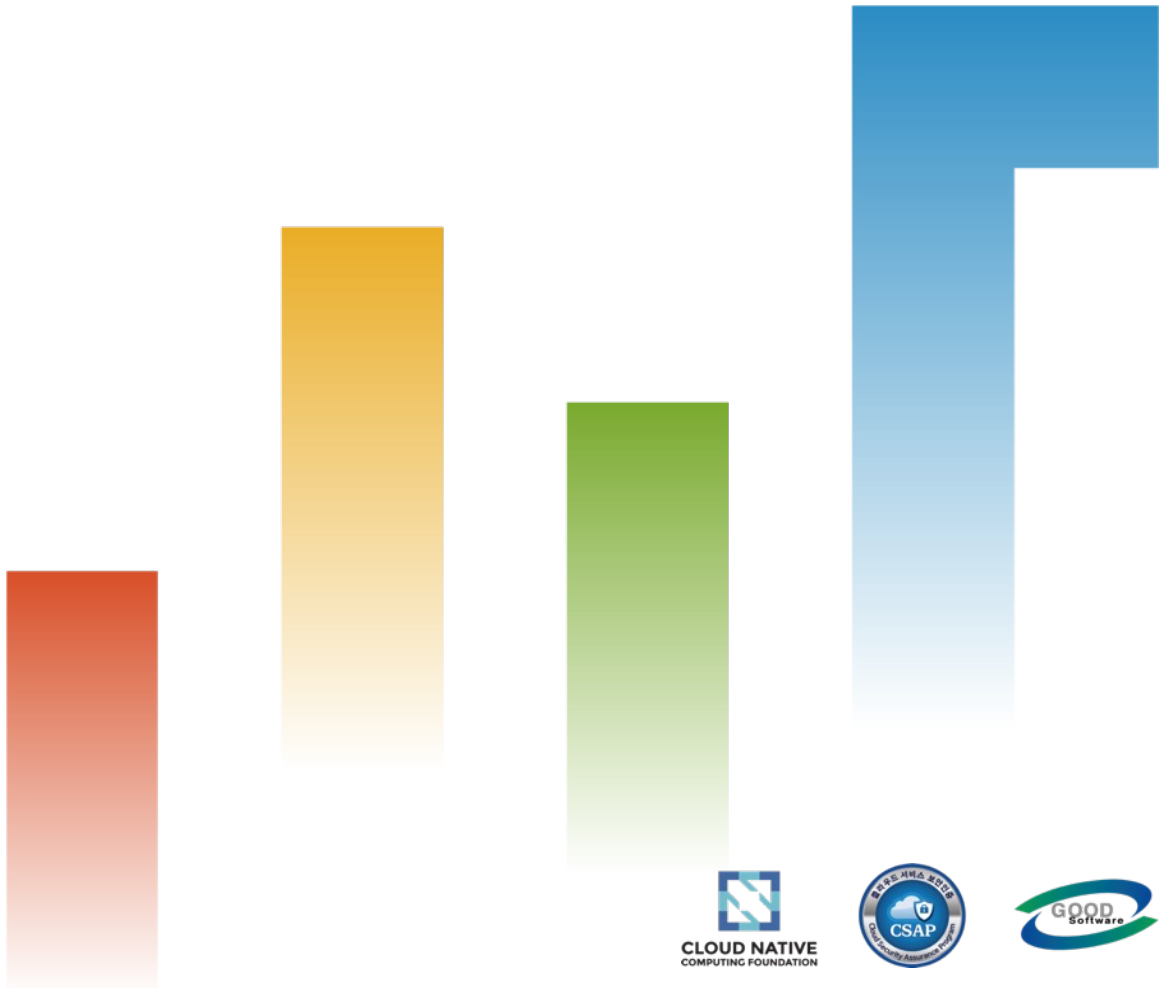


Kubernetes 모니터링

기술 문서 2024.04.02



Kubernetes 모니터링

컨테이너 가상화 기술은 도커가 등장하면서 크게 대중화되었습니다. 그리고 도커 사용이 증가하면서 컨테이너를 잘 사용하고 관리하기 위한 방안이 필요하게 되었으며 이에 다양한 컨테이너 오케스트레이션 도구들이 등장했습니다. 많은 컨테이너 오케스트레이션 도구 중 가장 널리 사용하는 것은 쿠버네티스입니다.

쿠버네티스는 컴퓨팅 리소스를 효율적으로 사용하기 위한 스케줄링 기능, 애플리케이션 업데이트를 위한 배포 기능, 그 외 컨테이너에 필요한 스토리지, 네트워크 등 다양한 기능을 제공합니다. 물론 그 자체로 거대한 플랫폼인 쿠버네티스도 모니터링은 필수적입니다.

