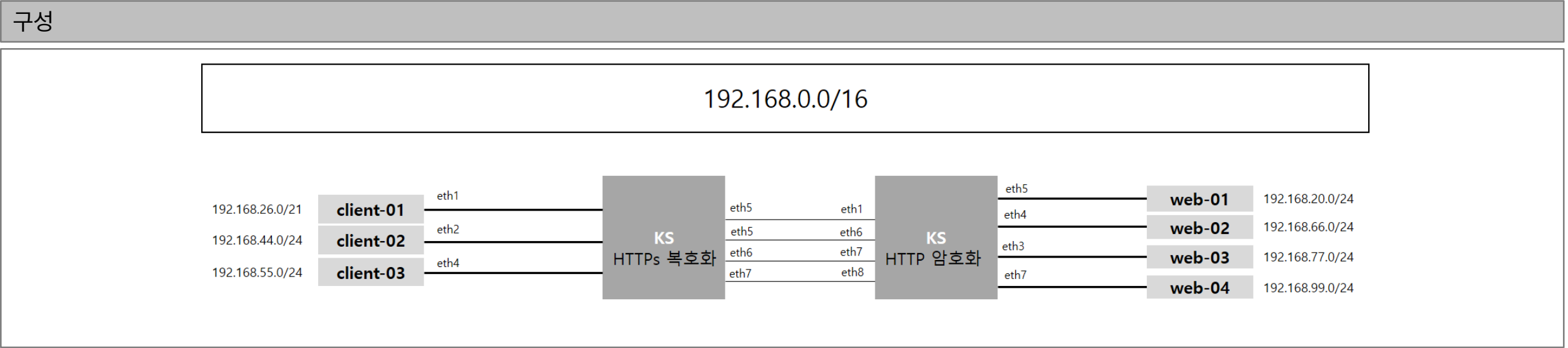


PoC 사내 측정 결과

hs.bang@piolink.com

구성 및 네트워크 정보



서브넷 네트워크 및 인스턴스 OS 정보

VPC : 192.168.0.0/16
Subnet:
- 클라이언트 인스턴스: /24prefix 3개
KS 간 인터링크: /24prefix 4개
서버 인스턴스: /24prefix 4개
인스턴스 OS : Linux CentOS7
KS PLOS : 2.2.5.7.5

클라이언트/서버 인스턴스 공통 적용 설정

conntrack 비활성화	TCP 성능 변수	소켓 개수 확장
<pre>#!/bin/bash chains="PREROUTING OUTPUT" conditions="dport sport" ports="80 443" iptables -F -t raw for chain in \$chains; do for condition in \$conditions; do for port in \$ports; do iptables -t raw -I \$chain -p tcp --\${condition} \$port -j NOTRACK done done done</pre>	<pre>cat <<EOF tee /etc/sysctl.conf net.ipv4.tcp_window_scaling=1 net.core.rmem_default=253952 net.core.wmem_default=253952 net.core.rmem_max=16777216 net.core.wmem_max=16777216 net.ipv4.tcp_rmem=253952 253952 16777216 net.ipv4.tcp_wmem=253952 253952 16777216 net.core.netdev_max_backlog=30000 net.core.somaxconn=1024 net.ipv4.tcp_max_syn_backlog = 1024 net.ipv4.ip_local_port_range = 1024 65535 net.ipv4.tcp_tw_reuse = 1 net.ipv4.tcp_timestamps = 1 net.ipv4.tcp_max_tw_buckets = 1800000 net.ipv4.tcp_timestamps = 1 net.ipv4.tcp_tw_reuse = 1 EOF sudo sysctl --system</pre>	<pre>ulimit -n 250000</pre>

클라이언트 loop-back IP주소 설정

loop-back IP x.x.x.100-200 생성
<pre>#!/bin/bash host=`ifconfig -a grep inet grep broadcast awk '{print \$2}' sed 's/[0-9]\+\$//g` for i in {100..200} do echo \$i echo -e DEVICE=lo:\$i\nBOOTPROTO=static\nIPADDR=\$host\$i\nNETMASK=255.255.255.255\nONBOOT=yes\nNAME=loopback > /etc/sysconfig/network-scripts/ifcfg-lo:\$i done service network restart</pre>

WEB 인스턴스 설정

nginx conf

```
server {  
  listen    443 ssl;  
  server_name localhost;  
  server_name  pio.com;  
  ssl_certificate /root/ssl/nginx-ssl.crt;  
  ssl_certificate_key /root/ssl/nginx-ssl.key;  
  access_log /var/log/nginx/host.access.log main;  
  root /var/www/chat;  
  index index.html index.htm;  
  location / {  
    try_files $uri $uri/ = 404;  
  }  
  error_page 500 502 503 504 /50x.html;  
  location = /50x.html {  
    root /usr/share/nginx/html;  
  }  
}
```

html

128kb.html
1kb.html

텍스트로 채워서 설정

KS 설정

/opt/nginx/conf/process.conf

서버 RX 부분 0번 큐에 쏘리는 현상 때문에 0번 워커 프로세스는 없음

```
## 8vCPU
worker_processes 8;
worker_cpu_affinity 00000100 00000010 00000100 00001000 00010000 00100000 01000000 10000000;

## 16vCPU
worker_processes 16;
worker_cpu_affinity 00000000000000100 00000000000000010 00000000000000100 00000000000001000 00000000000010000 0000000000100000 0000000001000000 0000000001000000 000000000100000000
00000001000000000 00000010000000000 00000100000000000 00001000000000000 00010000000000000 00100000000000000 01000000000000000 10000000000000000 ;
```

KS-EXT

```
advl7cslb c1
group vip
ssl 1
ssl-detection enable
apply
filter 10
dport 443
apply
lb-method type sh
connection-pooling mode aggressive
real 100
real 101
real 102
real 103
apply
backend-timeout 1
apply
rule 1
group vip
adapt-location-scheme disable
backend-timeout 1
apply
apply
exit
```

KS-INT

```
!
advl7slb s1
vip 192.168.30.100 protocol http vport 443
return-to-sender enable
apply
backend-timeout 1
apply
rts-real 1
ip 192.168.98.101
interface inter
apply
rts-real 2
ip 172.17.1.101
interface inter-01
apply
rts-real 3
ip 172.17.2.101
interface inter-02
apply
rts-real 4
ip 192.168.88.85
interface inter-03
apply
group web
lb-method type sh
connection-pooling mode aggressive
real 20
real 66
real 77
real 99
apply
rule 1
group web
request-scheme https
backend-timeout 1
backend-ssl profile 2
backend-ssl sni disable
apply
apply
```

K6 성능 측정 script

cps.js

```
import http from "k6/http";
import { check, sleep } from "k6";
import { Rate } from "k6/metrics";

export const options = {
  noConnectionReuse: true, ## VU 커넥션 재사용 여부
  vus: 60, ## 트래픽 전송할 클라이언트 개수
  duration: '3m'
}

var script_errors = Rate("script_errors");

function wrapWithErrorCounting(fn) {
  return (data) => {
    try {
      fn(data);
      script_errors.add(0);
    } catch (e) {
      script_errors.add(1);
      throw e;
    }
  }
}

function simpleTest() {
  let response = http.get("https://192.168.30.100/1kb.html"); ## 전송 대상
  check(response, { ## 성공 판단 여부
    "200 OK": (r) => r.status === 200,
  });
}

export default wrapWithErrorCounting(simpleTest);
```

tps.js

```
import http from "k6/http";
import { check, sleep } from "k6";
import { Rate } from "k6/metrics";
export const options = {
  vus: 250,
  noVUConnectionReuse: true,
  duration: '3m'
}

var script_errors = Rate("script_errors");
function wrapWithErrorCounting(fn) {
  return (data) => {
    try {
      fn(data);
      script_errors.add(0);
    } catch (e) {
      script_errors.add(1);
      throw e;
    }
  }
}

function simpleTest() {
  for (let id = 1; id <= 10; id++) {
    let response = http.get("https://192.168.30.100/1kb.html");
    check(response, {
      "200 OK": (r) => r.status === 200,
    });
  }
}

export default wrapWithErrorCounting(simpleTest);
```

K6 성능 측정 script

throughput.js

```
import http from "k6/http";
import { check, sleep } from "k6";
import { Rate } from "k6/metrics";
export const options = {
  vus: 150,
  noVUConnectionReuse: true,
  duration: '3m'
}
var script_errors = Rate("script_errors");
function wrapWithErrorCounting(fn) {
  return (data) => {
    try {
      fn(data);
      script_errors.add(0);
    } catch (e) {
      script_errors.add(1);
      throw e;
    }
  }
}

function simpleTest() {
  for (let id = 1; id <= 10; id++) {
    let response = http.get("https://192.168.30.100/100kb.html");
    check(response, {
      "200 OK": (r) => r.status === 200,
    });
  }
}

export default wrapWithErrorCounting(simpleTest);
```

실행 명령어

ex)
k6 run ~/script/https/cps.js --local-ips=192.168.26.100-192.168.26.200 --insecure-skip-tls-verify

설명)
k6 run <스크립트 경로> --local-ips=<local ip 주소 범위> --<https 경우 옵션 사용>

**** k6 옵션 설명 : <https://k6.io/docs/using-k6/k6-options/reference/>**

NHN PoC 프로젝트 내 성능 측정 결과

EXT-ks HTTPS 복호화	cps						rps						throughput						throughput + ssl decryption mirroring enable					
	type	vcpu/mem	vus/vip/real	resource	size	result(max.)	type	vcpu/mem	vus/vip/real	resource	size	result(max.)	type	vcpu/mem	vus/vip/real	resource	size	result(max.)	type	vcpu/mem	vus/vip/real	resource	size	result(max.)
	client	8vCPU/16GB	60*3	avg 79%	1kb	25739	client	8vCPU/16GB	250 *3	avg 65%	1kb	112105	client	8vCPU/16GB	150*3	avg 35%	32kb	12G	client	8vCPU/16GB	50*3	avg 35%	32kb	9G
	ks	16vCPU/32GB	1	94%			ks	16vCPU/32GB	1	92%			ks	16vCPU/32GB	1	64%			ks	16vCPU/32GB	1	83%		
	nginx	4vCPU/8GB	4	avg 3%			nginx	4vCPU/8GB	4	avg 9%			nginx	4vCPU/8GB	4	avg 12%			nginx	4vCPU/8GB	4	avg 15%		
	1Conn 1 Get / advl7slb Conn pooling per-client						1Conn 10 Get / advl7slb Conn pooling per-client						1Conn 10 Get / Conn pooling per-client						1Conn 10 Get / Conn pooling per-client					
	cps						rps						throughput						throughput					
	type	vcpu/mem	vus/vip/real	resource	size	result(max.)	type	vcpu/mem	vus/vip/real	resource	size	result(max.)	type	vcpu/mem	vus/vip/real	resource	size	result(max.)	type	vcpu/mem	vus/vip/real	resource	size	result(max.)
	client	8vCPU/16GB	60*3	avg 35%	1kb	12173	client	8vCPU/16GB	250*3	avg 45%	1kb	62017	client	8vCPU/16GB	150*3	avg 49%	32kb	7G	client	8vCPU/16GB	50*3	avg 20%	32kb	6G
	ks	8vCPU/16GB	1	93%			ks	8vCPU/16GB	1	99%			ks	8vCPU/16GB	1	86%			ks	8vCPU/16GB	1	70%		
nginx	4vCPU/8GB	4	avg 3%	nginx			4vCPU/8GB	4	19%	nginx			4vCPU/8GB	4	avg 9%	nginx			4vCPU/8GB	5	avg 12%			
1Conn 1 Get / advl7slb Conn pooling per-client						1Conn 10 Get / advl7slb Conn pooling per-client						1Conn 10 Get / Conn pooling per-client						1Conn 10 Get / Conn pooling per-client						
INT-ks HTTP 암호화	cps						rps						throughput						throughput					
	type	vcpu/mem	vus/vip/real	resource	size	result(max.)	type	vcpu/mem	vus/vip/real	resource	size	result(max.)	type	vcpu/mem	vus/vip/real	resource	size	result(max.)	type	vcpu/mem	vus/vip/real	resource	size	result(max.)
	client	8vCPU/16GB	60*3	avg 49%	1kb	40133	client	8vCPU/16GB	250*3	avg 55%	1kb	133071	client	8vCPU/16GB	150*3	avg 36%	32kb	12G	client	8vCPU/16GB	50*3	avg 36%	32kb	9G
	ks	16vCPU/32GB	1	42%			ks	16vCPU/32GB	1	95%			ks	16vCPU/32GB	1	44%			ks	16vCPU/32GB	1	70%		
	nginx	4vCPU/8GB	4	avg 3%			nginx	4vCPU/8GB	4	avg 15%			nginx	4vCPU/8GB	4	avg 22%			nginx	4vCPU/8GB	4	avg 15%		
	1Conn 1 Get / advl7slb Conn pooling aggressive						1Conn 10 Get / advl7slb Conn pooling aggressive						1Conn 10 Get / advl7slb Conn pooling per-client						1Conn 10 Get / advl7slb Conn pooling per-client					
	cps						rps						throughput						throughput + ssl decryption mirroring enable					
	type	vcpu/mem	vus/vip/real	resource	size	result(max.)	type	vcpu/mem	vus/vip/real	resource	size	result(max.)	type	vcpu/mem	vus/vip/real	resource	size	result(max.)	type	vcpu/mem	vus/vip/real	resource	size	result(max.)
	client	8vCPU/16GB	60*3	avg 35%	1kb	12173	client	8vCPU/16GB	250*3	avg 46%	1kb	62017	client	8vCPU/16GB	150*3	avg 20%	32kb	7G	client	8vCPU/16GB	50*3	avg 20%	32kb	6G
	ks	8vCPU/16GB	1	30%			ks	8vCPU/16GB	1	77%			ks	8vCPU/16GB	1	76%			ks	8vCPU/16GB	1	52%		
nginx	4vCPU/8GB	4	avg 3%	nginx			4vCPU/8GB	4	avg 19%	nginx			4vCPU/8GB	4	avg 10%	nginx			4vCPU/8GB	4	avg 15%			
1Conn 1 Get / advl7slb Conn pooling per-client						1Conn 10 Get / advl7slb Conn pooling per-client						1Conn 10 Get / advl7slb Conn pooling per-client						1Conn 10 Get / advl7slb Conn pooling per-client						

