

차세대 에이전트 성능 테스트 보고서

스탠다드네트웍스
전송개발팀

엠피코어 CPU 부하 이슈에 따른 성능 테스트 보고서

3월 14일 목요일, 엠피코어에서 서버에 설치된 13대의 Agent의 CPU 사용률이 높아지는 문제 분석

```
top - 13:50:34 up 280 days, 22:17, 2 users, load average: 18.94, 16.69, 12.43
Tasks: 176 total, 3 running, 94 sleeping, 0 stopped, 0 zombie
%Cpu(s): 95.8 us, 3.0 sy, 0.0 ni, 0.6 id, 0.0 wa, 0.0 hi, 0.6 si, 0.0 st
KiB Mem : 32499188 total, 9845120 free, 13661264 used, 8992804 buff/cache
KiB Swap: 0 total, 0 free, 0 used. 18417044 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
7006	ec2-user	20	0	6558472	1.0g	12624	S	1.0	3.3	295:13.60	java11
571	ec2-user	20	0	6597136	1.0g	13220	S	233.6	3.3	520:41.20	java11
17937	ec2-user	20	0	6552912	1.0g	13360	S	203.3	3.3	542:03.98	java11
25285	ec2-user	20	0	6565296	1.0g	12796	S	39.9	3.2	456:43.65	java11
26829	ec2-user	20	0	6596244	1.0g	13276	S	240.5	3.2	548:38.89	java11
15180	ec2-user	20	0	6552840	1.0g	10048	S	1.3	3.2	185:33.69	java11
30578	ec2-user	20	0	6561300	1.0g	13728	S	1.3	3.2	337:02.92	java11
19859	ec2-user	20	0	6584572	1.0g	13096	S	1.0	3.2	489:05.55	java11
14145	ec2-user	20	0	6563892	999.0m	11824	S	1.3	3.1	314:13.28	java11
14099	ec2-user	20	0	6572852	988.6m	12716	S	1.3	3.1	301:24.06	java11
22055	ec2-user	20	0	6550760	979.5m	12596	S	1.0	3.1	330:33.43	java11
16897	ec2-user	20	0	6561696	986716	12868	S	1.3	3.0	331:00.12	java11
8300	ec2-user	20	0	6560000	971660	12716	S	1.3	3.0	311:18.52	java11

사용 중인 Agent 상세 정보 확인 결과

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
26889	ec2-user	20	0	6593880	1.0g	13340	R	58.1	3.3	20:13.27	GC Thread#5
26832	ec2-user	20	0	6593880	1.0g	13340	R	57.8	3.3	20:17.85	GC Thread#0
26890	ec2-user	20	0	6593880	1.0g	13340	R	57.1	3.3	20:16.32	GC Thread#6
26888	ec2-user	20	0	6593880	1.0g	13340	R	55.8	3.3	20:17.05	GC Thread#4
26886	ec2-user	20	0	6593880	1.0g	13340	R	55.5	3.3	20:13.73	GC Thread#2
26885	ec2-user	20	0	6593880	1.0g	13340	R	51.2	3.3	20:16.62	GC Thread#1
26887	ec2-user	20	0	6593880	1.0g	13340	R	50.8	3.3	20:16.79	GC Thread#3
26891	ec2-user	20	0	6593880	1.0g	13340	R	48.2	3.3	20:17.74	GC Thread#7

1. 엠피코어 CPU 부하 문제 분석 내용

- 엠피코어 측 서버 환경(CPU 8Core) 에서 에이전트를 13대 설치하여 사용 중, CPU 사용량 증가 및 성능 저하 문제 발생
- 프로세스 상세 정보 확인 중, 특정 프로세스가 서버의 CPU 사용량의 많은 부분을 차지하는 것을 확인

2. 엠피코어 문제 재연 테스트

- 엠피코어 측의 서버와 성능이 비슷한 스탠다드네트웍스 개발 서버(CPU 8코어, 클럭수 2.1 Ghz, 메모리 8 GB) 내에서 13대의 에이전트를 구축, 13대의 에이전트로 총합 10만 건의 메시지를 분산 발송한다.
- 메시지를 전송한 모든 에이전트에서 메시지 전송 추이 및 CPU, 메모리 사용량 추이를 분석하여 CPU 사용량 폭증, 성능 저하 문제가 발생하는지 확인한다.