하지만 쿠버네티스의 다양한 리소스들은 각기 다른 데이터 수집 방안을 필요로 합니다. 또 쿠버네티스 사용자마다 관심있는 지표가 다릅니다. 게다가 단순히 데이터를 수집하고 나열하는 것만으로는 모니터링 가시성을 확보할 수 없습니다. 다양한 모니터링 니즈를 충족하면서 문제 상황을 직관적으로 감지할 수 있는 UI가 필요합니다. 많은 사용자들이 와탭의 쿠버네티스 모니터링 솔루션을 찾는 이유이기도 합니다.

쿠버네티스 모니터링이란?

컨테이너 환경에서의 애플리케이션은 컨테이너화(Containerization)됩니다. 하지만 쿠버네티스는 컨테이너화된 애플리케이션을 Pod라는 단위로 배포합니다. 쿠버네티스의 최소 배포 단위는 컨테이너가 아닌 Pod이기 때문입니다. 컨테이너는 Pod에 담겨있으며, 1개 Pod에는 서로 다른 2개 이상의 컨테이너가 들어갈 수 있습니다. 따라서 쿠버네티스 모니터링을 위해서는 컨테이너 관점, Pod 관점의 모니터링이 모두 필요합니다. 와탭은 컨테이너와 Pod의 각각의 다양한 모니터링 데이터를 수집하여 별도의 카테고리에 저장합니다.

카테고리는 사용자 프로젝트로부터 수집하는 메트릭스(Metrics) 데이터를 구분하는 이름입니다. 각 카테고리는 서로 다른 태그와 필드로 구성되어 있습니다. 와탭 쿠버네티스의 컨테이너 맵은 크게 컨테이너 뷰와 Pod 뷰를 제공하여 사용자가 원하는 분석 대상을 선택할 수 있습니다.

쿠버네티스 모니터링 주요 특징점

- **쿠버네티스 기반의 통합 모니터링**
멀티 클라우드와 하이브리드 클라우드를 지원하여 쿠버네티스 기반의 시스템을 통합 모니터링합니다.
- **MSA에서 서비스별 호출관계 분석**
Pod 단위의 분산 애플리케이션 환경에서 트랜잭션의 API 호출 관계를 추적합니다.
- **상관관계를 한번에 파악**
Host, 컨테이너, 애플리케이션을 함께 모니터링하여 'Root Cause Analysis'를 해결할 수 있습니다.
- **컨테이너화된 애플리케이션의 상세 분석**

컨테이너에 숨어있는 Java, Node.js, Python 기반의 애플리케이션 분석에 필요한 히트맵, 트레이스 기능을 제공합니다. 특히, 모던 애플리케이션 환경에 유용한 멀티 서비스 대시보드, MSA 분석, 멀티 트랜잭션 추적까지 와탭 하나로 끝내세요.

- **컨테이너 맵을 통한 통합적이고 유연한 가시성 확보**

쿠버네티스 플랫폼 리소스 정보와 쿠버네티스에서 실행 중인 애플리케이션의 매트릭스, 이벤트 정보, 로그를 통합으로 제공합니다. 단일 뷰를 보며 인프라 담당자와 애플리케이션 담당자 사이의 원활한 의사 소통은 물론, 다양한 관점의 가시성을 확보할 수 있습니다.

- **리소스 상세 모니터링 및 사용량 추이 확인**

컨테이너와 Node의 리소스 사용량 및 추이 정보가 담긴 대시보드를 제공해 컨테이너들이 여러 Node 위에서 원활히 실행되도록 관리할 수 있습니다. 사용자는 와탭 프로젝트 내에 속한 컨테이너 또는 Node의 전체 목록에서 시작해 개별 대상에 대한 세부 상태까지 확인해 컨테이너와 Node를 더욱 면밀히 관찰할 수 있습니다.

- **마스터에 접속하지 않아도 메타 정보 조회 가능**

주요 리소스의 기본 메타 정보를 컨테이너 맵 상에서 바로 확인할 수 있으며, 별도 마스터 메타 대시보드를 통해 다양한 구성 요소의 전체 메타 정보를 한 눈에 확인할 수 있습니다. 명령어 기반으로 일일이 메타 정보를 확인할 필요가 없어 시간을 절약할 수 있습니다.

- **로그 관리 업무에서 벗어나 로그 분석에 집중**

컨테이너의 각종 로그 파일과 표준 출력을 실시간으로 저장해 수명 주기가 일정하지 않은 컨테이너 환경에서 로그를 확인할 수 있습니다. 컨테이너에 일일이 접속하지 않아도 애플리케이션에서 발생한 의미 있는 정보를 비롯해 주요 콘텐츠에 대한 분류 및 인덱싱 작업을 수행하여 로그 분석에만 집중할 수 있습니다.