K6 성능 툴 프로메테우스매트릭으로 그래파나 대시보드 연동

참고 사이트

- <https://sanggi-jayg.tistory.com/91> (설치 관련)
- <https://velog.io/@kimhalin/%EC%84%9C%EB%B2%84-%EB%B6%80%ED%95%98-%ED%85%8C%EC%8A%A4%ED%8A%B8-K6>
- <https://devocean.sk.com/blog/techBoardDetail.do?ID=164237>

K6 성능 툴 influxdb / 그래파나 대시보드 연동

참고자료

- <https://velog.io/@heka1024/Grafana-k6%EC%9C%BC%EB%A1%9C-%EB%B6%80%ED%95%98-%ED%85%8C%EC%8A%A4%ED%8A%B8%ED%95%98%EA%B8%B0>

PoC 결과

hs.bang@piolink.com

성능 테스트 결과 - Summary

항목	소항목	측정 기준	비고	결과		
				측정 수치	CPU 사용량	MEM 사용량
1. 8 코어 인스턴스	1.1. CPS	<ul style="list-style-type: none"> 요청 형태 : 1 GET / 1 Connection 응답 사이즈 : 1KB 		13,650	91.4 %	12.8 %
	1.2. TPS	<ul style="list-style-type: none"> 요청 형태 : 10 GET / 1 Connection 응답 사이즈 : 1KB 		77,024	95.1 %	12.13 %
	1.3. Throughput	<ul style="list-style-type: none"> 요청 형태 : 10 GET / 1 Connection 응답 사이즈 : 100KB 	복호화 미러링 X	11 Gbps	63 %	11 %
	1.4. Throughput	<ul style="list-style-type: none"> 요청 형태 : 10 GET / 1 Connection 응답 사이즈 : 100KB 	복호화 미러링 O	10 Gbps	100 %	12 %
2. 16 코어 인스턴스	2.1. CPS	<ul style="list-style-type: none"> 요청 형태 : 1 GET / 1 Connection 응답 사이즈 : 1KB 		24,807	92.7 %	12.33 %
	2.2. TPS	<ul style="list-style-type: none"> 요청 형태 : 10 GET / 1 Connection 응답 사이즈 : 1KB 		124,587	92.9 %	11.84 %
	2.3. Throughput	<ul style="list-style-type: none"> 요청 형태 : 10 GET / 1 Connection 응답 사이즈 : 100KB 	복호화 미러링 X	15 Gbps	57 %	11 %
	2.4. Throughput	<ul style="list-style-type: none"> 요청 형태 : 10 GET / 1 Connection 응답 사이즈 : 100KB 	복호화 미러링 O	14 Gbps	95 %	11 %

대항목	1. 8 코어 인스턴스	시험 형태	1 GET / 1 Connection / 응답 사이즈 1 KB										
중항목	1.1. CPS	비고											
시험 결과		상세 결과											
<table><tr><th>응답사이즈</th><th>항목</th><th>측정값</th></tr><tr><td rowspan="3">1KB</td><td>CPS</td><td>13,650</td></tr><tr><td>CPU Usage</td><td>91.4 %</td></tr><tr><td>MEM Usage</td><td>12.8 %</td></tr></table>		응답사이즈	항목	측정값	1KB	CPS	13,650	CPU Usage	91.4 %	MEM Usage	12.8 %	<p>○ PAS-KS 에서 확인한 CPS 처리 현황</p> <div><pre>===== ADVL7CSLB: c1 ----- RPS/Peak : 13650/13969 Requests : 2259410 Cps : 13650 ----- Current sessions Client : 17 Server : 102 Total : 119 =====</pre></div> <p>○ PAS-KS 에서 확인한 CPU / MEM 사용량</p> <div><pre>ks-ext(config)# sho resource ===== RESOURCE ----- CPU Usage : 91.4% ----- Memory Management Total : 6144 MB Used : 1083 MB Free : 5060 MB Usage : 17.64% ----- Packet Total : 9935 MB Used : 1276 MB Free : 8659 MB Usage : 12.85% ----- Log Storage Total : 42845 MB Used : 4960 MB Free : 35709 MB Usage : 13% =====</pre></div>	
응답사이즈	항목	측정값											
1KB	CPS	13,650											
	CPU Usage	91.4 %											
	MEM Usage	12.8 %											

대항목	1.8 코어 인스턴스	시험 형태	1 GET / 1 Connection / 응답 사이즈 1 KB
중항목	1.1. CPS	비고	

상세 결과

○ 계측 클라이언트 처리 결과 화면

[클라이언트01]

```

1 root@client-01:~/sh x 2 root@client-02:~/sh x 3 root@client-03:~/sh x
script: /root/script/https/cps.js
output: -

scenarios: (100.00%) 1 scenario, 60 max VUs, 3m30s max duration (incl. graceful st
op):
    * default: 60 looping VUs for 3m0s (gracefulStop: 30s)

✓ 200 OK
checks.....: 100.00% ✓ 811211 x 0
data_received.....: 1.8 GB 9.8 MB/s
data_sent.....: 419 MB 2.3 MB/s
http_req_blocked.....: avg=10.78ms min=1.11ms med=10.5ms max=56.0
4ms p(90)=11.13ms p(95)=11.47ms
http_req_connecting.....: avg=238.14µs min=116.1µs med=202.4µs max=16.8
1ms p(90)=333.84µs p(95)=423.32µs
http_req_duration.....: avg=2.46ms min=825.98µs med=2.08ms max=61.5
9ms p(90)=2.56ms p(95)=2.98ms
{ expected_response:true }...: avg=2.46ms min=825.98µs med=2.08ms max=61.5
9ms p(90)=2.56ms p(95)=2.98ms
http_req_failed.....: 0.00% ✓ 0 x 811211
http_req_receiving.....: avg=62.68µs min=12.45µs med=
1ms p(90)=73.8µs p(95)=126.18µs
http_req_sending.....: avg=44.91µs min=7.65µs med=
ms p(90)=47.25µs p(95)=130.87µs
http_req_tls_handshaking.....: avg=10.5ms min=889.56µs med=10.24ms max=55.7
1ms p(90)=10.87ms p(95)=11.21ms
http_req_waiting.....: avg=2.35ms min=751.59µs med=2ms max=61.4
9ms p(90)=2.42ms p(95)=2.69ms
http_reqs.....: 811211 4506.500073/s
iteration_duration.....: avg=13.3ms min=3.64ms med=12.66ms max=71.0
9ms p(90)=13.68ms p(95)=14.96ms
iterations.....: 811211 4506.500073/s
script_errors.....: 0.00% ✓ 0 x 811211
vus.....: 60 min=60 max=60
vus_max.....: 60 min=60 max=60

running (3m00.0s), 00/60 VUs, 811211 complete and 0 interrupted iterations
default ✓ [=====] 60 VUs 3m0s

```

[클라이언트02]

```

1 root@client-01:~/sh x 2 root@client-02:~/sh x 3 root@client-03:~/sh x 4 KS-ext x +
✓ 200 OK
checks.....: 100.00% ✓ 808699 x 0
data_received.....: 1.8 GB 9.8 MB/s
data_sent.....: 418 MB 2.3 MB/s
http_req_blocked.....: avg=10.8ms min=1.26ms med=10.52ms max=53.61ms p(9
0)=11.15ms p(95)=11.5ms
http_req_connecting.....: avg=248.97µs min=122.32µs med=209.59µs max=22.97ms p(9
0)=348.07µs p(95)=442.75µs
http_req_duration.....: avg=2.48ms min=803.53µs med=2.09ms max=61.53ms p(9
0)=2.58ms p(95)=3.08ms
{ expected_response:true }...: avg=2.48ms min=803.53µs med=2.09ms max=61.53ms p(9
0)=2.58ms p(95)=3.08ms
http_req_failed.....: 0.00% ✓ 0 x 808699
http_req_receiving.....: avg=70.26µs min=14.44µs med=45.31µs max=16.63ms p(9
0)=78.4µs p(95)=144.6µs
http_req_sending.....: avg=52.72µs min=9.31µs med=21.27µs max=15.97ms p(9
0)=62.72µs p(95)=144.6µs
http_req_waiting.....: avg=2.35ms min=727.44µs med=2ms max=61.43ms p(9
0)=2.42ms p(95)=2.71ms
http_reqs.....: 808699 4492.41101/s
iteration_duration.....: avg=13.34ms min=2.99ms med=12.69ms max=72.59ms p(9
0)=13.73ms p(95)=15.19ms
iterations.....: 808699 4492.41101/s
script_errors.....: 0.00% ✓ 0 x 808699
vus.....: 60 min=60 max=60
vus_max.....: 60 min=60 max=60

running (3m00.0s), 00/60 VUs, 808699 complete and 0 interrupted iterations
default ✓ [=====] 60 VUs 3m0s

```

[클라이언트03]

```

1 root@client-01:~/sh x 2 root@client-02:~/sh x 3 root@client-03:~/sh x 4 KS-ext x +
✓ 200 OK
checks.....: 100.00% ✓ 810818 x 0
data_received.....: 1.8 GB 9.8 MB/s
data_sent.....: 419 MB 2.3 MB/s
http_req_blocked.....: avg=10.77ms min=1.18ms med=10.51ms max=55.43ms p(90)=11.15ms p(95)=11
5ms
http_req_connecting.....: avg=248.22µs min=115.34µs med=208.21µs max=20.1ms p(90)=351.21µs p(95)=44
.28µs
http_req_duration.....: avg=2.47ms min=595.32µs med=2.08ms max=61.82ms p(90)=2.58ms p(95)=3.
4ms
{ expected_response:true }...: avg=2.47ms min=595.32µs med=2.08ms max=61.82ms p(90)=2.58ms p(95)=3.
4ms
http_req_failed.....: 0.00% ✓ 0 x 810818
http_req_receiving.....: avg=68.61µs min=11.83µs med=43.74µs max=15.1ms p(90)=82.39µs p(95)=14
.77µs
http_req_sending.....: avg=50.49µs min=7.62µs med=20.15µs max=15.06ms p(90)=58.22µs p(95)=15
ms
http_req_tls_handshaking.....: avg=10.49ms min=934.16µs med=10.24ms max=55.22ms p(90)=10.88ms p(95)=11
ms
http_req_waiting.....: avg=2.35ms min=537.56µs med=2ms max=61.33ms p(90)=2.42ms p(95)=2.
ms
http_reqs.....: 810818 4504.471047/s
iteration_duration.....: avg=13.31ms min=2.25ms med=12.68ms max=75.32ms p(90)=13.71ms p(95)=15
1ms
iterations.....: 810818 4504.471047/s
script_errors.....: 0.00% ✓ 0 x 810818
vus.....: 60 min=60 max=60
vus_max.....: 60 min=60 max=60

running (3m00.0s), 00/60 VUs, 810818 complete and 0 interrupted iterations
default ✓ [=====] 60 VUs 3m0s