- 동일한 문제가 발생하는 경우, CPU 사용량 및 메모리 사용량과 모든 에이전트의 메시지 전송 속도 추이를 분석하여 해당 문제를 분석한다.

문제 재연 테스트 순서에 맞춰 테스트를 진행한 결과, 차세대 에이전트의 전송 추이는 위의 그래프와 같습니다.

• 재연 테스트 중, Agent 상세 정보 확인 결과

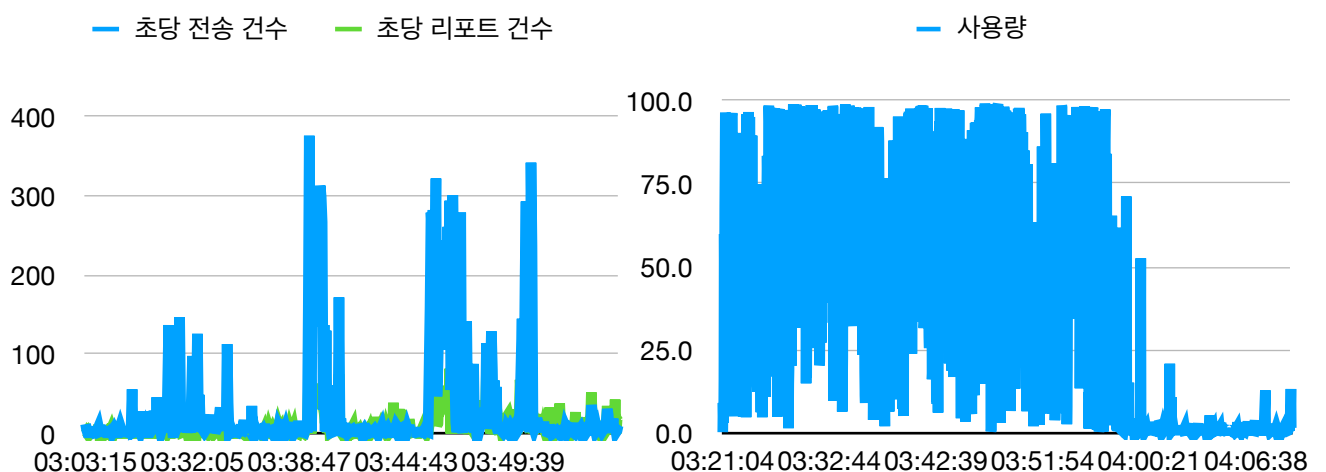
```
top - 03:02:22 up 347 days, 6:15, 16 users, load average: 101.01, 39.89, 15.01
Tasks: 86 total, 6 running, 80 sleeping, 0 stopped, 0 zombie
Cpu(s): 96.9%us, 2.0%sy, 0.0%ni, 0.1%id, 0.9%wa, 0.1%hi, 0.1%si, 0.0%st
Mem: 8043568k total, 7936376k used, 107192k free, 1844k buffers
Swap: 8175612k total, 6257444k used, 1918168k free, 56636k cached

  PID USER      PR  NI  VIRT  RES  SHR  S  %CPU  %MEM    TIME+  COMMAND
 9341 smsmt    20   0 6434m 381m 8656 R  65.8   4.9   0:12.47 GC Thread#7
 9339 smsmt    20   0 6434m 381m 8656 R  62.0   4.9   0:12.31 GC Thread#5
 9305 smsmt    20   0 6434m 381m 8656 R  61.0   4.9   0:10.08 GC Thread#0
 9337 smsmt    20   0 6434m 381m 8656 R  59.1   4.9   0:12.35 GC Thread#3
 9338 smsmt    20   0 6434m 381m 8656 R  58.7   4.9   0:10.78 GC Thread#4
 9335 smsmt    20   0 6434m 381m 8656 R  51.0   4.9   0:12.31 GC Thread#1
 9336 smsmt    20   0 6434m 381m 8656 S  48.2   4.9   0:10.74 GC Thread#2
```

재연 테스트 중, 스탠다드네트웍스 개발 서버에서도 앱코어에서와 동일한 문제가 발생하였습니다.

