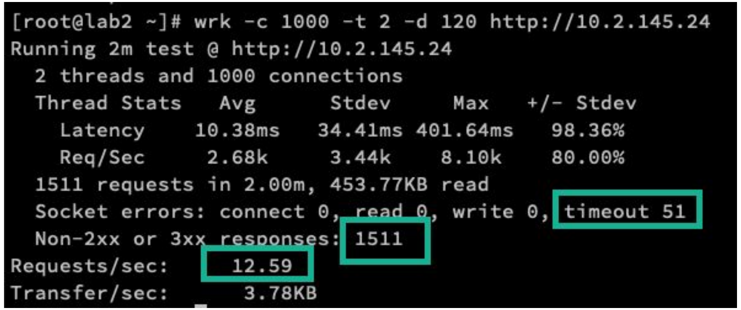
測試

rps：12

socket timeout：51

error response：1511

服務rps太低，request錯誤太多



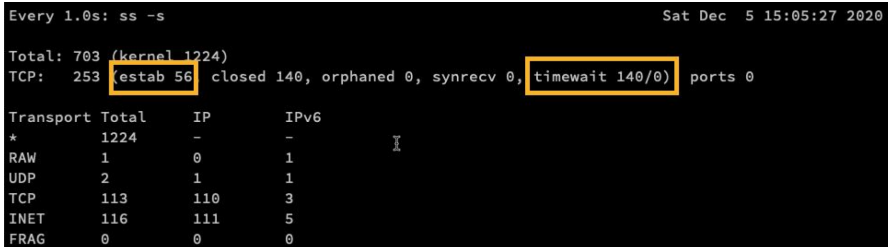
```
[root@lab2 ~]# wrk -c 1000 -t 2 -d 120 http://10.2.145.24
Running 2m test @ http://10.2.145.24
  2 threads and 1000 connections
  Thread Stats   Avg      Stdev     Max   +/- Stdev
    Latency    10.38ms   34.41ms 401.64ms   98.36%
    Req/Sec     2.68k     3.44k    8.10k    80.00%
  1511 requests in 2.00m, 453.77KB read
  Socket errors: connect 0, read 0, write 0, timeout 51
  Non-2xx or 3xx responses: 1511
Requests/sec:     12.59
Transfer/sec:      3.78KB
```

先從connection開始

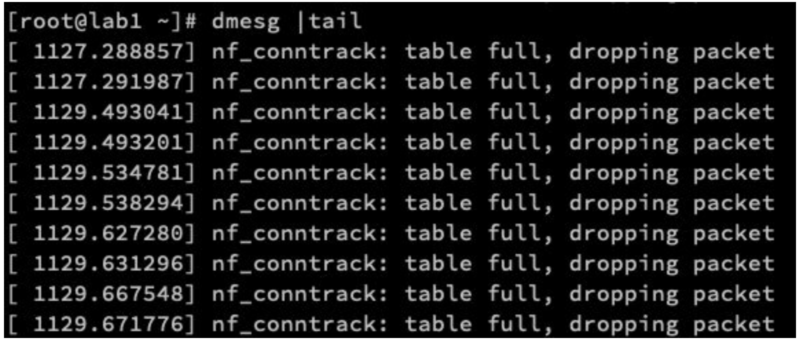
查TCP連接數(ss -s)
有close及timewait

查系統記錄看連線異常(dmesg|tail)
發現drop packet

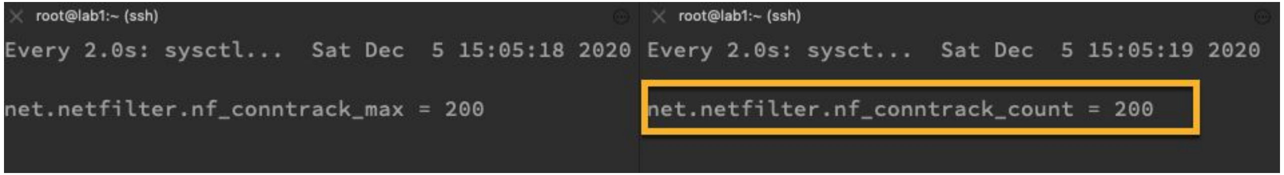
查系統參數，已經達到預設的限制
sysctl net.netfilter.nf_conntrack_max
sysctl net.netfilter.nf_conntrack_count



```
Every 1.0s: ss -s                                           Sat Dec  5 15:05:27 2020

Total: 703 (kernel 1224)
TCP:   253 (estab 56, closed 140, orphaned 0, synrecv 0, timewait 140/0), ports 0

Transport Total     IP        IPv6
*         1224      -         -
RAW       1         0         1
UDP       2         1         1
TCP       113       110       3
INET      116       111       5
FRAG      0         0         0
```



```
[root@lab1 ~]# dmesg |tail
[ 1127.288857] nf_conntrack: table full, dropping packet
[ 1127.291987] nf_conntrack: table full, dropping packet
[ 1129.493041] nf_conntrack: table full, dropping packet
[ 1129.493201] nf_conntrack: table full, dropping packet
[ 1129.534781] nf_conntrack: table full, dropping packet
[ 1129.538294] nf_conntrack: table full, dropping packet
[ 1129.627280] nf_conntrack: table full, dropping packet
[ 1129.631296] nf_conntrack: table full, dropping packet
[ 1129.667548] nf_conntrack: table full, dropping packet
[ 1129.671776] nf_conntrack: table full, dropping packet
```