```

200 OK 를 100% 수신 → PAS-KS 를 통한 요청에 문제가 없음을 의미

대항목	1. 8 코어 인스턴스	시험 형태	10 GET / 1 Connection / 응답 사이즈 1 KB										
중항목	1.2. TPS	비고											
시험 결과		상세 결과											
<table><tr><th>응답사이즈</th><th>항목</th><th>측정값</th></tr><tr><td rowspan="3">1KB</td><td>TPS</td><td>77,024</td></tr><tr><td>CPU Usage</td><td>95.1 %</td></tr><tr><td>MEM Usage</td><td>12.13 %</td></tr></table>		응답사이즈	항목	측정값	1KB	TPS	77,024	CPU Usage	95.1 %	MEM Usage	12.13 %	<p>○ PAS-KS 에서 확인한 TPS 처리 현황</p> <pre>===== ADV7CSLB: c1 ----- RPS/Peak : 77024/77985 Requests : 1180124 Cps : 7687 Current sessions Client : 332 Server : 384 Total : 716</pre> <p>○ PAS-KS 에서 확인한 CPU / MEM 사용량</p> <pre>ks-ext(config)# sho resource ===== RESOURCE ----- CPU Usage : 95.11% Memory Management Total : 6144 MB Used : 1083 MB Free : 5060 MB Usage : 17.63% Packet Total : 9935 MB Used : 1205 MB Free : 8730 MB Usage : 12.13% Log Storage Total : 42845 MB Used : 4968 MB Free : 35701 MB Usage : 13% =====</pre>	
응답사이즈	항목	측정값											
1KB	TPS	77,024											
	CPU Usage	95.1 %											
	MEM Usage	12.13 %											

대항목	1. 8 코어 인스턴스	시험 형태	10 GET / 1 Connection / 응답 사이즈 1 KB
중항목	1.2. TPS	비고	

상세 결과

○ 계측 클라이언트 처리 결과 화면

[클라이언트01]

[클라이언트02]

[클라이언트03]

```

[KS-ext] 2 root@client-01:~#
[KS-ext] 2 root@client-02:~#
[KS-ext] 4 root@client-03:~#

[root@client-01 sh]#
[root@client-01 sh]# sh rps.sh

A K G
.10

execution: local
script: /root/script/https/rps.js
output: -

scenarios: (100.00%) 1 scenario, 250 max VUs, 3m30s max duration (incl. graceful stop):
* default: 250 looping VUs for 3m0s (gracefulStop: 30s)

✓ 200 OK

checks.....: 100.00% / 4640570 x 0
data_received.....: 6.6 GB 37 MB/s
data_sent.....: 720 MB 4 MB/s
http_req_blocked.....: avg=5.18ms min=574ns med=1.03µs max=110.42ms p(90)=14.62µs p(95)=33.55µs
http_req_connecting.....: avg=36.96µs min=0s med=0s max=31.19ms p(90)=90.75µs p(95)=54.21ms
http_req_duration.....: avg=4.45ms min=527.66µs med=3.34ms max=89.02ms p(90)=6.02ms p(95)=10.71ms
{ expected_response:true }.....: avg=4.45ms min=527.66µs med=3.34ms max=89.02ms p(90)=6.02ms p(95)=10.71ms
http_req_failed.....: 0.00% / 0 x 4640570
http_req_receiving.....: avg=22.22µs min=9.42µs med=16.85µs max=44.94ms p(90)=27.57µs p(95)=33.55µs
http_req_sending.....: avg=20.43µs min=2.94µs med=5.17µs max=25.98ms p(90)=14.62µs p(95)=22.23µs
http_req_tls_handshaking.....: avg=5.14ms min=0s med=0s max=110.18ms p(90)=90.75µs p(95)=54.21ms
http_req_waiting.....: avg=4.41ms min=476.22µs med=3.3ms max=88.99ms p(90)=6.02ms p(95)=10.71ms
http_reqs.....: 4640570 25768.567266/s
iteration_duration.....: avg=96.95ms min=29.16ms med=92.39ms max=298.87ms p(90)=123.7ms p(95)=138.97ms
iterations.....: 464057 2576.856727/s
script_errors.....: 0.00% / 0 x 464057
vus.....: 250 min=250 max=250
vus_max.....: 250 min=250 max=250

running (3m00.1s), 000/250 VUs, 464057 complete and 0 interrupted iterations
default ✓ [=====] 250 VUs 3m0s

```

```

[KS-ext] 2 root@client-01:~#
[KS-ext] 2 root@client-02:~#
[KS-ext] 4 root@client-03:~#

[root@client-02 sh]#
[root@client-02 sh]# sh rps.sh

A K G
.10

execution: local
script: /root/script/https/rps.js
output: -

scenarios: (100.00%) 1 scenario, 250 max VUs, 3m30s max duration (incl. graceful stop):
* default: 250 looping VUs for 3m0s (gracefulStop: 30s)

✓ 200 OK

checks.....: 100.00% / 4549260 x 0
data_received.....: 6.5 GB 36 MB/s
data_sent.....: 703 MB 3.9 MB/s
http_req_blocked.....: avg=5.23ms min=599ns med=1.06µs max=1.06ms p(90)=1.56ms p(95)=54.91ms
http_req_connecting.....: avg=47.8µs min=0s med=0s max=1s p(90)=12.26µs p(95)=245.31µs
http_req_duration.....: avg=4.63ms min=531.91µs med=3.44ms max=74.73ms p(90)=6.65ms p(95)=11.41ms
{ expected_response:true }.....: avg=4.63ms min=531.91µs med=3.44ms max=74.73ms p(90)=6.65ms p(95)=11.41ms
http_req_failed.....: 0.00% / 0 x 4549260
http_req_receiving.....: avg=25.49µs min=9.99µs med=19.04µs max=36.41ms p(90)=31.01µs p(95)=37.6µs
http_req_sending.....: avg=25.43µs min=3.01µs med=5.76µs max=31.42ms p(90)=16.91µs p(95)=26.7µs
http_req_tls_handshaking.....: avg=5.18ms min=0s med=0s max=123.65ms p(90)=94.02µs p(95)=54.47ms
http_req_waiting.....: avg=4.59ms min=498.99µs med=3.4ms max=74.69ms p(90)=6.54ms p(95)=11.27ms
http_reqs.....: 4549260 25176.39184/s
iteration_duration.....: avg=99.22ms min=30.23ms med=94.13ms max=1.16s p(90)=126.82ms p(95)=142.78ms
iterations.....: 454926 2517.639184/s
script_errors.....: 0.00% / 0 x 454926
vus.....: 250 min=250 max=250
vus_max.....: 250 min=250 max=250

running (3m00.0s), 000/250 VUs, 454926 complete and 0 interrupted iterations
default ✓ [=====] 250 VUs 3m0s

```

```

[KS-ext] 2 root@client-01:~#
[KS-ext] 2 root@client-02:~#
[KS-ext] 4 root@client-03:~#

[root@client-03 sh]# sh rps.sh

A K G
.10

execution: local
script: /root/script/https/rps.js
output: -

scenarios: (100.00%) 1 scenario, 250 max VUs, 3m30s max duration (incl. graceful stop):
* default: 250 looping VUs for 3m0s (gracefulStop: 30s)

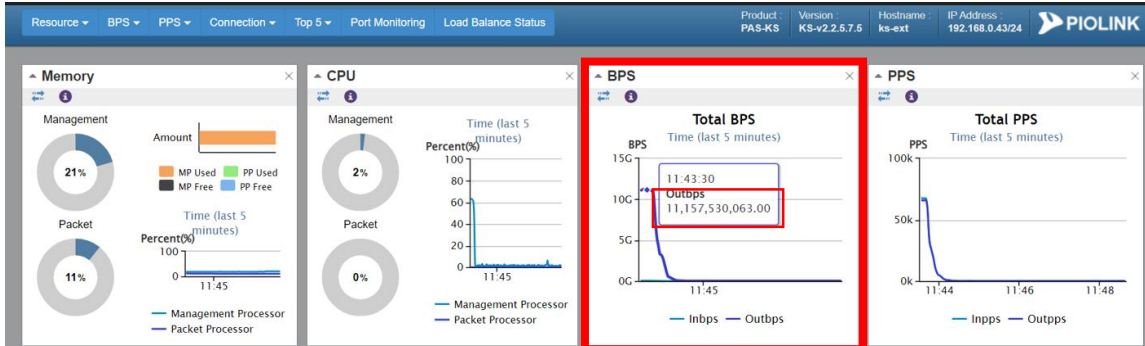
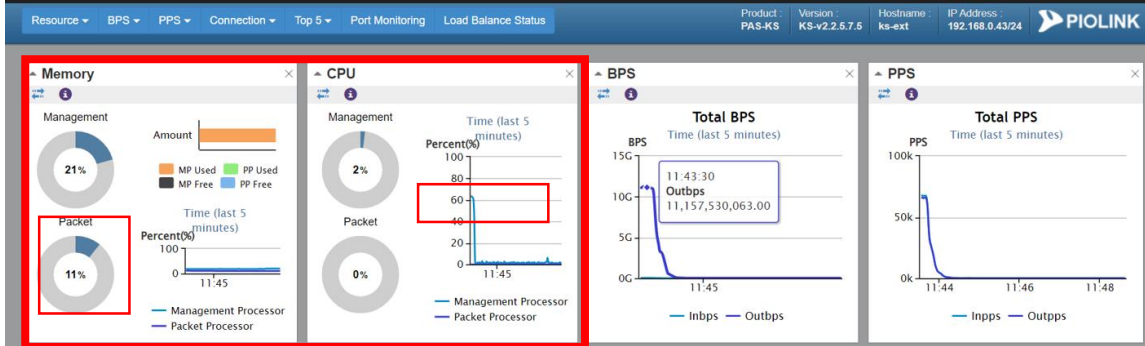
✓ 200 OK

checks.....: 100.00% / 4534390 x 0
data_received.....: 6.5 GB 36 MB/s
data_sent.....: 703 MB 3.9 MB/s
http_req_blocked.....: avg=5.23ms min=599ns med=1.06µs max=1.06ms p(90)=1.56ms p(95)=54.91ms
http_req_connecting.....: avg=47.8µs min=0s med=0s max=1s p(90)=12.26µs p(95)=245.31µs
http_req_duration.....: avg=4.63ms min=531.91µs med=3.44ms max=74.73ms p(90)=6.65ms p(95)=11.41ms
{ expected_response:true }.....: avg=4.63ms min=531.91µs med=3.44ms max=74.73ms p(90)=6.65ms p(95)=11.41ms
http_req_failed.....: 0.00% / 0 x 4534390
http_req_receiving.....: avg=25.49µs min=9.99µs med=19.04µs max=36.41ms p(90)=31.01µs p(95)=37.6µs
http_req_sending.....: avg=25.43µs min=3.01µs med=5.76µs max=31.42ms p(90)=16.91µs p(95)=26.7µs
http_req_tls_handshaking.....: avg=5.18ms min=0s med=0s max=123.65ms p(90)=94.02µs p(95)=54.47ms
http_req_waiting.....: avg=4.59ms min=498.99µs med=3.4ms max=74.69ms p(90)=6.54ms p(95)=11.27ms
http_reqs.....: 4534390 25176.39184/s
iteration_duration.....: avg=99.22ms min=30.23ms med=94.13ms max=1.16s p(90)=126.82ms p(95)=142.78ms
iterations.....: 453439 2517.639184/s
script_errors.....: 0.00% / 0 x 453439
vus.....: 250 min=250 max=250
vus_max.....: 250 min=250 max=250

running (3m00.1s), 000/250 VUs, 453439 complete and 0 interrupted iterations
default ✓ [=====] 250 VUs 3m0s

```

200 OK 를 100% 수신 → PAS-KS 를 통한 요청에 문제가 없음을 의미

대항목	1. 8 코어 인스턴스	시험 형태	10 GET / 1 Connection / 응답 사이즈 100 KB										
중항목	1.3. Throughput	비고	SSL 복호화 미러링 X										
시험 결과		상세 결과											
<table><tr><th>응답사이즈</th><th>항목</th><th>측정값</th></tr><tr><td rowspan="3">100 KB</td><td>Throughput</td><td>11 Gbps</td></tr><tr><td>CPU Usage</td><td>63 %</td></tr><tr><td>MEM Usage</td><td>11 %</td></tr></table>		응답사이즈	항목	측정값	100 KB	Throughput	11 Gbps	CPU Usage	63 %	MEM Usage	11 %	<p>○ PAS-KS 에서 확인한 Throughput 처리 현황</p>  <p>○ PAS-KS 에서 확인한 CPU / MEM 사용량</p> 	
응답사이즈	항목	측정값											
100 KB	Throughput	11 Gbps											
	CPU Usage	63 %											
	MEM Usage	11 %											