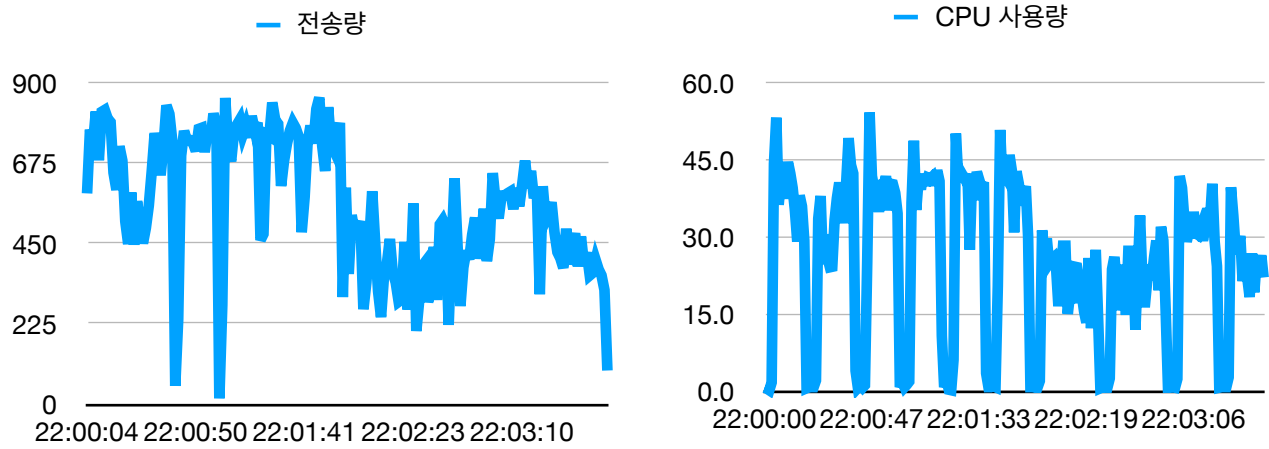
문제가 발견된 서버의 Agent들의 로그 분석 결과 전체적으로 Agent들이 성능 저하 문제가 발견 되었고, 특정 Agent 동작시 일부 Agent가 간헐적으로 멈추는 현상이 발견되었습니다.

• 앱코어와 동일한 환경에서의 차세대 에이전트 13대 전송 테스트



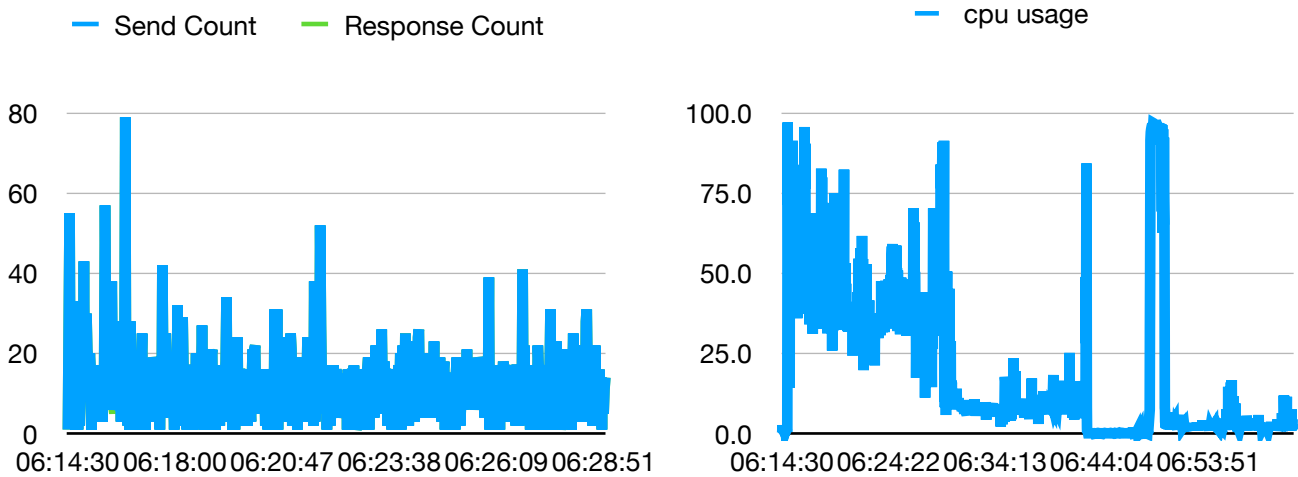
위의 그래프는 13개 에이전트 중 2개의 에이전트의 전송량 및 리포트량을 분석한 그래프입니다. 특정 Agent가 동작 시 다른 Agent의 처리 속도가 03시 30~40분경 현저히 저하 되었습니다.