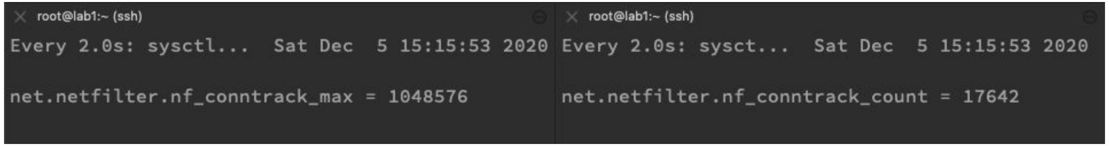


```
Every 2.0s: sysctl...   Sat Dec  5 15:05:18 2020

net.netfilter.nf_conntrack_max = 200
```

```
Every 2.0s: sysct...   Sat Dec  5 15:05:19 2020

net.netfilter.nf_conntrack_count = 200
```

先提高nf_conntrack_max看看

sysctl -w net.netfilter.nf_conntrack_max=1048576

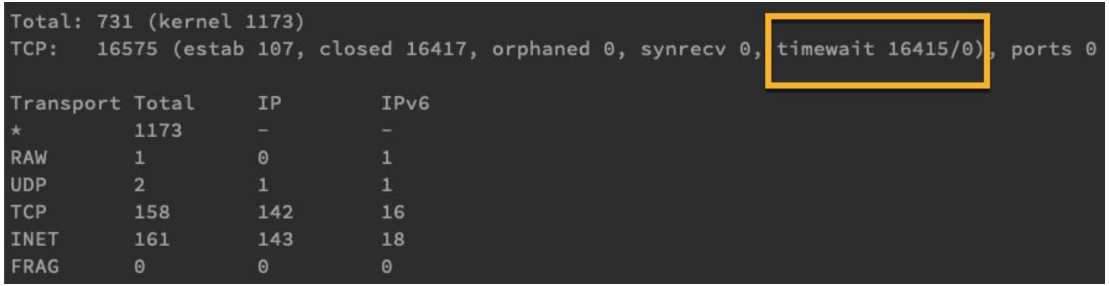


```
Every 2.0s: sysctl...  Sat Dec  5 15:15:53 2020

net.netfilter.nf_conntrack_max = 1048576
```

```
Every 2.0s: sysct...  Sat Dec  5 15:15:53 2020

net.netfilter.nf_conntrack_count = 17642
```

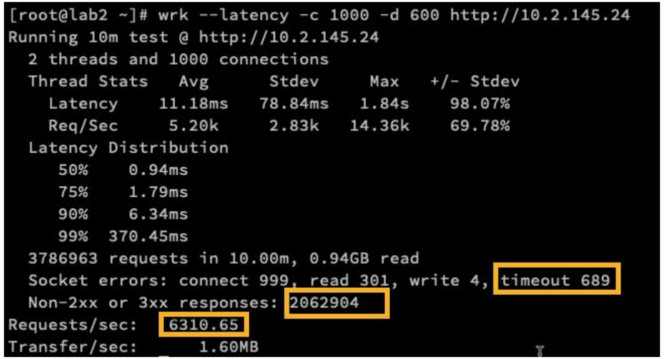
再測試，雖然rps到6310了

但觀察到timewait更多



```
Total: 731 (kernel 1173)
TCP:   16575 (estab 107, closed 16417, orphaned 0, synrecv 0, timewait 16415/0), ports 0

Transport Total     IP        IPv6
*         1173      -         -
RAW       1         0         1
UDP       2         1         1
TCP       158       142       16
INET      161       143       18
FRAG      0         0         0
```

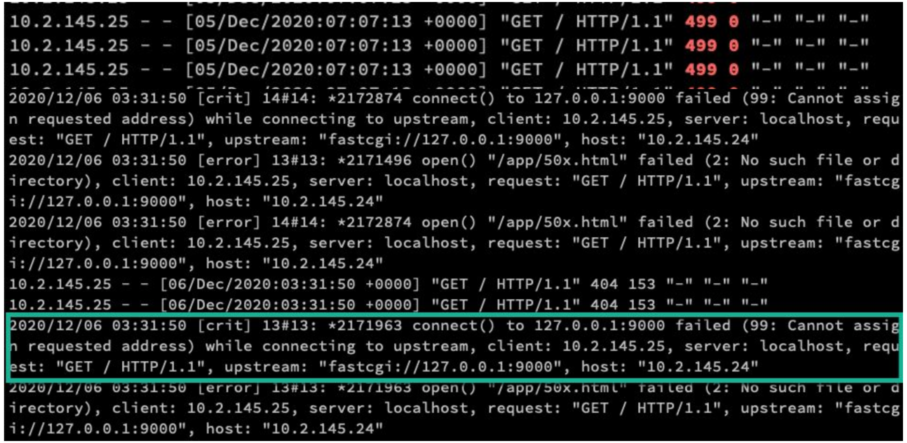
失敗的request和socket error timeout更多了



```
[root@lab2 ~]# wrk --latency -c 1000 -d 600 http://10.2.145.24
Running 10m test @ http://10.2.145.24
  2 threads and 1000 connections
  Thread Stats   Avg      Stdev     Max   +/- Stdev
    Latency    11.18ms   78.84ms   1.84s    98.07%
    Req/Sec     5.20k     2.83k   14.36k    69.78%
  Latency Distribution
     50%    0.94ms
     75%    1.79ms
     90%    6.34ms
     99%  370.45ms
  3786963 requests in 10.00m, 0.94GB read
  Socket errors: connect 999, read 301, write 4, timeout 689
  Non-2xx or 3xx responses: 2062904
Requests/sec:   6310.65
Transfer/sec:      1.60MB
```