대항목	1.8 코어 인스턴스	시험 형태	10 GET / 1 Connection / 응답 사이즈 100 KB
중항목	1.3. Throughput	비고	SSL 복호화 미러링 X

상세 결과

○ 계측 클라이언트 처리 결과 화면

[클라이언트01]

```

✓ 200 OK
checks.....: 100.00% ✓ 825840 x 0
data_received.....: 100 GB 589 MB/s
data_sent.....: 130 MB 720 kB/s
http_req_blocked.....: avg=400.63µs min=743ns med=1.7µs max=49.1ms
http_req_connecting.....: avg=39.78µs min=0s med=0s max=20.55ms
http_req_duration.....: avg=32.22ms min=3.25ms med=30.42ms max=284.4ms
{ expected_response:true }.....: avg=32.22ms min=3.25ms med=30.42ms max=284.4ms
http_req_failed.....: 0.00% ✓ 0 x 825840
http_req_receiving.....: avg=6.89ms min=99.07µs med=6.18ms max=73.08ms
http_req_sending.....: avg=28.03µs min=4.26µs med=8.41µs max=23.69ms
http_req_tls_handshaking.....: avg=353.82µs min=0s med=0s max=48.85ms
http_req_waiting.....: avg=25.29ms min=2.37ms med=22.97ms max=275.7ms
http_reqs.....: 825840 4583.707112/s
iteration_duration.....: avg=327.07ms min=159.59ms med=318ms max=651.5ms
iterations.....: 82584 458.370711/s
script_errors.....: 0.00% ✓ 0 x 82584
vus.....: 150 min=150 max=150
vus_max.....: 150 min=150 max=150

running (3m00.2s), 000/150 VUs, 82584 complete and 0 interrupted iterations
default / [=====] 150 VUs 3m0s
root@client-01 sh#

```

[클라이언트02]

```

1 root@client-01:~$ 2 root@client-02:~$ 3 root@client-03:~$ 4 KS-ext
{ expected_response:true }.....: avg=5.18ms min=592.38µs med=3.79ms max=73.21ms p(90)=8.29ms p(95)=14.02ms
http_req_failed.....: 0.00% ✓ 0 x 4316560
http_req_receiving.....: avg=25.29µs min=10.16µs med=19.19µs max=53.21ms p(90)=30.81µs p(95)=37.25µs
http_req_sending.....: avg=21.88µs min=3.09µs med=5.75µs max=30.98ms p(90)=16.88µs p(95)=24.17µs
http_req_tls_handshaking.....: avg=5.14ms min=0s med=0s max=130.42ms p(90)=88.88µs p(95)=53.34ms
http_req_waiting.....: avg=5.13ms min=563.9µs med=3.75ms max=73.14ms p(90)=8.17ms p(95)=13.93ms
http_reqs.....: 4316560 23973.78747/s
iteration_duration.....: avg=104.23ms min=38.73ms med=98.53ms max=279.27ms p(90)=135.01ms p(95)=153.62ms
iterations.....: 431656 2397.378747/s
script_errors.....: 0.00% ✓ 0 x 431656
vus.....: 250 min=250 max=250
vus_max.....: 250 min=250 max=250

running (3m00.1s), 000/250 VUs, 431656 complete and 0 interrupted iterations
default / [=====] 250 VUs 3m0s
root@client-02 sh# sh throw.sh

execution: local
script: /root/script/throughput/throw.sh
output: -

scenarios: (100.00%) 1 scenario, 150 max VUs, 3m30s max duration (incl. graceful stop):
* default: 150 looping VUs for 3m0s (gracefulStop: 30s)

✓ 200 OK
checks.....: 100.00% ✓ 570050 x 0
data_received.....: 73 GB 407 MB/s
data_sent.....: 90 MB 457 kB/s
http_req_blocked.....: avg=623.19µs min=801ns med=1.8µs max=46.13ms p(90)=1.21ms p(95)=5.87ms
http_req_connecting.....: avg=71.23µs min=0s med=0s max=16.75ms p(90)=14.26µs p(95)=25.13µs
http_req_duration.....: avg=46.69ms min=2.67ms med=46.1ms max=217.37ms p(90)=61.6ms p(95)=67.94ms
{ expected_response:true }.....: avg=46.69ms min=2.67ms med=46.1ms max=217.37ms p(90)=61.6ms p(95)=67.94ms
http_req_failed.....: 0.00% ✓ 0 x 570050
http_req_receiving.....: avg=18ms min=110.98µs med=21ms max=68.64ms p(90)=26.48ms p(95)=28.33ms
http_req_sending.....: avg=22.16µs min=68µs med=9.02µs max=21.54ms p(90)=24.83µs p(95)=48.47µs
http_req_tls_handshaking.....: avg=544.47µs min=0s med=0s max=41.47ms p(90)=92.35µs p(95)=51.55ms
http_req_waiting.....: avg=28.66ms min=1.8ms med=26.02ms max=183.5ms p(90)=43.75ms p(95)=51.05ms
http_reqs.....: 570050 3162.936712/s

running (3m00.2s), 000/150 VUs, 57005 complete and 0 interrupted iterations
default / [=====] 150 VUs 3m0s

```

[클라이언트03]

```

1 root@client-01:~$ 2 root@client-02:~$ 3 root@client-03:~$ 4 KS-ext
{ expected_response:true }.....: avg=5.14ms min=512.96µs med=3.77ms max=82.86ms p(90)=8.21ms p(95)=13.96ms
http_req_failed.....: 0.00% ✓ 0 x 4330360
http_req_receiving.....: avg=24.23µs min=9.66µs med=18.32µs max=50.98ms p(90)=29.77µs p(95)=36.12µs
http_req_sending.....: avg=20.06µs min=2.99µs med=5.46µs max=51.28ms p(90)=15.84µs p(95)=22.88µs
http_req_tls_handshaking.....: avg=5.14ms min=0s med=0s max=136.58ms p(90)=89.78µs p(95)=53.34ms
http_req_waiting.....: avg=5.1ms min=482.81µs med=3.73ms max=82.84ms p(90)=8.1ms p(95)=13.89ms
http_reqs.....: 4330360 24046.625123/s
iteration_duration.....: avg=103.89ms min=44.75ms med=98.29ms max=294.27ms p(90)=134.26ms p(95)=152.91ms
iterations.....: 433036 2404.682512/s
script_errors.....: 0.00% ✓ 0 x 433036
vus.....: 250 min=250 max=250
vus_max.....: 250 min=250 max=250

running (3m00.1s), 000/250 VUs, 433036 complete and 0 interrupted iterations
default / [=====] 250 VUs 3m0s
root@client-03 sh# sh throw.sh

execution: local
script: /root/script/throughput/throw.sh
output: -

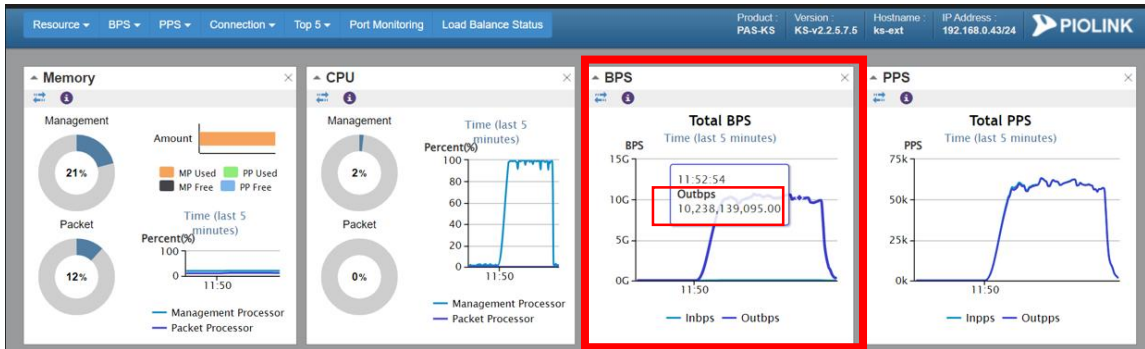
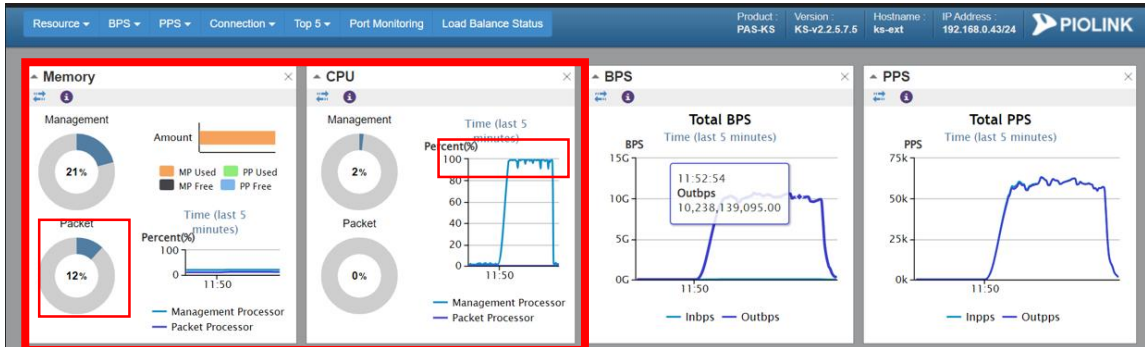
scenarios: (100.00%) 1 scenario, 150 max VUs, 3m30s max duration (incl. graceful stop):
* default: 150 looping VUs for 3m0s (gracefulStop: 30s)

✓ 200 OK
checks.....: 100.00% ✓ 567370 x 0
data_received.....: 73 GB 405 MB/s
data_sent.....: 80 MB 495 kB/s
http_req_blocked.....: avg=627.44µs min=775ns med=1.83µs max=45.35ms p(90)=1.23ms p(95)=5.87ms
http_req_connecting.....: avg=72.61µs min=0s med=0s max=17.83ms p(90)=13.74µs p(95)=24.97µs
http_req_duration.....: avg=46.9ms min=3.88ms med=46.3ms max=200.33ms p(90)=61.72ms p(95)=68.05ms
{ expected_response:true }.....: avg=46.9ms min=3.88ms med=46.3ms max=200.33ms p(90)=61.72ms p(95)=68.05ms
http_req_failed.....: 0.00% ✓ 0 x 567370
http_req_receiving.....: avg=18.09ms min=108.97µs med=21.18ms max=67.71ms p(90)=26.58ms p(95)=28.51ms
http_req_sending.....: avg=22.22µs min=6.4µs med=9.01µs max=21.17ms p(90)=24.8µs p(95)=48.58µs
http_req_tls_handshaking.....: avg=547.21µs min=0s med=0s max=42.79ms p(90)=92.92µs p(95)=51.53ms
http_req_waiting.....: avg=28.78ms min=2.17ms med=26.13ms max=183.5ms p(90)=43.98ms p(95)=51.15ms
http_reqs.....: 567370 3148.01666/s

running (3m00.2s), 000/150 VUs, 56737 complete and 0 interrupted iterations
default / [=====] 150 VUs 3m0s

```

200 OK 를 100% 수신 → PAS-KS 를 통한 요청에 문제가 없음을 의미

대항목	1. 8 코어 인스턴스	시험 형태	10 GET / 1 Connection / 응답 사이즈 100 KB										
중항목	1.4. Throughput	비고	SSL 복호화 미러링 O										
시험 결과		상세 결과											
<table><tr><th>응답사이즈</th><th>항목</th><th>측정값</th></tr><tr><td rowspan="3">100 KB</td><td>Throughput</td><td>10 Gbps</td></tr><tr><td>CPU Usage</td><td>100 %</td></tr><tr><td>MEM Usage</td><td>12 %</td></tr></table>		응답사이즈	항목	측정값	100 KB	Throughput	10 Gbps	CPU Usage	100 %	MEM Usage	12 %	<p>○ PAS-KS 에서 확인한 Throughput 처리 현황</p>  <p>○ PAS-KS 에서 확인한 CPU / MEM 사용량</p> 	
응답사이즈	항목	측정값											
100 KB	Throughput	10 Gbps											
	CPU Usage	100 %											
	MEM Usage	12 %											