모니터링 대상

쿠버네티스 모니터링 대상은 쿠버네티스 클러스터와 쿠버네티스 상에서 실행 중인 애플리케이션입니다.

- 쿠버네티스 클러스터 모니터링
 - 개별 컨테이너의 CPU, Memory, Disk(Volume), Network
 - Namespace, Deployment, ReplicaSet, Pod 등 서비스를 이루는 쿠버네티스 구성 요소
 - 개별 Node(호스트)의 CPU, Memory, Disk, Network
- 쿠버네티스에서 실행 중인 애플리케이션 모니터링: Java, Node.js, Python 등

이어서 설명할 모니터링 절차도 클러스터 모니터링을 위한 절차와 애플리케이션 모니터링을 위한 절차로 구분되어 진행합니다.

모니터링 절차

쿠버네티스 클러스터 모니터링 절차

1. [와탭 모니터링 서비스](#)에서 클러스터 프로젝트를 생성하세요.
와탭 모니터링 서비스 상에서 신규 쿠버네티스(Kubernetes) 프로젝트를 생성하면 해당 프로젝트가 **클러스터 프로젝트**입니다. [다음 문서](#)를 참조하세요.
2. 쿠버네티스 환경 내에 마스터 에이전트(whatap-master-agent) 및 노드 에이전트(whatap-node-agent)를 설치하세요. [다음 문서](#)를 참조하세요.
마스터 에이전트, 노드 에이전트는 사용자의 클러스터 내에서 Pod 형태로 실행합니다.
3. (선택사항) 클러스터 프로젝트 내 다수의 Namespace 존재 시 사용자는 와탭 모니터링 서비스에서 각 네임스페이스(Namespace) 프로젝트로 분리 가능합니다. [다음 문서](#)를 참조하세요.

✔ 쿠버네티스 Namespace는 다수의 팀과 사용자가 존재하는 환경에서 단일 클러스터 내 리소스 그룹 격리를 위한 매커니즘입니다. 사용자 쿠버네티스 클러스터 내의 Namespace 목록은 `kubectl get namespace` 명령어를 통해 확인할 수 있습니다.

쿠버네티스의 애플리케이션 모니터링 절차

컨테이너 내에서 실행 중인 애플리케이션 모니터링을 위한 마이크로 애플리케이션 에이전트 설치합니다. 지원하는 마이크로 애플리케이션은 Java, Node.js, Python 등입니다.

프로젝트 개념 이해하기

클러스터 및 프로젝트 구성


와탭 쿠버네티스 모니터링 서비스는 하나의 사용자 쿠버네티스 클러스터 환경을 대상으로 1개 이상의 와탭 프로젝트로 관리합니다.

사용자의 쿠버네티스 클러스터는 와탭 서비스 상에서 기본적으로는 1개의 **클러스터 프로젝트**로 표현합니다. 처음 쿠버네티스 상품의 프로젝트를 생성하면 해당 프로젝트가 바로 **클러스터 프로젝트**입니다. 만약 쿠버네티스 클러스터 내에 여러 네임스페이스가 존재하고, 컨테이너/파드(Pod) 수량이 많아 프로젝트를 나누려면, 쿠버네티스 내에 네임스페이스 별로 **네임스페이스(Namespace) 프로젝트**를 생성하여 분리할 수 있습니다.

프로젝트 목록

프로젝트명 / PCODE


☆ CP k8s
⚙️



노드	18	컨테이너	77	Namespace	3
노드 CPU 코어	516	Pod	34	Deployment	4
노드 메모리	4.2T	애플리케이션	0	ReplicaSet	5

1 Kubernetes

☆
⚙️



노드	1	컨테이너	1	Namespace	1
노드 CPU 코어	32	Pod	1	Deployment	1
노드 메모리	264.7G	애플리케이션	0	ReplicaSet	1

2 Namespace

1 클러스터 프로젝트

- 쿠버네티스 클러스터 내의 모든 자원 정보를 모니터링하기 위한 프로젝트입니다.
- **클러스터 프로젝트**를 통해 마스터 에이전트와 노드 에이전트 설치를 진행합니다.
- 프로젝트 목록에서 **클러스터 프로젝트**는 프로젝트 이름 옆에 CP 태그를, 가장 아래에 Kubernetes 태그를 표시합니다.
- **클러스터 프로젝트** 생성에 관한 자세한 내용은 [다음 문서](#)를 참조하세요.