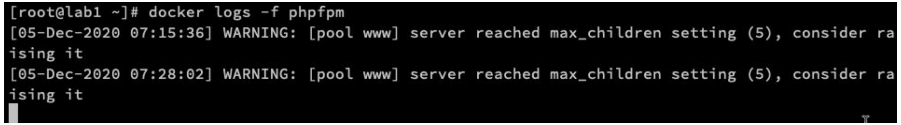
針對request error，先看nginx及php的log

nginx的log發現大量的http 499
以及crit connect failed



```
10.2.145.25 - - [05/Dec/2020:07:07:13 +0000] "GET / HTTP/1.1" 499 0 "-" "-" "-"
10.2.145.25 - - [05/Dec/2020:07:07:13 +0000] "GET / HTTP/1.1" 499 0 "-" "-" "-"
10.2.145.25 - - [05/Dec/2020:07:07:13 +0000] "GET / HTTP/1.1" 499 0 "-" "-" "-"
2020/12/06 03:31:50 [crit] 14#14: *2172874 connect() to 127.0.0.1:9000 failed (99: Cannot assig
n requested address) while connecting to upstream, client: 10.2.145.25, server: localhost, requ
est: "GET / HTTP/1.1", upstream: "fastcgi://127.0.0.1:9000", host: "10.2.145.24"
2020/12/06 03:31:50 [error] 13#13: *2171496 open() "/app/50x.html" failed (2: No such file or d
irectory), client: 10.2.145.25, server: localhost, request: "GET / HTTP/1.1", upstream: "fastcg
i://127.0.0.1:9000", host: "10.2.145.24"
2020/12/06 03:31:50 [error] 14#14: *2172874 open() "/app/50x.html" failed (2: No such file or d
irectory), client: 10.2.145.25, server: localhost, request: "GET / HTTP/1.1", upstream: "fastcg
i://127.0.0.1:9000", host: "10.2.145.24"
10.2.145.25 - - [06/Dec/2020:03:31:50 +0000] "GET / HTTP/1.1" 404 153 "-" "-" "-"
10.2.145.25 - - [06/Dec/2020:03:31:50 +0000] "GET / HTTP/1.1" 404 153 "-" "-" "-"
2020/12/06 03:31:50 [crit] 13#13: *2171963 connect() to 127.0.0.1:9000 failed (99: Cannot assig
n requested address) while connecting to upstream, client: 10.2.145.25, server: localhost, requ
est: "GET / HTTP/1.1", upstream: "fastcgi://127.0.0.1:9000", host: "10.2.145.24"
2020/12/06 03:31:50 [error] 13#13: *2171963 open() "/app/50x.html" failed (2: No such file or d
irectory), client: 10.2.145.25, server: localhost, request: "GET / HTTP/1.1", upstream: "fastcg
i://127.0.0.1:9000", host: "10.2.145.24"
```

php的log說server已經到達max_children限制（5）



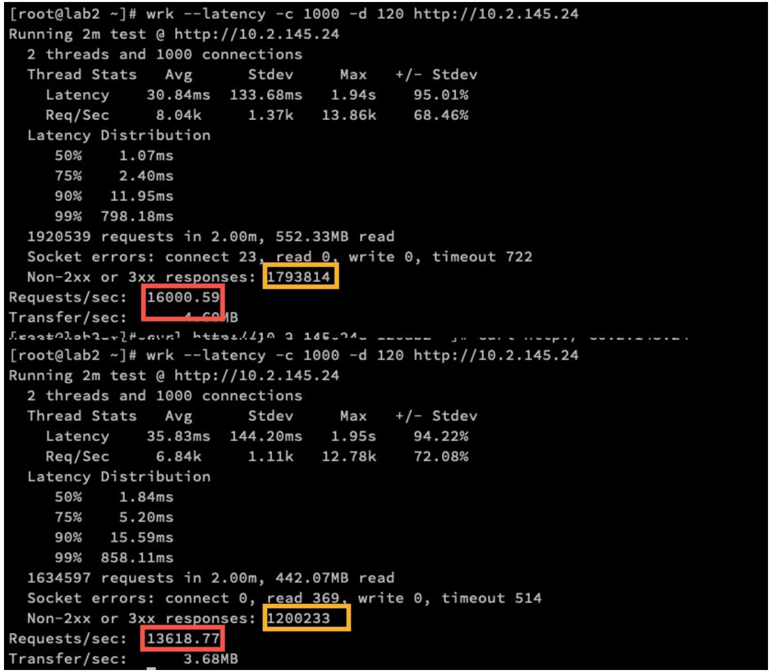
```
[root@lab1 ~]# docker logs -f phpfpm
[05-Dec-2020 07:15:36] WARNING: [pool www] server reached max_children setting (5), consider ra
ising it
[05-Dec-2020 07:28:02] WARNING: [pool www] server reached max_children setting (5), consider ra
ising it
```