대항목	1.8 코어 인스턴스	시험 형태	10 GET / 1 Connection / 응답 사이즈 100 KB
중항목	1.4. Throughput	비고	SSL 복호화 미러링 O

상세 결과

○ 계측 클라이언트 처리 결과 화면

[클라이언트01]

```

✓ 200 OK
checks.....: 100.00% ✓ 825840 x 0
data_received.....: 100 GB 589 MB/s
data_sent.....: 130 MB 720 kB/s
http_req_blocked.....: avg=400.63µs min=743ns med=1.7µs max=49.1ms
http_req_connecting.....: avg=39.78µs min=0s med=0s max=20.55ms
http_req_duration.....: avg=32.22ms min=3.25ms med=30.42ms max=284.4ms
{ expected_response:true }.....: avg=32.22ms min=3.25ms med=30.42ms max=284.4ms
http_req_failed.....: 0.00% ✓ 0 x 825840
http_req_receiving.....: avg=6.89ms min=99.07µs med=6.18ms max=73.08ms
http_req_sending.....: avg=28.03µs min=4.26µs med=8.41µs max=23.69ms
http_req_tls_handshaking.....: avg=353.82µs min=0s med=0s max=48.85ms
http_req_waiting.....: avg=25.29ms min=2.37ms med=22.97ms max=275.7ms
http_reqs.....: 825840 4583.707112/s
iteration_duration.....: avg=327.07ms min=159.59ms med=318ms max=651.5ms
iterations.....: 82584 458.370711/s
script_errors.....: 0.00% ✓ 0 x 82584
vus.....: 150 min=150 max=150
vus_max.....: 150 min=150 max=150

running (3m00.2s), 000/150 VUs, 82584 complete and 0 interrupted iterations
default / [=====] 150 VUs 3m0s
root@client-01 sh#

```

[클라이언트02]

```

1 root@client-01:~$ 2 root@client-02:~$ 3 root@client-03:~$ 4 KS-ext
{ expected_response:true }.....: avg=5.18ms min=592.38µs med=3.79ms max=73.21ms p(90)=8.29ms p(95)=14.02ms
http_req_failed.....: 0.00% ✓ 0 x 4316560
http_req_receiving.....: avg=25.29µs min=10.16µs med=19.19µs max=53.21ms p(90)=30.81µs p(95)=37.25µs
http_req_sending.....: avg=21.88µs min=3.09µs med=5.75µs max=30.98ms p(90)=16.88µs p(95)=24.17µs
http_req_tls_handshaking.....: avg=5.14ms min=0s med=0s max=130.42ms p(90)=88.88µs p(95)=53.34ms
http_req_waiting.....: avg=5.13ms min=563.9µs med=3.75ms max=73.14ms p(90)=8.17ms p(95)=13.93ms
http_reqs.....: 4316560 23973.78747/s
iteration_duration.....: avg=104.23ms min=38.73ms med=98.53ms max=279.27ms p(90)=135.01ms p(95)=153.62ms
iterations.....: 431656 2397.378747/s
script_errors.....: 0.00% ✓ 0 x 431656
vus.....: 250 min=250 max=250
vus_max.....: 250 min=250 max=250

running (3m00.1s), 000/250 VUs, 431656 complete and 0 interrupted iterations
default / [=====] 250 VUs 3m0s
root@client-02 sh# sh throw.sh

execution: local
script: /root/script/throughput/throw.sh
output: -

scenarios: (100.00%) 1 scenario, 150 max VUs, 3m30s max duration (incl. graceful stop):
* default: 150 looping VUs for 3m0s (gracefulStop: 30s)

✓ 200 OK
checks.....: 100.00% ✓ 570050 x 0
data_received.....: 73 GB 407 MB/s
data_sent.....: 90 MB 457 kB/s
http_req_blocked.....: avg=623.19µs min=801ns med=1.8µs max=46.13ms p(90)=1.21ms p(95)=5.87ms
http_req_connecting.....: avg=71.23µs min=0s med=0s max=16.75ms p(90)=14.26µs p(95)=25.13µs
http_req_duration.....: avg=46.69ms min=2.67ms med=46.1ms max=217.37ms p(90)=61.6ms p(95)=67.94ms
{ expected_response:true }.....: avg=46.69ms min=2.67ms med=46.1ms max=217.37ms p(90)=61.6ms p(95)=67.94ms
http_req_failed.....: 0.00% ✓ 0 x 570050
http_req_receiving.....: avg=18ms min=110.98µs med=21ms max=68.64ms p(90)=26.48ms p(95)=28.33ms
http_req_sending.....: avg=22.16µs min=68µs med=9.02µs max=21.54ms p(90)=24.83µs p(95)=48.47µs
http_req_tls_handshaking.....: avg=544.47µs min=0s med=0s max=41.47ms p(90)=92.35µs p(95)=51.55ms
http_req_waiting.....: avg=28.66ms min=1.1ms med=26.02ms max=183.5ms p(90)=43.75ms p(95)=51.05ms
http_reqs.....: 570050 3162.93671/s

running (3m00.2s), 000/150 VUs, 57005 complete and 0 interrupted iterations
default / [=====] 150 VUs 3m0s

```

[클라이언트03]

```

1 root@client-01:~$ 2 root@client-02:~$ 3 root@client-03:~$ 4 KS-ext
{ expected_response:true }.....: avg=5.14ms min=512.96µs med=3.77ms max=82.86ms p(90)=8.21ms p(95)=13.96ms
http_req_failed.....: 0.00% ✓ 0 x 4330360
http_req_receiving.....: avg=24.23µs min=9.66µs med=18.32µs max=50.98ms p(90)=29.77µs p(95)=36.12µs
http_req_sending.....: avg=20.06µs min=2.99µs med=5.46µs max=51.28ms p(90)=15.84µs p(95)=22.88µs
http_req_tls_handshaking.....: avg=5.14ms min=0s med=0s max=136.58ms p(90)=89.78µs p(95)=53.34ms
http_req_waiting.....: avg=5.1ms min=482.81µs med=3.73ms max=82.84ms p(90)=8.1ms p(95)=13.89ms
http_reqs.....: 4330360 24046.625123/s
iteration_duration.....: avg=103.89ms min=44.75ms med=98.29ms max=294.27ms p(90)=134.26ms p(95)=152.91ms
iterations.....: 433036 2404.682512/s
script_errors.....: 0.00% ✓ 0 x 433036
vus.....: 250 min=250 max=250
vus_max.....: 250 min=250 max=250

running (3m00.1s), 000/250 VUs, 433036 complete and 0 interrupted iterations
default / [=====] 250 VUs 3m0s
root@client-03 sh# sh throw.sh

execution: local
script: /root/script/throughput/throw.sh
output: -

scenarios: (100.00%) 1 scenario, 150 max VUs, 3m30s max duration (incl. graceful stop):
* default: 150 looping VUs for 3m0s (gracefulStop: 30s)

✓ 200 OK
checks.....: 100.00% ✓ 567370 x 0
data_received.....: 73 GB 405 MB/s
data_sent.....: 80 MB 495 kB/s
http_req_blocked.....: avg=627.44µs min=775ns med=1.83µs max=45.35ms p(90)=1.23ms p(95)=5.87ms
http_req_connecting.....: avg=72.61µs min=0s med=0s max=17.83ms p(90)=13.74µs p(95)=24.97µs
http_req_duration.....: avg=46.9ms min=3.88ms med=46.3ms max=200.33ms p(90)=61.72ms p(95)=68.05ms
{ expected_response:true }.....: avg=46.9ms min=3.88ms med=46.3ms max=200.33ms p(90)=61.72ms p(95)=68.05ms
http_req_failed.....: 0.00% ✓ 0 x 567370
http_req_receiving.....: avg=18.09ms min=108.97µs med=21.18ms max=67.71ms p(90)=26.58ms p(95)=28.51ms
http_req_sending.....: avg=22.22µs min=6.4µs med=9.01µs max=21.17ms p(90)=24.8µs p(95)=48.58µs
http_req_tls_handshaking.....: avg=547.21µs min=0s med=0s max=42.79ms p(90)=92.92µs p(95)=51.53ms
http_req_waiting.....: avg=28.78ms min=2.17ms med=26.13ms max=183.5ms p(90)=43.98ms p(95)=51.15ms
http_reqs.....: 567370 3148.01666/s

running (3m00.2s), 000/150 VUs, 56737 complete and 0 interrupted iterations
default / [=====] 150 VUs 3m0s

```

200 OK 를 100% 수신 → PAS-KS 를 통한 요청에 문제가 없음을 의미

대항목	2. 16 코어 인스턴스	시험 형태	1 GET / 1 Connection / 응답 사이즈 1 KB										
중항목	2.1. CPS	비고											
시험 결과		상세 결과											
<table><tr><th>응답사이즈</th><th>항목</th><th>측정값</th></tr><tr><td rowspan="3">1KB</td><td>CPS</td><td>24,807</td></tr><tr><td>CPU Usage</td><td>92.7 %</td></tr><tr><td>MEM Usage</td><td>12.33 %</td></tr></table>		응답사이즈	항목	측정값	1KB	CPS	24,807	CPU Usage	92.7 %	MEM Usage	12.33 %	<div>○ PAS-KS 에서 확인한 CPS 처리 현황</div> <div><pre>===== ADV77CSLB: c1 ----- RPS/Peak : 24807/52138 Requests : 24269060 Cps : 24807 Current sessions Client : 21 Server : 206</pre></div> <div>○ PAS-KS 에서 확인한 CPU / MEM 사용량</div> <div><pre>ks-ext(config)# sho resource ===== RESOURCE ----- CPU Usage : 92.71% Memory Management Total : 6144 MB Used : 1360 MB Free : 4783 MB Usage : 22.14% Packet Total : 26093 MB Used : 3217 MB Free : 22875 MB Usage : 12.33% Log Storage Total : 42845 MB Used : 5018 MB Free : 35651 MB Usage : 13%</pre></div>	
응답사이즈	항목	측정값											
1KB	CPS	24,807											
	CPU Usage	92.7 %											
	MEM Usage	12.33 %											

대항목	2. 16 코어 인스턴스	시험 형태	1 GET / 1 Connection / 응답 사이즈 1 KB
중항목	2.1. CPS	비고	

상세 결과

○ 계측 클라이언트 처리 결과 화면

[클라이언트01]

```

1 root@client-01:~/sh x 2 root@client-02:~/sh x 3 centos@client-02:~/sh
✓ 200 OK
checks.....: 100.00% ✓ 1502132 x 0
data_received.....: 3.2 GB 18 MB/s
data_sent.....: 777 MB 4.3 MB/s
http_req_blocked.....: avg=4.33ms min=1.04ms med=4.17ms
max=39.82ms p(90)=5.08ms p(95)=5.8ms
http_req_connecting.....: avg=332.45µs min=112.07µs med=232.97µs
max=21.65ms p(90)=584.86µs p(95)=803.34µs
http_req_duration.....: avg=2.78ms min=566.33µs med=2.45ms
max=69.49ms p(90)=3.56ms p(95)=4.61ms
{ expected_response:true }...: avg=2.78ms min=566.33µs med=2.45ms
max=69.49ms p(90)=3.56ms p(95)=4.61ms
http_req_failed.....: 0.00% ✓ 0 x 1502132
http_req_receiving.....: avg=170.44µs min=11.74µs med=55.62µs
max=21.72ms p(90)=401.2µs p(95)=631.36µs
http_req_sending.....: avg=152.44µs min=7.25µs med=30.01µs
max=21.62ms p(90)=379.36µs p(95)=622.78µs
http_req_tls_handshaking.....: avg=3.97ms min=852.41µs med=3.85ms
max=39.58ms p(90)=4.72ms p(95)=5.32ms
http_req_waiting.....: avg=2.46ms min=458.94µs med=2.22ms
max=69.44ms p(90)=3ms p(95)=3.59ms
http_reqs.....: 1502132 8345.04609/s
iteration_duration.....: avg=7.18ms min=2ms med=6.71ms
max=73.34ms p(90)=8.68ms p(95)=10.81ms
iterations.....: 1502132 8345.04609/s
script_errors.....: 0.00% ✓ 0 x 1502132
vus.....: 60 min=60 max=60
vus_max.....: 60 min=60 max=60

running (3m00.0s), 00/60 VUs, 1502132 complete and 0 interrupted iteration
default ✓ [=====] 60 VUs 3m0s