• 앱코어와 동일한 환경에서의 차세대 에이전트 1대 전송 테스트



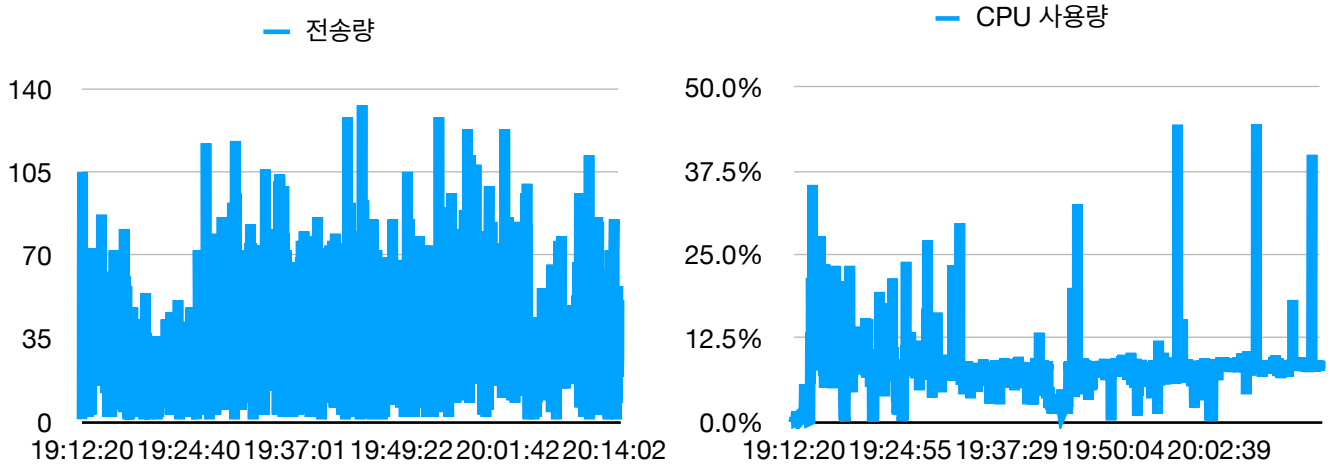
차세대 에이전트의 경우, 평균 초당 전송량은 500 TPS입니다.

• 앱코어와 동일한 환경에서의 레거시 에이전트(대당 인증코드 5개, 총 13대) 전송 테스트



레거시 에이전트를 13개 동시 전송하는 경우, 평균 초당 전송량은 약 20 TPS입니다.

• 앱코어와 동일한 환경에서의 레거시 에이전트(인증코드 5개, 총 1대) 전송 테스트

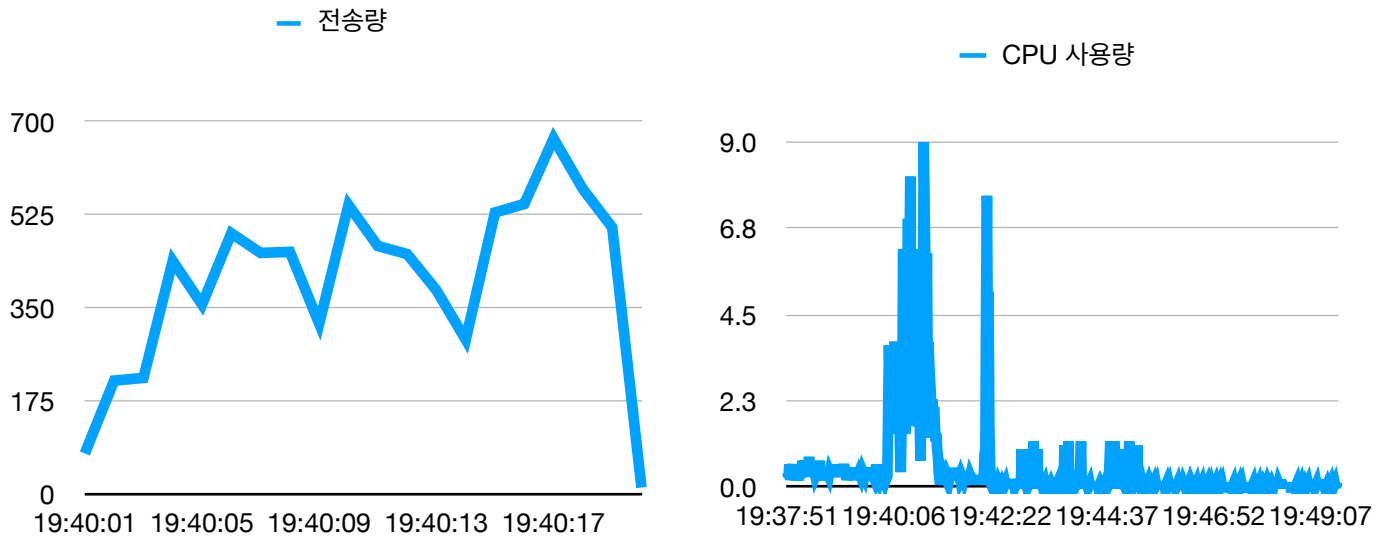


레거시 에이전트의 경우, 평균 초당 전송량은 약 35 TPS입니다.