先調php的設定，max_children=20之後

request些微下降：16000 → 13618

成功的response增加：1793814 → 1200233

吞吐量還是不夠，失敗率還是太高



```
[root@lab2 ~]# wrk --latency -c 1000 -d 120 http://10.2.145.24
Running 2m test @ http://10.2.145.24
  2 threads and 1000 connections
  Thread Stats   Avg      Stdev     Max   +/- Stdev
    Latency    30.84ms  133.68ms   1.94s    95.01%
    Req/Sec     8.04k     1.37k   13.86k    68.46%
  Latency Distribution
     50%    1.07ms
     75%    2.40ms
     90%   11.95ms
     99%  798.18ms
  1920539 requests in 2.00m, 552.33MB read
  Socket errors: connect 23, read 0, write 0, timeout 722
  Non-2xx or 3xx responses: 1793814
Requests/sec:  16000.59
Transfer/sec:      4.60MB
```

```
[root@lab2 ~]# wrk --latency -c 1000 -d 120 http://10.2.145.24
Running 2m test @ http://10.2.145.24
  2 threads and 1000 connections
  Thread Stats   Avg      Stdev     Max   +/- Stdev
    Latency    35.83ms  144.20ms   1.95s    94.22%
    Req/Sec     6.84k     1.11k   12.78k    72.08%
  Latency Distribution
     50%    1.84ms
     75%    5.20ms
     90%   15.59ms
     99%  858.11ms
  1634597 requests in 2.00m, 442.07MB read
  Socket errors: connect 0, read 369, write 0, timeout 514
  Non-2xx or 3xx responses: 1200233
Requests/sec:  13618.77
Transfer/sec:      3.68MB
```

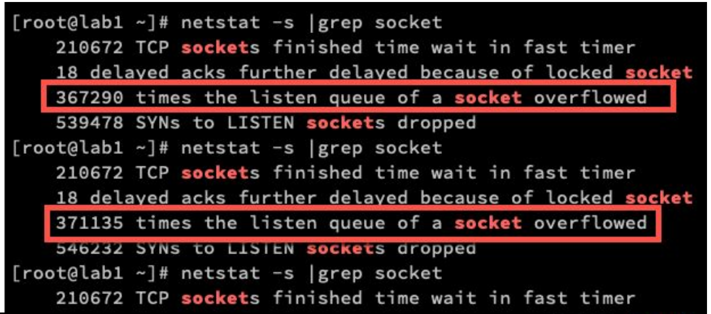
檢查socket方面的異常

netstat看到socket overflowed，dropped

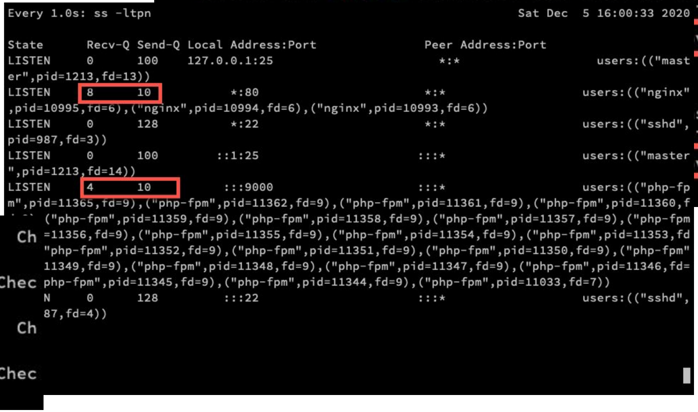
檢查socket queue（ss -ltnp）

recevie queue接近send queue的上限了

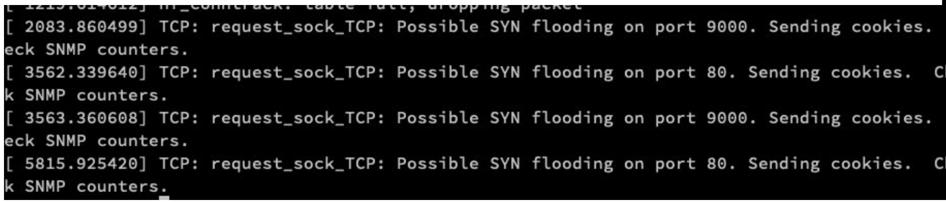
dmesg|tail



```
[root@lab1 ~]# netstat -s |grep socket
    210672 TCP sockets finished time wait in fast timer
    18 delayed acks further delayed because of locked socket
    367290 times the listen queue of a socket overflowed
    539478 SYNs to LISTEN sockets dropped
[root@lab1 ~]# netstat -s |grep socket
    210672 TCP sockets finished time wait in fast timer
    18 delayed acks further delayed because of locked socket
    371135 times the listen queue of a socket overflowed
    546232 SYNs to LISTEN sockets dropped
[root@lab1 ~]# netstat -s |grep socket
    210672 TCP sockets finished time wait in fast timer
                                               locked socket
                                               verflowed

                                               st timer
                                               locked socket
                                               verflowed
```



```
Every 1.0s: ss -ltpn                                         Sat Dec  5 16:00:33 2020

State      Recv-Q Send-Q Local Address:Port               Peer Address:Port
LISTEN     0      100    127.0.0.1:25                      *:*                   users:(("mast
er",pid=1213,fd=13))
LISTEN     8      10          *:80                         *:*                  users:(("nginx"
,pid=10995,fd=6),("nginx",pid=10994,fd=6),("nginx",pid=10993,fd=6))
LISTEN     0      128         *:22                         *:*                  users:(("sshd",
pid=987,fd=3))
LISTEN     0      100        ::1:25                       :::*                  users:(("master
",pid=1213,fd=14))
LISTEN     4      10          :::9000                     :::*                  users:(("php-fp
m",pid=11365,fd=9),("php-fpm",pid=11362,fd=9),("php-fpm",pid=11361,fd=9),("php-fpm",pid=11360,f
    ("php-fpm",pid=11359,fd=9),("php-fpm",pid=11358,fd=9),("php-fpm",pid=11357,fd=9),("php-fpm
    =11356,fd=9),("php-fpm",pid=11355,fd=9),("php-fpm",pid=11354,fd=9),("php-fpm",pid=11353,fd
    "php-fpm",pid=11352,fd=9),("php-fpm",pid=11351,fd=9),("php-fpm",pid=11350,fd=9),("php-fpm"
    11349,fd=9),("php-fpm",pid=11348,fd=9),("php-fpm",pid=11347,fd=9),("php-fpm",pid=11346,fd=
    php-fpm",pid=11345,fd=9),("php-fpm",pid=11344,fd=9),("php-fpm",pid=11033,fd=7))
    N      0      128         :::22                       :::*                  users:(("sshd",
    87,fd=4))
```