```

[클라이언트02]

```

1 root@client-01:~/sh x 2 root@client-02:~/sh x 3 centos@client-02:~/sh
script: /root/script/https/cps.js
output: -

scenarios: (100.00%) 1 scenario, 60 max VUs, 3m30s max duration (incl. graceful s
top):
* default: 60 looping VUs for 3m0s (gracefulStop: 30s)

✓ 200 OK
checks.....: 100.00% ✓ 1478892 x 0
data_received.....: 3.2 GB 18 MB/s
data_sent.....: 765 MB 4.2 MB/s
http_req_blocked.....: avg=4.41ms min=1.1ms med=4.23ms max=45.37
ms p(90)=5.17ms p(95)=5.91ms
http_req_connecting.....: avg=344.27µs min=119.69µs med=241.59µs max=19.58
ms p(90)=604.01µs p(95)=831.95µs
http_req_duration.....: avg=2.82ms min=584.34µs med=2.48ms max=65.89
ms p(90)=3.63ms p(95)=4.71ms
{ expected_response:true }...: avg=2.82ms min=584.34µs med=2.48ms max=65.89
ms p(90)=3.63ms p(95)=4.71ms
http_req_failed.....: 0.00% ✓ 0 x 1478892
http_req_receiving.....: avg=169.09µs min=12.11µs med=55.56µs max=22.93
ms p(90)=393.26µs p(95)=624.62µs
http_req_sending.....: avg=15.29µs min=7.5µs med=30.33µs max=21.03
ms p(90)=375.57µs p(95)=618.0µs
http_req_tls_handshaking.....: avg=4.02ms min=876.00µs med=3.9ms max=45.03
ms p(90)=4.79ms p(95)=5.39ms
http_req_waiting.....: avg=2.5ms min=452.96µs med=2.25ms max=65.52
ms p(90)=3.07ms p(95)=3.68ms
iterations.....: 1478892 8153.932339/s
script_errors.....: 0.00% ✓ 0 x 1478892
vus.....: 60 min=60 max=60
vus_max.....: 60 min=60 max=60

running (3m00.0s), 00/60 VUs, 1478892 complete and 0 interrupted iterations
default ✓ [=====] 60 VUs 3m0s

```

[클라이언트03]

```

1 root@client-01:~/sh x 2 root@client-02:~/sh x 3 root@client-03:~/sh x 4 KS-ext
✓ 200 OK
checks.....: 100.00% ✓ 1467767 x 0
data_received.....: 3.2 GB 18 MB/s
data_sent.....: 759 MB 4.2 MB/s
http_req_blocked.....: avg=4.43ms min=1.06ms med=4.25ms max=45.52ms
p(90)=5.21ms p(95)=5.97ms
http_req_connecting.....: avg=352.34µs min=118.3µs med=245.08µs max=21.71ms
p(90)=626.63µs p(95)=862.71µs
http_req_duration.....: avg=2.85ms min=596.13µs med=2.5ms max=70.53ms
p(90)=3.69ms p(95)=4.81ms
{ expected_response:true }...: avg=2.85ms min=596.13µs med=2.5ms max=70.53ms
p(90)=3.69ms p(95)=4.81ms
http_req_failed.....: 0.00% ✓ 0 x 1467767
http_req_receiving.....: avg=178.75µs min=13.16µs med=58.33µs max=23.47ms
p(90)=417.4µs p(95)=658.46µs
http_req_sending.....: avg=160.69µs min=8.44µs med=32.35µs max=22.87ms
p(90)=397.41µs p(95)=652.71µs
http_req_tls_handshaking.....: avg=4.04ms min=867.43µs med=3.91ms max=45.28ms
p(90)=4.81ms p(95)=5.44ms
http_req_waiting.....: avg=2.51ms min=487.75µs med=2.26ms max=70.34ms
p(90)=3.09ms p(95)=3.72ms
http_reqs.....: 1467767 8153.932339/s
iteration_duration.....: avg=7.35ms min=2.15ms med=6.85ms max=74.58ms
p(90)=11.17ms
iterations.....: 1467767 8153.932339/s
script_errors.....: 0.00% ✓ 0 x 1467767
vus.....: 60 min=60 max=60
vus_max.....: 60 min=60 max=60

running (3m00.0s), 00/60 VUs, 1467767 complete and 0 interrupted iterations
default ✓ [=====] 60 VUs 3m0s

```

200 OK 를 100% 수신 → PAS-KS 를 통한 요청에 문제가 없음을 의미

대항목	2. 16 코어 인스턴스	시험 형태	10 GET / 1 Connection / 응답 사이즈 1 KB										
중항목	2.2. TPS	비고											
시험 결과		상세 결과											
<table><tr><th>응답사이즈</th><th>항목</th><th>측정값</th></tr><tr><td rowspan="3">1KB</td><td>TPS</td><td>124,587</td></tr><tr><td>CPU Usage</td><td>92.9 %</td></tr><tr><td>MEM Usage</td><td>11.84 %</td></tr></table>		응답사이즈	항목	측정값	1KB	TPS	124,587	CPU Usage	92.9 %	MEM Usage	11.84 %	<div>○ PAS-KS 에서 확인한 TPS 처리 현황</div> <div><pre>===== ADVL7CSLB: c1 ----- RPS/Peak : 124587/129140 Requests : 48170056 Cps : 12432 Current sessions Client : 594 Server : 680 Total : 1274 =====</pre></div> <div>○ PAS-KS 에서 확인한 CPU / MEM 사용량</div> <div><pre>ks-ext(config)# sho resource ===== RESOURCE ----- CPU Usage : 92.99% Memory Management Total : 6144 MB Used : 1641 MB Free : 4502 MB Usage : 26.71% Packet Total : 26093 MB Used : 3089 MB Free : 23004 MB Usage : 11.84% Log Storage Total : 42845 MB Used : 5021 MB Free : 35647 MB Usage : 13% =====</pre></div>	
응답사이즈	항목	측정값											
1KB	TPS	124,587											
	CPU Usage	92.9 %											
	MEM Usage	11.84 %											

대항목	2. 16 코어 인스턴스	시험 형태	10 GET / 1 Connection / 응답 사이즈 1 KB
중항목	2.2. TPS	비고	

상세 결과

○ 계측 클라이언트 처리 결과 화면

[클라이언트01]

```

1 root@client-01:~/sh x 2 root@client-02:~/sh x 3 root@client-03:~/sh x
[root@client-01:~/sh]# sh rps.sh
[root@client-01:~/sh]# sh rps.sh

execution: local
script: /root/script/https/rps.js
output: -

scenarios: (100.00%) 1 scenario, 250 max VUs, 3m30s max duration (incl. graceful stop):
* default: 250 looping VUs for 3m0s (gracefulStop: 30s)

✓ 200 OK
checks.....: 100.00% ✓ 7900990 x 0
data_received.....: 1.1 GB 6.8 MB/s
data_sent.....: 1.2 GB 6.8 MB/s
http_req_blocked.....: avg=904.78µs min=563ns med=977ns max=104.68ms p(90)=1.15ms p(95)=7.61ms
http_req_connecting.....: avg=113.34µs min=0s med=0s max=73.54ms p(90)=1.197µs p(95)=526.39µs
http_req_duration.....: avg=4.71ms min=481.91µs med=3.94ms max=115.49ms p(90)=7.25ms p(95)=10.37ms
{ expected_response:true }....: avg=4.71ms min=481.91µs med=3.94ms max=115.49ms p(90)=7.25ms p(95)=10.37ms
http_req_failed.....: 0.00% ✓ 0 x 7900990
http_req_receiving.....: avg=27.07µs min=8.8µs med=15.79µs max=36.99µs p(90)=6.55µs p(95)=33.22µs
http_req_sending.....: avg=74.66µs min=2.88µs med=4.91µs max=98.65ms p(90)=1.83µs p(95)=183.23µs
http_req_tls_handshaking.....: avg=786.6µs min=0s med=0s max=84.13ms p(90)=8.51µs p(95)=0.33ms
http_req_waiting.....: avg=4.61ms min=455.92µs med=3.87ms max=102.32ms p(90)=7.06ms p(95)=10.09ms
http_reqs.....: 7900990 43884.438957/s
iteration_duration.....: avg=56.91ms min=20.73ms med=53.72ms max=198.01ms p(90)=7.615ms p(95)=85.44ms
iterations.....: 7900990 4388.443896/s
script_errors.....: 0.00% ✓ 0 x 7900990
vus.....: 250 min=250 max=250
vus_max.....: 250 min=250 max=250

running (3m00.0s), 000/250 VUs, 790099 complete and 0 interrupted iterations
default [=====] 250 VUs 3m0s

```

[클라이언트02]

```

1 root@client-01:~/sh x 2 root@client-02:~/sh x 3 root@client-03:~/sh x
[root@client-02:~/sh]# sh rps.sh
[root@client-02:~/sh]# sh rps.sh

execution: local
script: /root/script/https/rps.js
output: -

scenarios: (100.00%) 1 scenario, 250 max VUs, 3m30s max duration (incl. graceful stop):
* default: 250 looping VUs for 3m0s (gracefulStop: 30s)

✓ 200 OK
checks.....: 100.00% ✓ 6855160 x 0
data_received.....: 9.8 GB 5.4 MB/s
data_sent.....: 1.1 GB 5.9 MB/s
http_req_blocked.....: avg=904.78µs min=563ns med=977ns max=104.68ms p(90)=1.15ms p(95)=7.61ms
http_req_connecting.....: avg=113.34µs min=0s med=0s max=73.54ms p(90)=1.197µs p(95)=526.39µs
http_req_duration.....: avg=4.71ms min=481.91µs med=3.94ms max=115.49ms p(90)=7.25ms p(95)=10.37ms
{ expected_response:true }....: avg=4.71ms min=481.91µs med=3.94ms max=115.49ms p(90)=7.25ms p(95)=10.37ms
http_req_failed.....: 0.00% ✓ 0 x 6855160
http_req_receiving.....: avg=26.4µs min=9.07µs med=17.68µs max=42.93ms p(90)=7.2µs p(95)=7.74ms
http_req_sending.....: avg=53.35µs min=3.01µs med=5.43µs max=41.91ms p(90)=7.2µs p(95)=7.74ms
http_req_tls_handshaking.....: avg=900.76µs min=0s med=0s max=68.54ms p(90)=7.2µs p(95)=7.74ms
http_req_waiting.....: avg=5.41ms min=428.86µs med=4.59ms max=86.36ms p(90)=7.2µs p(95)=7.74ms
http_reqs.....: 6855160 38077.041105/s
iteration_duration.....: avg=65.6ms min=22.78ms med=62.62ms max=195.27ms p(90)=7.2µs p(95)=7.74ms
iterations.....: 685516 3807.704111/s
script_errors.....: 0.00% ✓ 0 x 685516
vus.....: 250 min=250 max=250
vus_max.....: 250 min=250 max=250

running (3m00.0s), 000/250 VUs, 685516 complete and 0 interrupted iterations
default [=====] 250 VUs 3m0s