이는 오히려 차세대 에이전트 1개의 처리 가능한 초당 전송량(평균 500TPS), 레거시 에이전트(약 35 TPS) 와 대 비해서도 떨어지는 것으로 확인됩니다.

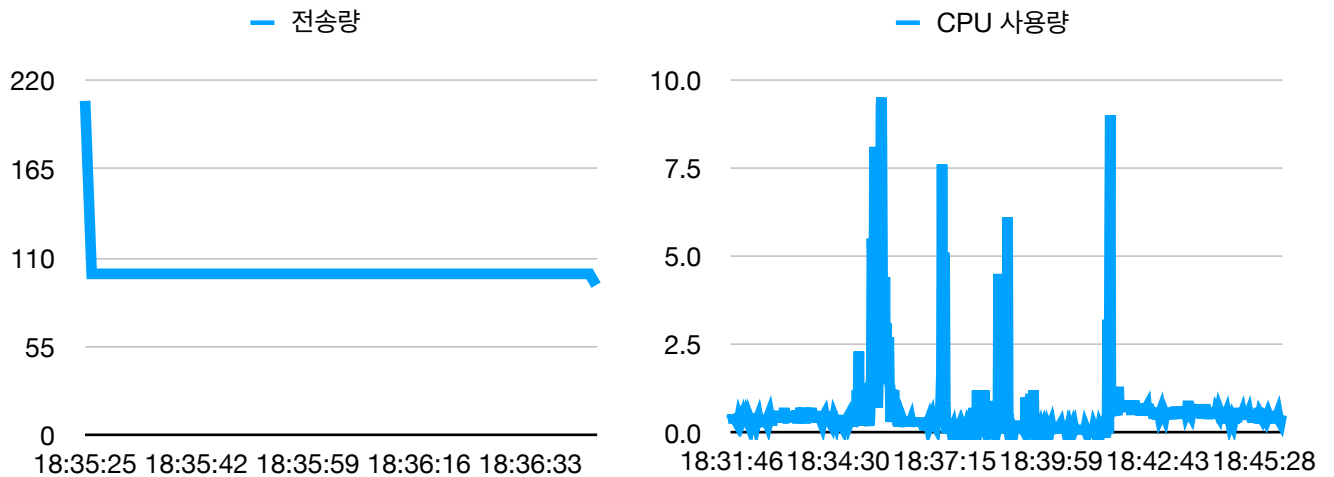
서버 자원이 충분하지 않은 서버에서 프로세스를 여러 대 설치 하는 것이 서버의 부하율을 높이고, 오히려 성능이 저 하되는 것으로 의심이 되어 성능이 더 좋은 서버에서 추가 테스트를 진행하였습니다.