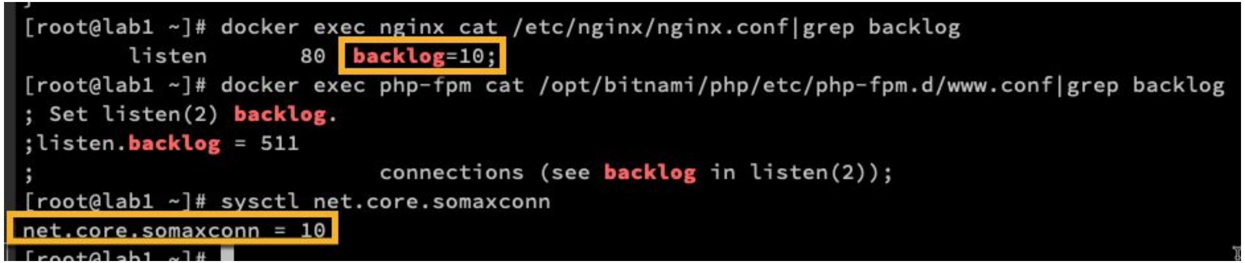
```
[ 1219.814812] nf_conntrack: table full, dropping packet
[ 2083.860499] TCP: request_sock_TCP: Possible SYN flooding on port 9000. Sending cookies.  Ch
eck SNMP counters.
[ 3562.339640] TCP: request_sock_TCP: Possible SYN flooding on port 80. Sending cookies.  Chec
k SNMP counters.
[ 3563.360608] TCP: request_sock_TCP: Possible SYN flooding on port 9000. Sending cookies.  Ch
eck SNMP counters.
[ 5815.925420] TCP: request_sock_TCP: Possible SYN flooding on port 80. Sending cookies.  Chec
k SNMP counters.
```

檢查各地配置的queue

系統層級：sysctl net.core.somaxconn

nginx：docker exec nginx cat /etc/nginx/nginx.conf|grep backlog

php：docker exec php-fpm cat /opt/bitnami/php/etc/php-fpm.d/www.conf|grep backlog



```
[root@lab1 ~]# docker exec nginx cat /etc/nginx/nginx.conf|grep backlog
        listen       80  backlog=10;
[root@lab1 ~]# docker exec php-fpm cat /opt/bitnami/php/etc/php-fpm.d/www.conf|grep backlog
; Set listen(2) backlog.
;listen.backlog = 511
;                               connections (see backlog in listen(2));
[root@lab1 ~]# sysctl net.core.somaxconn
net.core.somaxconn = 10
[root@lab1 ~]#
```