```

[클라이언트03]

```

1 root@client-01:~/sh x 2 root@client-02:~/sh x 3 root@client-03:~/sh x
[root@client-03:~/sh]# sh rps.sh
[root@client-03:~/sh]# sh rps.sh

execution: local
script: /root/script/https/rps.js
output: -

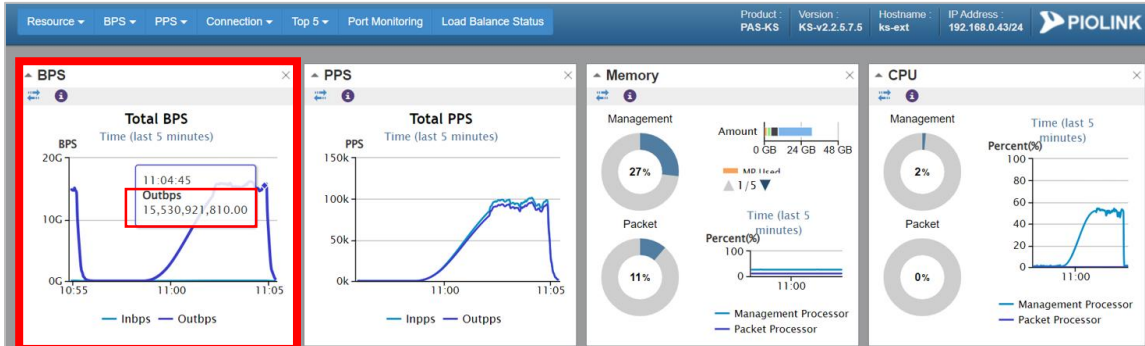
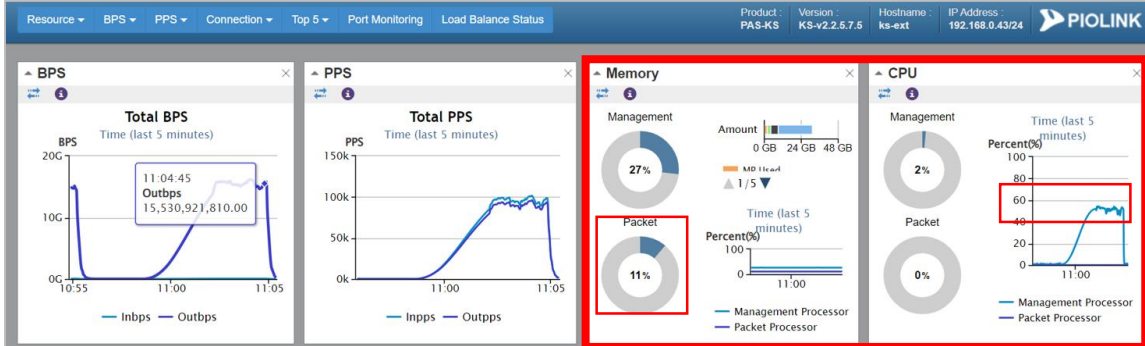
scenarios: (100.00%) 1 scenario, 250 max VUs, 3m30s max duration (incl. graceful stop):
* default: 250 looping VUs for 3m0s (gracefulStop: 30s)

✓ 200 OK
checks.....: 100.00% ✓ 6789120 x 0
data_received.....: 0.7 GB 5.4 MB/s
data_sent.....: 1.1 GB 5.8 MB/s
http_req_blocked.....: avg=1.01ms min=589ns med=992ns max=93.83ms p(90)=1.18ms p(95)=8.65ms
http_req_connecting.....: avg=101µs min=0s med=0s max=68.9ms p(90)=12.49µs p(95)=387µs
http_req_duration.....: avg=5.54ms min=515.87µs med=4.67ms max=91.18ms p(90)=8.35ms p(95)=11.54ms
{ expected_response:true }....: avg=5.54ms min=515.87µs med=4.67ms max=91.18ms p(90)=8.35ms p(95)=11.54ms
http_req_failed.....: 0.00% ✓ 0 x 6789120
http_req_receiving.....: avg=29.2µs min=9.73µs med=18.13µs max=63.66ms p(90)=29.8µs p(95)=36.99µs
http_req_sending.....: avg=57.2µs min=3.04µs med=5.5µs max=67.08ms p(90)=17.94µs p(95)=107.92µs
http_req_tls_handshaking.....: avg=908.2µs min=0s med=0s max=93.63ms p(90)=86.72µs p(95)=7.74ms
http_req_waiting.....: avg=5.45ms min=477.9µs med=4.61ms max=91.15ms p(90)=8.21ms p(95)=11.3ms
http_reqs.....: 6789120 37707.644404/s
iteration_duration.....: avg=66.24ms min=23.49ms med=63.01ms max=193.34ms p(90)=8.8ms p(95)=97.49ms
iterations.....: 678912 3770.76444/s
script_errors.....: 0.00% ✓ 0 x 678912
vus.....: 250 min=250 max=250
vus_max.....: 250 min=250 max=250

running (3m00.0s), 000/250 VUs, 678912 complete and 0 interrupted iterations
default [=====] 250 VUs 3m0s

```

200 OK 를 100% 수신 → PAS-KS 를 통한 요청에 문제가 없음을 의미

대항목	2. 16 코어 인스턴스	시험 형태	10 GET / 1 Connection / 응답 사이즈 100 KB										
중항목	2.3. Throughput	비고	SSL 복호화 미러링 X										
시험 결과		상세 결과											
<table><tr><th>응답사이즈</th><th>항목</th><th>측정값</th></tr><tr><td rowspan="3">100 KB</td><td>Throughput</td><td>15 Gbps</td></tr><tr><td>CPU Usage</td><td>57 %</td></tr><tr><td>MEM Usage</td><td>11 %</td></tr></table>		응답사이즈	항목	측정값	100 KB	Throughput	15 Gbps	CPU Usage	57 %	MEM Usage	11 %	<p>○ PAS-KS 에서 확인한 Throughput 처리 현황</p>  <p>○ PAS-KS 에서 확인한 CPU / MEM 사용량</p> 	
응답사이즈	항목	측정값											
100 KB	Throughput	15 Gbps											
	CPU Usage	57 %											
	MEM Usage	11 %											

대항목	2. 16 코어 인스턴스	시험 형태	10 GET / 1 Connection / 응답 사이즈 100 KB
중항목	2.3. Throughput	비고	SSL 복호화 미러링 X

상세 결과

○ 계측 클라이언트 처리 결과 화면

[클라이언트01]

```

1 root@client-01:~#
2 root@client-02:~#
3 root@client-03:~#
4 KS-ext
+

4ms max=425.32ms p(90)=12.26ms p(95)=14.6ms
http_reqs..... 1050360 5830.57259/s
iteration_duration..... avg=257.13ms min=133.79ms med=255.22ms max=644.57ms p(90)=298.02ms p(95)=310.63ms
iterations..... 105036 583.05753/s
script_errors..... 0.00% ✓ 0 x 105036
vus..... 150 min=150 max=150
vus_max..... 150 min=150 max=150

running (3m00.1s), 000/150 VUs, 105036 complete and 0 interrupted iterations
default / [=====] 150 VUs 3m0s
[root@client-01 sh]# sh throw.sh

AKG.io

execution: local
script: /root/script/throughput/throw.js
output: -

scenarios: (100.00%) 1 scenario, 150 max VUs, 3m30s max duration (incl. graceful stop):
* default: 150 looping VUs for 3m0s (gracefulStop: 30s)

✓ 200 OK
checks..... 100.00% ✓ 1084970 x 0
data_sent..... 170 MB 946 kB/s
http_req_blocked..... avg=817.63µs min=763ms med=1.71µs max=44.03ms p(90)=1.39ms p(95)=7.72ms
http_req_connecting..... avg=74.17µs min=0s med=0s max=28.04ms p(90)=14.46µs p(95)=274.99µs
http_req_duration..... avg=23.59ms min=2.39ms med=23.22ms max=3.03s p(90)=30.84ms p(95)=34.89ms
{ expected_response:true }..... avg=23.59ms min=2.39ms med=23.22ms max=3.03s p(90)=30.84ms p(95)=34.89ms
http_req_failed..... 0.00% ✓ 0 x 1084970
http_req_receiving..... avg=16.3ms min=96.38µs med=16.24ms max=60.13ms p(90)=21.63ms p(95)=23.34ms
http_req_sending..... avg=51.85µs min=4.25µs med=8.5µs max=26.39ms p(90)=62.34µs p(95)=114.99µs
http_req_tls_handshaking..... avg=734.82µs min=0s med=0s max=43.51ms p(90)=96.74µs p(95)=7.13ms
http_req_waiting..... avg=7.63ms min=1.42ms med=6.43ms max=3s p(90)=12.38ms p(95)=14.74ms
http_reqs..... 1084970 6022.309142/s
iteration_duration..... avg=248.94ms min=137.97ms med=246.44ms max=3.27s p(90)=287.85ms
iterations..... 108497 602.230914/s
script_errors..... 0.00% ✓ 0 x 108497
vus..... 150 min=150 max=150
vus_max..... 150 min=150 max=150

running (3m00.2s), 000/150 VUs, 108497 complete and 0 interrupted iterations
default / [=====] 150 VUs 3m0s

```

[클라이언트02]

```

1 root@client-01:~#
2 root@client-02:~#
3 root@client-03:~#
4 KS-ext
+

max=24.49ms p(90)=45.28µs p(95)=114.72µs
http_req_tls_handshaking..... avg=991.46µs min=0s med=0s
max=37.95ms p(90)=103.73µs p(95)=9.66ms
http_req_waiting..... avg=9.09ms min=1.72ms med=7.74ms
max=439.02ms p(90)=15.24ms p(95)=17.95ms
http_reqs..... 793750 4405.78044/s
iteration_duration..... avg=340.29ms min=163.12ms med=333.85ms
max=870.63ms p(90)=392.52ms p(95)=405.13ms
iterations..... 79375 440.578044/s
script_errors..... 0.00% ✓ 0 x 79375
vus..... 150 min=150 max=150
vus_max..... 150 min=150 max=150

running (3m00.2s), 000/150 VUs, 79375 complete and 0 interrupted iterations
default / [=====] 150 VUs 3m0s
[root@client-02 sh]# sh throw.sh

AKG.io

execution: local
script: /root/script/throughput/throw.js
output: -

scenarios: (100.00%) 1 scenario, 150 max VUs, 3m30s max duration (incl. graceful stop):
* default: 150 looping VUs for 3m0s (gracefulStop: 30s)

✓ 200 OK
checks..... 100.00% ✓ 822470 x 0
data_sent..... 128 MB 717 kB/s
http_req_blocked..... avg=0.02ms min=0s med=0s max=1.83µs p(90)=42.6ms p(95)=1.45ms p(95)=274.99µs
http_req_connecting..... avg=74.09µs min=0s med=0s max=23.34ms p(90)=15.02µs p(95)=274.99µs
http_req_duration..... avg=31.92ms min=3.18ms med=30.22ms max=243.15ms p(90)=40.23ms p(95)=40.23ms
{ expected_response:true }..... avg=31.92ms min=3.18ms med=30.22ms max=243.15ms p(90)=40.23ms p(95)=40.23ms
http_req_failed..... 0.00% ✓ 0 x 822470
http_req_receiving..... avg=22.55ms min=122.16µs med=22.06ms max=77.15ms p(90)=28.69ms p(95)=39.47ms
http_req_sending..... avg=42.26µs min=4.57µs med=9.02µs max=23.43ms p(90)=47.23µs p(95)=97.7µs
http_req_tls_handshaking..... avg=9.13ms min=0s med=0s max=38.55ms p(90)=97.7µs p(95)=15.2ms
http_req_waiting..... avg=9.13ms min=0s med=0s max=38.55ms p(90)=97.7µs p(95)=15.2ms
http_reqs..... 822470 4405.78044/s
iteration_duration..... avg=340.29ms min=163.12ms med=333.85ms
iterations..... 82247 440.578044/s
script_errors..... 0.00% ✓ 0 x 82247
vus..... 150 min=150 max=150
vus_max..... 150 min=150 max=150

running (3m00.2s), 000/150 VUs, 82247 complete and 0 interrupted iterations
default / [=====] 150 VUs 3m0s