• 16Core 서버에서 차세대 Agent 16대 설치 후 tps 800 설정 후 테스트



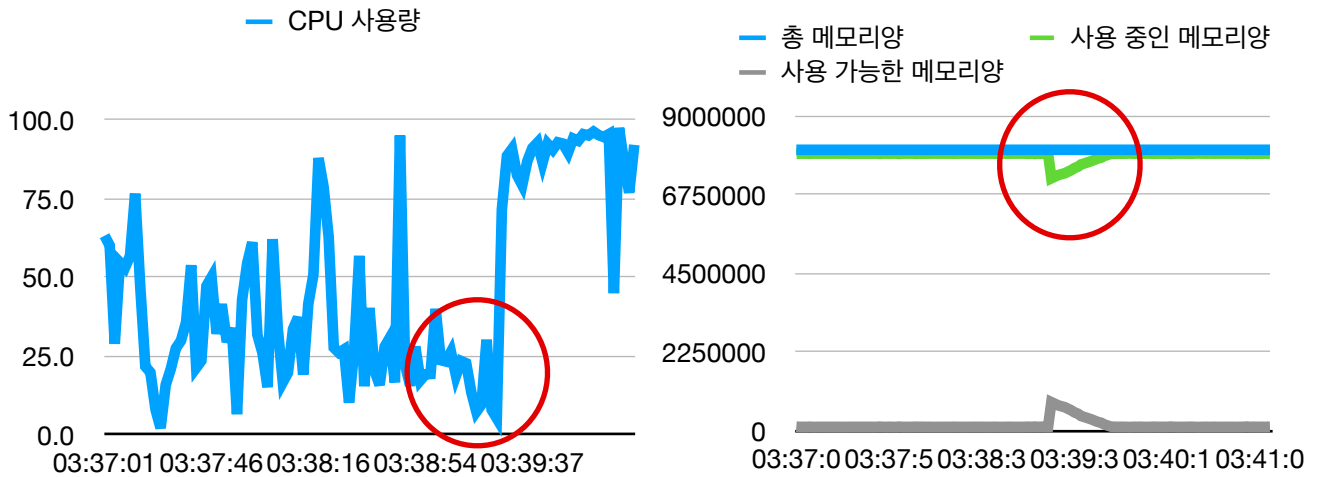
16 Core의 서버 환경에서 16개 차세대 에이전트의 TPS를 800으로 설정하고 메시지를 동시에 발송하는 경우, 에이전트의 평균 TPS는 350입니다.

• 16Core 서버에서 차세대 Agent 16대 설치 후 tps 100 설정 후 테스트



16 Core의 서버 환경에서 16개 차세대 에이전트의 TPS를 100으로 설정하고 메시지를 동시에 발송하는 경우, 에이전트의 평균 TPS는 110입니다.

• 차세대 에이전트에서 간헐적으로 CPU 사용량이 폭증하던 이유



이번에 차세대 에이전트에서 간헐적으로 CPU 사용량이 폭증하거나 전송이 지연되었던 이유는 서버 자원 대비 에이전트가 많이 설치되어 **에이전트 간 자원 경쟁**과 프로세스에서 주기적으로 발생하는 GC로 인해 CPU 사용량이 증가하였기 때문으로 확인됩니다.

• 차세대 에이전트 설치 시 서버 환경을 고려한 에이전트 대수 및 TPS 가이드

차세대 에이전트는 서버의 CPU Core 수에 정비례하게 성능이 향상되므로, 에이전트를 설치할 서버의 환경에 따라 다른 에이전트의 수, TPS 값으로 설정하는 것이 중요합니다.

100 +- 30 정도의 안정적 TPS로 메시지를 전송하고 싶으신 경우에는 CPU 코어 수 = 에이전트 수로 설정해 주시고, 더 빠른 TPS로 메시지를 전송하고 싶으신 경우에는 에이전트 수를 CPU 코어 수보다 적게 설정해 주셔야 합니다.

아래는 내부 테스트를 진행하여 확인한 8코어와 16 코어에서 안정적으로 메시지 전송이 가능한 에이전트 대수와 TPS에 대한 가이드입니다.

• 8Core

1개 에이전트 : 600TPS 까지 안정적으로 처리

4개 에이전트 : 약 150TPS 까지 안정적으로 처리

8개 에이전트 : 각 100TPS 까지 안정적으로 처리

- 16Core

16개 에이전트 : 각 100TPS 까지 안정적으로 처리

만약 서버의 자원이 충분하지 않은 상태에서 에이전트를 여러 대를 설치하여 메시지를 전송하려 하는 경우, 안정적으로 TPS 가 보장되지 않거나 프로세스의 성능 저하, 간헐적으로 CPU 사용량이 증가하는 등의 문제가 발생할 수 있습니다. 에이전트 설치 및 TPS 설정에 관해 궁금한 점이 있는 경우, 스탠다드네트웍스로 문의부탁드립니다.