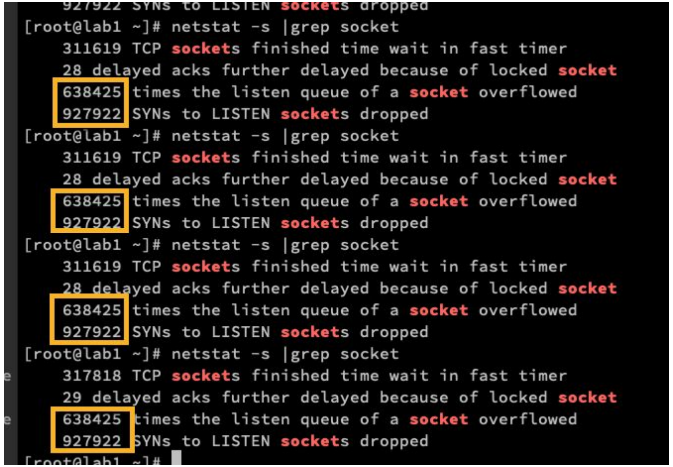
覺得queue太小了，調整

sysctl -w net.core.somaxconn=65535

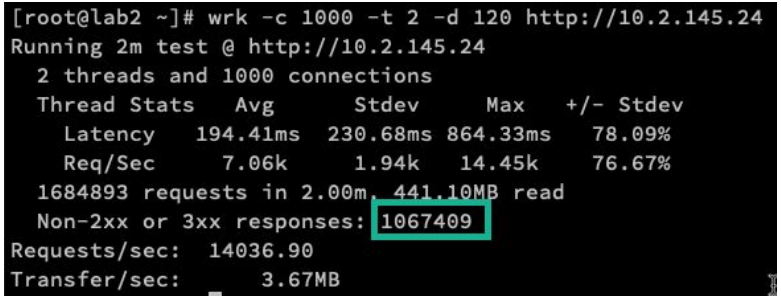
不再掉封包

測試結果不再出現socket error

但是失敗的request更多了



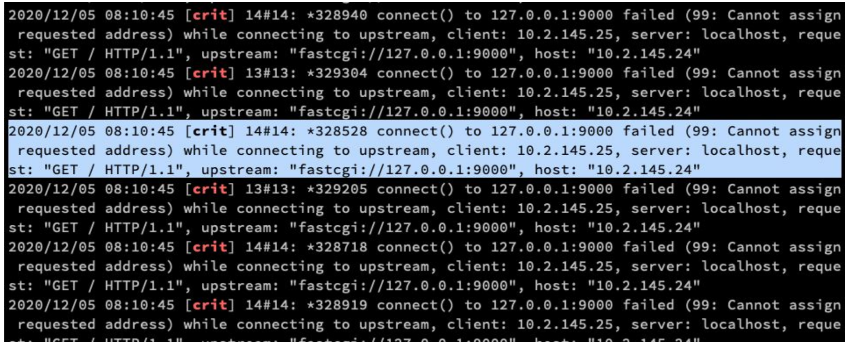
```
    927922 SYNs to LISTEN sockets dropped
[root@lab1 ~]# netstat -s |grep socket
    311619 TCP sockets finished time wait in fast timer
    28 delayed acks further delayed because of locked socket
    638425 times the listen queue of a socket overflowed
    927922 SYNs to LISTEN sockets dropped
[root@lab1 ~]# netstat -s |grep socket
    311619 TCP sockets finished time wait in fast timer
    28 delayed acks further delayed because of locked socket
    638425 times the listen queue of a socket overflowed
    927922 SYNs to LISTEN sockets dropped
[root@lab1 ~]# netstat -s |grep socket
    311619 TCP sockets finished time wait in fast timer
    28 delayed acks further delayed because of locked socket
    638425 times the listen queue of a socket overflowed
    927922 SYNs to LISTEN sockets dropped
[root@lab1 ~]# netstat -s |grep socket
    317818 TCP sockets finished time wait in fast timer
    29 delayed acks further delayed because of locked socket
    638425 times the listen queue of a socket overflowed
    927922 SYNs to LISTEN sockets dropped
[root@lab1 ~]#
```



```
[root@lab2 ~]# wrk -c 1000 -t 2 -d 120 http://10.2.145.24
Running 2m test @ http://10.2.145.24
  2 threads and 1000 connections
  Thread Stats   Avg      Stdev     Max   +/- Stdev
    Latency   194.41ms  230.68ms 864.33ms   78.09%
    Req/Sec     7.06k     1.94k   14.45k    76.67%
  1684893 requests in 2.00m, 441.10MB read
  Non-2xx or 3xx responses: 1067409
Requests/sec:  14036.90
Transfer/sec:      3.67MB
```

再看nginx log，先前的crit還沒解決

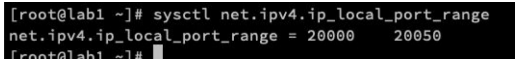
[99: Cannot assign requested address]



```
2020/12/05 08:10:45 [crit] 14#14: *328940 connect() to 127.0.0.1:9000 failed (99: Cannot assign
 requested address) while connecting to upstream, client: 10.2.145.25, server: localhost, reque
st: "GET / HTTP/1.1", upstream: "fastcgi://127.0.0.1:9000", host: "10.2.145.24"
2020/12/05 08:10:45 [crit] 13#13: *329304 connect() to 127.0.0.1:9000 failed (99: Cannot assign
 requested address) while connecting to upstream, client: 10.2.145.25, server: localhost, reque
st: "GET / HTTP/1.1", upstream: "fastcgi://127.0.0.1:9000", host: "10.2.145.24"
2020/12/05 08:10:45 [crit] 14#14: *328528 connect() to 127.0.0.1:9000 failed (99: Cannot assign
 requested address) while connecting to upstream, client: 10.2.145.25, server: localhost, reque
st: "GET / HTTP/1.1", upstream: "fastcgi://127.0.0.1:9000", host: "10.2.145.24"
2020/12/05 08:10:45 [crit] 13#13: *329205 connect() to 127.0.0.1:9000 failed (99: Cannot assign
 requested address) while connecting to upstream, client: 10.2.145.25, server: localhost, reque
st: "GET / HTTP/1.1", upstream: "fastcgi://127.0.0.1:9000", host: "10.2.145.24"
2020/12/05 08:10:45 [crit] 14#14: *328718 connect() to 127.0.0.1:9000 failed (99: Cannot assign
 requested address) while connecting to upstream, client: 10.2.145.25, server: localhost, reque
st: "GET / HTTP/1.1", upstream: "fastcgi://127.0.0.1:9000", host: "10.2.145.24"
2020/12/05 08:10:45 [crit] 14#14: *328919 connect() to 127.0.0.1:9000 failed (99: Cannot assign
 requested address) while connecting to upstream, client: 10.2.145.25, server: localhost, reque
st: "GET / HTTP/1.1", upstream: "fastcgi://127.0.0.1:9000", host: "10.2.145.24"
```

client(nginx)要連接server(php)時，配不到port

設定：sysctl net.ipv4.ip_local_port_range
增加這邊的範圍



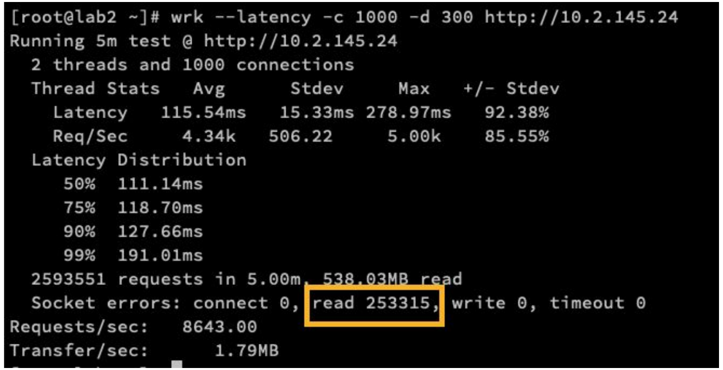
```
[root@lab1 ~]# sysctl net.ipv4.ip_local_port_range
net.ipv4.ip_local_port_range = 20000    20050
[root@lab1 ~]# 
```