```

[클라이언트03]

```

1 root@client-01:~#
2 root@client-02:~#
3 root@client-03:~#
4 KS-ext
+

running (3m00.2s), 000/150 VUs, 78973 complete and 0 interrupted iterations
default / [=====] 150 VUs 3m0s
[root@client-03 sh]# sh throw.sh

AKG.io

execution: local
script: /root/script/throughput/throw.js
output: -

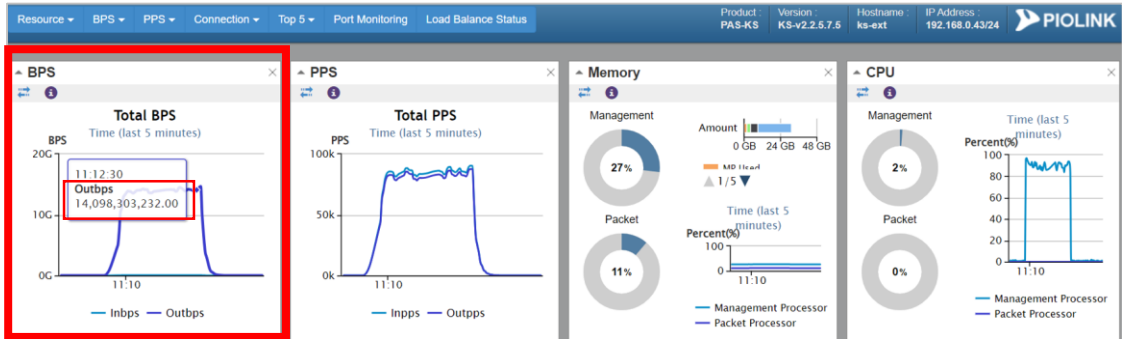
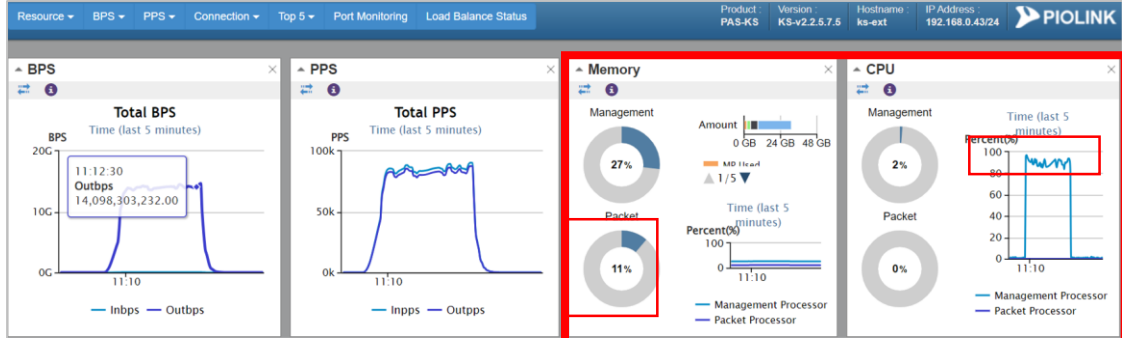
scenarios: (100.00%) 1 scenario, 150 max VUs, 3m30s max duration (incl. graceful stop):
* default: 150 looping VUs for 3m0s (gracefulStop: 30s)

running (1m01.4s), 150/150 VUs, 28251 complete and 0 interrupted iterations
✓ 200 OK
checks..... 100.00% ✓ 828920 x 0
data_received..... 107 MB 392 kB/s
data_sent..... 130 MB 723 kB/s
http_req_blocked..... avg=1.01ms min=763ms med=1.83µs
max=39.38ms p(90)=1.45ms p(95)=4.61ms
http_req_connecting..... avg=70.71µs min=0s med=0s
max=22.49ms p(90)=14.99µs p(95)=24.22µs
http_req_duration..... avg=31.48ms min=3.5ms med=29.84ms
max=261.61ms p(90)=39.98ms p(95)=4.33ms
{ expected_response:true }..... avg=31.48ms min=3.5ms med=29.84ms
max=261.61ms p(90)=39.98ms p(95)=4.33ms
http_req_failed..... 0.00% ✓ 0 x 828920
http_req_receiving..... avg=22.36ms min=308.91µs med=21.85ms
max=71.15ms p(90)=20.45ms p(95)=0.44ms
http_req_sending..... avg=42.56µs min=4.68µs med=8.99µs
max=26.16ms p(90)=49.99µs p(95)=17µs
http_req_tls_handshaking..... avg=936.94µs min=0s med=0s
max=39.47ms p(90)=104.78µs p(95)=11ms
http_req_waiting..... avg=9.07ms min=1.94ms med=7.63ms
max=248.78ms p(90)=15.13ms p(95)=7.79ms
http_reqs..... 828920 4601.146842/s
iteration_duration..... avg=325.04ms min=152.92ms med=319.82ms
iterations..... 82892 460.114684/s
script_errors..... 0.00% ✓ 0 x 82892
vus..... 150 min=150 max=150
vus_max..... 150 min=150 max=150

running (3m00.2s), 000/150 VUs, 82892 complete and 0 interrupted iterations
default / [=====] 150 VUs 3m0s

```

200 OK 를 100% 수신 → PAS-KS 를 통한 요청에 문제가 없음을 의미

대항목	2. 16 코어 인스턴스	시험 형태	10 GET / 1 Connection / 응답 사이즈 100 KB										
중항목	2.3. Throughput	비고	SSL 복호화 미러링 O										
시험 결과		상세 결과											
<table><tr><th>응답사이즈</th><th>항목</th><th>측정값</th></tr><tr><td rowspan="3">100 KB</td><td>Throughput</td><td>14 Gbps</td></tr><tr><td>CPU Usage</td><td>95 %</td></tr><tr><td>MEM Usage</td><td>11 %</td></tr></table>		응답사이즈	항목	측정값	100 KB	Throughput	14 Gbps	CPU Usage	95 %	MEM Usage	11 %	<p>○ PAS-KS 에서 확인한 Throughput 처리 현황</p>  <p>○ PAS-KS 에서 확인한 CPU / MEM 사용량</p> 	
응답사이즈	항목	측정값											
100 KB	Throughput	14 Gbps											
	CPU Usage	95 %											
	MEM Usage	11 %											

대항목	2. 16 코어 인스턴스 (8 코어 / 16 GB 메모리)	시험 형태	10 GET / 1 Connection / 응답 사이즈 100 KB
중항목	2.3. Throughput	비고	SSL 복호화 미러링 O

상세 결과

○ 계측 클라이언트 처리 결과 화면

[클라이언트01]

```

1 root@client-01:~/sh 2 root@client-02:~/sh 3 root@client-03:~/sh 4 KS-ext +
4ms max=425.32ms p(90)=12.26ms p(95)=14.6ms
http_reqs.....: 1050360 5830.575299/s
iteration_duration.....: avg=257.13ms min=133.79ms med=255.22ms
max=644.57ms p(90)=298.02ms p(95)=310.63ms
iterations.....: 105036 583.05753/s
script_errors.....: 0.00% / 0 x 105036
vus.....: 150 min=150 max=150
vus_max.....: 150 min=150 max=150

running (3m00.1s), 000/150 VUs, 105036 complete and 0 interrupted iterations
default / [=====] 150 VUs 3m0s
[root@client-01 sh]# sh throw.sh

MKG.io

execution: local
script: /root/script/throughput/throw.js
output: -

scenarios: (100.00%) 1 scenario, 150 max VUs, 3m30s max duration (incl. graceful stop):
* default: 150 looping VUs for 3m0s (gracefulStop: 30s)

/ 200 OK
checks.....: 100.00% / 1084970 x 0
data_received.....: 140 GB 774 MB/s
data_sent.....: 170 MB 946 KB/s
http_req_blocked.....: avg=817.63us min=763ms med=1.71us max=44.03ms p(90)=1.39ms p(95)=1.46ms
http_req_connecting.....: avg=74.17us min=0s med=0s max=28.04ms p(90)=14.46us p(95)=14.46us
http_req_duration.....: avg=23.99ms min=2.36ms med=23.25ms max=3.03s p(90)=30.84ms p(95)=30.84ms
http_req_failed.....: 0.00% / 0 x 1084970
http_req_receiving.....: avg=16.3ms min=96.38us med=16.24ms max=60.13ms p(90)=21.63ms p(95)=21.63ms
http_req_receiving.....: avg=51.85us min=4.25us med=8.5us max=26.36ms p(90)=62.34us p(95)=62.34us
http_req_tls_handshaking.....: avg=734.82us min=0s med=0s max=43.51ms p(90)=96.74us p(95)=96.74us
http_req_waiting.....: avg=7.63ms min=1.42ms med=6.43ms max=3s p(90)=12.38ms p(95)=12.38ms
iteration_duration.....: avg=248.94ms min=137.97ms med=246.44ms max=3.27s p(90)=287.65ms p(95)=287.65ms
iterations.....: 108497 602.230914/s
script_errors.....: 0.00% / 0 x 108497
vus.....: 150 min=150 max=150
vus_max.....: 150 min=150 max=150

running (3m00.2s), 000/150 VUs, 108497 complete and 0 interrupted iterations
default / [=====] 150 VUs 3m0s

```

[클라이언트02]

```

1 root@client-01:~/sh 2 root@client-02:~/sh 3 root@client-03:~/sh 4 KS-ext +
max=24.49ms p(90)=45.28us p(95)=114.72us
http_req_tls_handshaking.....: avg=991.46us min=0s med=0s
max=37.95ms p(90)=103.73us p(95)=9.66ms
http_req_waiting.....: avg=9.09ms min=1.72ms med=7.74ms
max=439.02ms p(90)=15.24ms p(95)=17.95ms
http_reqs.....: 793750 4405.78044/s
iteration_duration.....: avg=340.29ms min=163.12ms med=333.85ms
max=870.65ms p(90)=392.52ms p(95)=405.13ms
iterations.....: 79375 440.578044/s
script_errors.....: 0.00% / 0 x 79375
vus.....: 150 min=150 max=150
vus_max.....: 150 min=150 max=150

running (3m00.2s), 000/150 VUs, 79375 complete and 0 interrupted iterations
default / [=====] 150 VUs 3m0s
[root@client-02 sh]# sh throw.sh

MKG.io

execution: local
script: /root/script/throughput/throw.js
output: -

scenarios: (100.00%) 1 scenario, 150 max VUs, 3m30s max duration (incl. graceful stop):
* default: 150 looping VUs for 3m0s (gracefulStop: 30s)

/ 200 OK
checks.....: 100.00% / 822470 x 0
data_received.....: 106 GB 562 MB/s
data_sent.....: 124 MB 717 KB/s
http_req_blocked.....: avg=1.02ms min=813ms med=1.83us max=42.6ms p(90)=1.45ms p(95)=9.7
http_req_connecting.....: avg=7.89us min=0s med=0s max=23.34ms p(90)=15.02us p(95)=274
http_req_duration.....: avg=31.72ms min=3.18ms med=30.22ms max=243.15ms p(90)=40.23ms p(95)=444
http_req_failed.....: 0.00% / 0 x 822470
http_req_receiving.....: avg=22.55ms min=122.16us med=22.06ms max=77.15ms p(90)=28.69ms p(95)=30.
http_req_receiving.....: avg=42.26us min=4.57us med=9.02us max=23.43ms p(90)=47.23us p(95)=114
http_req_tls_handshaking.....: avg=87.68us min=0s med=0s max=38.55ms p(90)=97.7us p(95)=17.
http_req_waiting.....: avg=9.13ms min=1.96ms med=7.7ms max=219.05ms p(90)=15.2ms p(95)=17.
iteration_duration.....: avg=224.7ms min=122.16us med=22.06ms max=77.15ms p(90)=28.69ms p(95)=30.
iterations.....: 82247 4565.40891/s
script_errors.....: 0.00% / 0 x 82247
vus.....: 150 min=150 max=150
vus_max.....: 150 min=150 max=150

running (3m00.2s), 000/150 VUs, 82247 complete and 0 interrupted iterations
default / [=====] 150 VUs 3m0s

```

[클라이언트03]

```

1 root@client-01:~/sh 2 root@client-02:~/sh 3 root@client-03:~/sh 4 KS-ext +
running (3m00.2s), 000/150 VUs, 78973 complete and 0 interrupted iterations
default / [=====] 150 VUs 3m0s
[root@client-03 sh]# sh throw.sh

MKG.io

execution: local
script: /root/script/throughput/throw.js
output: -

scenarios: (100.00%) 1 scenario, 150 max VUs, 3m30s max duration (incl. graceful stop):
* default: 150 looping VUs for 3m0s (gracefulStop: 30s)

running (1m01.4s), 150/150 VUs, 28251 complete and 0 interrupted iterations
/ 200 OK
checks.....: 100.00% / 828920 x 0
data_received.....: 107 GB 502 MB/s
data_sent.....: 130 MB 723 KB/s
http_req_blocked.....: avg=1.01ms min=763ms med=1.83us max=44.03ms p(90)=14.46us p(95)=14.46us
http_req_connecting.....: avg=70.71us min=0s med=0s max=28.04ms p(90)=14.46us p(95)=14.46us
http_req_duration.....: avg=31.48ms min=3.5ms med=29.84ms max=261.61ms p(90)=39.98ms p(95)=44.33ms
http_req_failed.....: 0.00% / 0 x 828920
http_req_receiving.....: avg=22.36ms min=308.91us med=21.85ms max=71.13ms p(90)=28.45ms p(95)=30.44ms
http_req_receiving.....: avg=42.56us min=4.68us med=8.99us max=26.16ms p(90)=62.34us p(95)=62.34us
http_req_tls_handshaking.....: avg=936.94us min=0s med=0s max=39.07ms p(90)=104.78us p(95)=9.11ms
http_req_waiting.....: avg=9.07ms min=1.94ms med=7.63ms max=240.78ms p(90)=15.13ms p(95)=17.7ms
iteration_duration.....: avg=325.84ms min=152.92ms med=319.82ms max=3.27s p(90)=287.65ms p(95)=287.65ms
iterations.....: 82892 460.114684/s
script_errors.....: 0.00% / 0 x 82892
vus.....: 150 min=150 max=150
vus_max.....: 150 min=150 max=150

running (3m00.2s), 000/150 VUs, 82892 complete and 0 interrupted iterations
default / [=====] 150 VUs 3m0s

```

200 OK 를 100% 수신 → PAS-KS 를 통한 요청에 문제가 없음을 의미