再測試

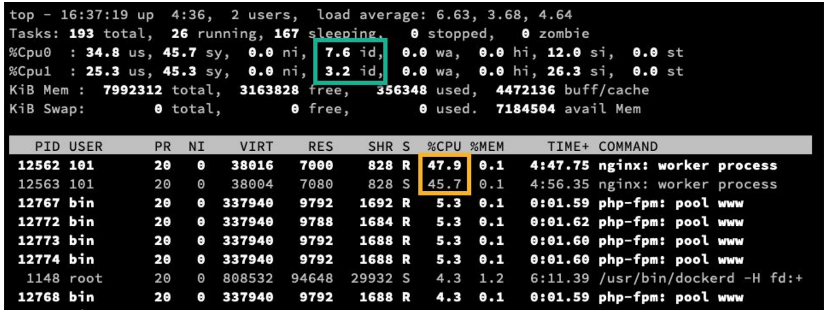
request都成功了（non-2xx & 3xx）

但再次看到socket read error

從top觀察到，大部分CPU被nginx佔用



```
[root@lab2 ~]# wrk --latency -c 1000 -d 300 http://10.2.145.24
Running 5m test @ http://10.2.145.24
  2 threads and 1000 connections
  Thread Stats   Avg      Stdev     Max   +/- Stdev
    Latency   115.54ms   15.33ms 278.97ms   92.38%
    Req/Sec     4.34k   506.22     5.00k    85.55%
  Latency Distribution
     50%  111.14ms
     75%  118.70ms
     90%  127.66ms
     99%  191.01ms
  2593551 requests in 5.00m, 538.03MB read
  Socket errors: connect 0, read 253315, write 0, timeout 0
Requests/sec:   8643.00
Transfer/sec:      1.79MB
```

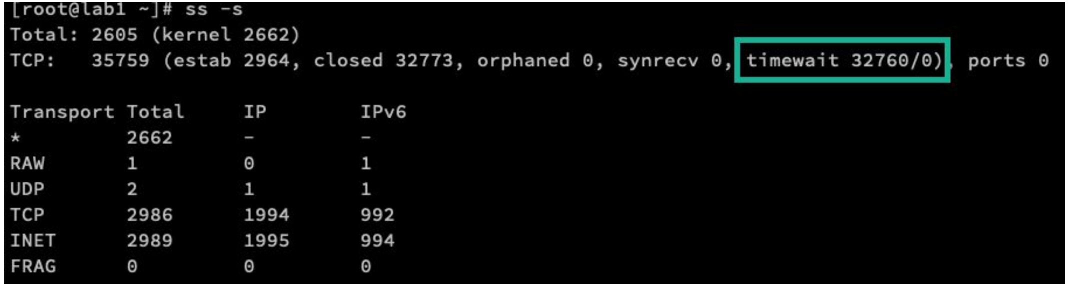


```
top - 16:37:19 up  4:36,  2 users,  load average: 6.63, 3.68, 4.64
Tasks: 193 total,  26 running, 167 sleeping,   0 stopped,   0 zombie
%Cpu0  : 34.8 us, 45.7 sy,  0.0 ni,  7.6 id,  0.0 wa,  0.0 hi, 12.0 si,  0.0 st
%Cpu1  : 25.3 us, 45.3 sy,  0.0 ni,  3.2 id,  0.0 wa,  0.0 hi, 26.3 si,  0.0 st
KiB Mem :  7992312 total,  3163828 free,   356348 used,  4472136 buff/cache
KiB Swap:        0 total,        0 free,        0 used.  7184504 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S  %CPU %MEM     TIME+ COMMAND
12562 101       20   0   38016   7000    828 R  47.9  0.1   4:47.75 nginx: worker process
12563 101       20   0   38004   7080    828 S  45.7  0.1   4:56.35 nginx: worker process
12767 bin       20   0  337940   9792   1692 R   5.3  0.1   0:01.59 php-fpm: pool www
12772 bin       20   0  337940   9788   1684 R   5.3  0.1   0:01.62 php-fpm: pool www
12773 bin       20   0  337940   9792   1688 R   5.3  0.1   0:01.60 php-fpm: pool www
12774 bin       20   0  337940   9792   1688 R   5.3  0.1   0:01.60 php-fpm: pool www
 1148 root      20   0  808532  94648  29932 S   4.3  1.2   6:11.39 /usr/bin/dockerd -H fd:+
12768 bin       20   0  337940   9792   1688 R   4.3  0.1   0:01.59 php-fpm: pool www
```

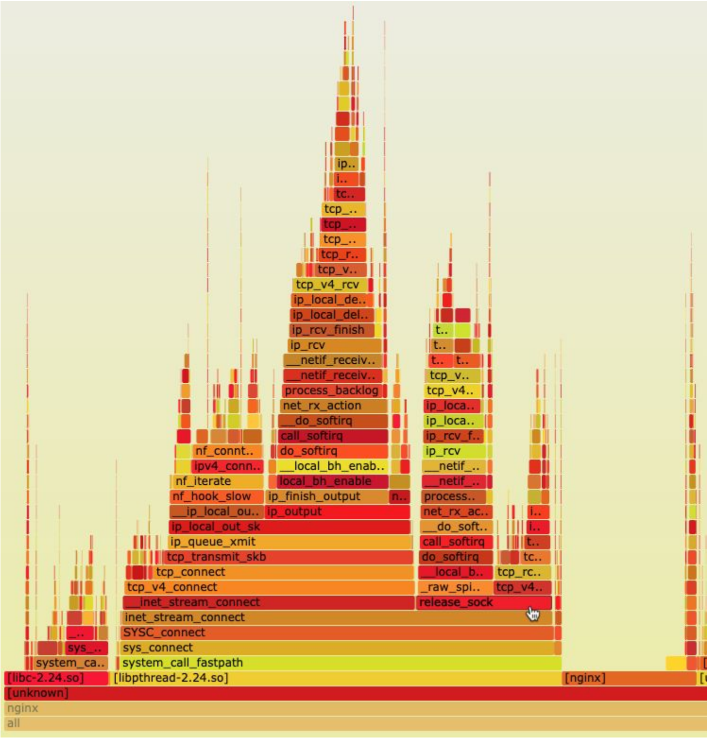
分析消耗CPU的程序-火焰圖

我觀察到的是inet_stream_connect
（這點與書中的有差異）

ss -s則觀察到有大量的等待連接



```
[root@lab1 ~]# ss -s
Total: 2605 (kernel 2662)
TCP:   35759 (estab 2964, closed 32773, orphaned 0, synrecv 0, timewait 32760/0), ports 0

Transport Total     IP        IPv6
*         2662      -         -
RAW       1         0         1
UDP       2         1         1
TCP       2986      1994      992
INET      2989      1995      994
FRAG      0         0         0
```



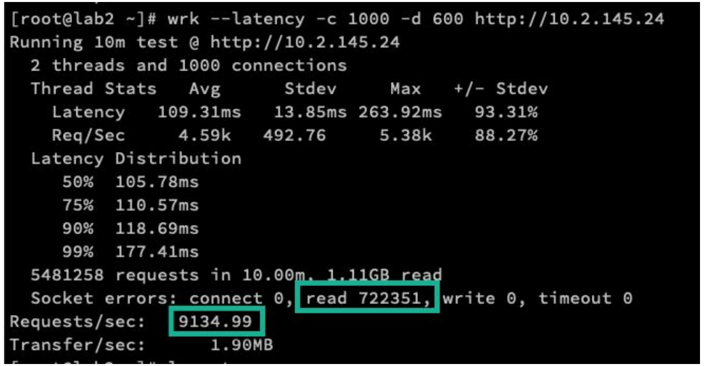
port reuse

sysctl -w net.ipv4.tcp_tw_reuse=1

書上的案例到這邊就正常了，但我的lab還沒

仍然有大量的socket read error

nginx log出現
13#13: 1024 worker_connections are not enough



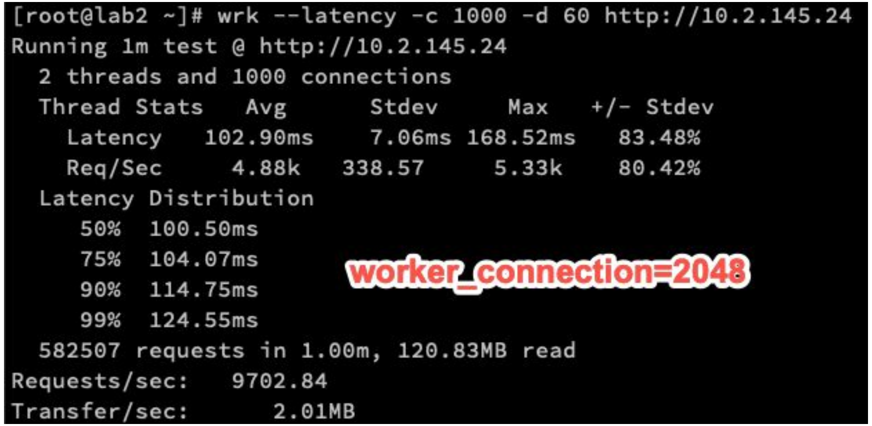
```
[root@lab2 ~]# wrk --latency -c 1000 -d 600 http://10.2.145.24
Running 10m test @ http://10.2.145.24
  2 threads and 1000 connections
  Thread Stats   Avg      Stdev     Max   +/- Stdev
    Latency   109.31ms   13.85ms 263.92ms   93.31%
    Req/Sec     4.59k   492.76     5.38k    88.27%
  Latency Distribution
     50%  105.78ms
     75%  110.57ms
     90%  118.69ms
     99%  177.41ms
  5481258 requests in 10.00m, 1.11GB read
  Socket errors: connect 0, read 722351, write 0, timeout 0
Requests/sec:   9134.99
Transfer/sec:      1.90MB
```

worker_connection

nginx config預設1024

調到2048

再測試，回應就正常了



```
[root@lab2 ~]# wrk --latency -c 1000 -d 60 http://10.2.145.24
Running 1m test @ http://10.2.145.24
  2 threads and 1000 connections
  Thread Stats   Avg      Stdev     Max   +/- Stdev
    Latency   102.90ms    7.06ms 168.52ms   83.48%
    Req/Sec     4.88k   338.57     5.33k    80.42%
  Latency Distribution
     50%  100.50ms
     75%  104.07ms
     90%  114.75ms
     99%  124.55ms
  582507 requests in 1.00m, 120.83MB read
Requests/sec:   9702.84
Transfer/sec:      2.01MB
```

worker_connection=2048