2 네임스페이스 프로젝트

- 쿠버네티스 클러스터 하위 네임스페이스(Namespace) 단위의 프로젝트입니다.
- 네임스페이스(Namespace) 단위의 자원 모니터링 및 애플리케이션 컨테이너의 모니터링을 수행하기 위한 프로젝트입니다.
- 프로젝트 목록에서 **네임스페이스 프로젝트**는 가장 아래에 `Namespace` 태그를 표시합니다.
- **네임스페이스 프로젝트** 관리에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

쿠버네티스 모니터링 지원 환경

쿠버네티스 모니터링을 시작하기 전에 다음 지원 사항을 확인해보세요. 와탭 쿠버네티스는 표준 쿠버네티스 1.16~1.28 버전 및 해당 버전에 대응하는 다음 쿠버네티스 환경에서 설치할 수 있습니다.

지원 플랫폼

- Amazon EKS (Elastic Kubernetes Service)
- AKS (Azure Kubernetes Service)
- GKE (Google Kubernetes Engine)
- OKE (Oracle Container Infrastructure Container Engine for Kubernetes)
- Naver Ncloud Kubernetes Service
- NKS (NHN Kubernetes Service)
- Red Hat OpenShift (3.xx / 4.xx)
- Cocktail
- Accordion
- PaaS-TA
- KaKao i Cloud
- Rancher

① NVIDIA GPU 지원

브라우저 지원

와탭 모니터링 서비스는 웹브라우저와 모바일 앱에서 이용할 수 있습니다.

브라우저	권장여부	지원버전
Google Chrome	O	84 이상
Mozilla Firefox	X	최신 버전
Edge	X	최신 버전
Safari	X	최신 버전

- ① • 브라우저 호환성과 성능을 이유로 Chrome 최신 버전 사용을 권장합니다.
- 사용자 인터페이스(User Interface, UI)는 HTML5 표준 기술로 구현하여 Internet Explorer는 지원하지 않습니다.

방화벽

와탭 에이전트는 수집 서버 TCP 6600 포트로 접속 가능해야 합니다. 모니터링 대상과 가까운 수집 서버 주소를 허용하세요.

출발지: 와탭 에이전트

목적지	목적지 IP	포트
와탭 서울 수집 서버	13.124.11.223 / 13.209.172.35	TCP 6600
와탭 도쿄 수집 서버	52.68.36.166 / 52.193.60.176	TCP 6600
와탭 싱가포르 수집 서버	18.138.0.93 / 18.139.67.236	TCP 6600
와탭 뭄바이 수집 서버	13.127.125.69 / 13.235.15.118	TCP 6600



목적지	목적지 IP	포트
와탭 캘리포니아 수집 서버	52.8.223.130 / 52.8.239.99	TCP 6600
와탭 프랑크푸르트 수집 서버	3.125.142.162 / 3.127.76.140	TCP 6600

에이전트에서 수집 서버로 직접 접속할 수 없다면 제공하는 Proxy 모듈을 이용해 경유하세요.

Proxy

모바일 앱

와탭 모바일 앱은 안드로이드와 iOS 환경을 지원합니다. 다음 링크로 이동하거나 QR 코드를 스캔해 앱을 설치할 수 있습니다. 모바일 앱에 대한 자세한 설명은 [다음 문서](#)를 참조하세요.

iOS	Android
 <p>iOS 12 버전 이상</p>	 <p>Android 5.0 버전 이상</p>

에이전트 설치

와탭 쿠버네티스 모니터링 서비스 이용을 위한 기본 설치 방법을 안내합니다.

와탭 모니터링 서비스를 사용하기 위해서는 [회원 가입](#) 후 프로젝트를 생성하고 대상 서버에 에이전트를 설치해야 합니다. 회원 가입에 관한 자세한 내용은 [다음 문서](#)를 참조하세요.

다음 동영상 가이드를 참조하세요.

프로젝트 생성하기

에이전트를 설치하기 전에 먼저 프로젝트를 생성하세요.

1. [와탭 모니터링 서비스](#)로 이동한 다음 로그인하세요.
2. 프로젝트를 생성하려면 화면 왼쪽 사이드 메뉴에서 [전체 프로젝트](#) > [+ 프로젝트](#) 버튼을 선택하세요.
3. [상품 선택](#) 화면에서 프로젝트에 설치할 상품을 선택하세요.
4. [프로젝트 이름](#), [데이터 서버 지역](#), [타임 존](#) 등의 항목을 차례로 설정하세요.

프로젝트 이름 *

프로젝트 이름을(를) 입력해주세요

데이터 서버 지역 * AWS demo Azure Kakao Huawei 가까운 지역의 서버를 선택하세요 ▼

데이터 서버 지역을(를) 선택해주세요

데이터 서버 지역은 리전(클라우드 서비스를 제공하기 위해 설치한 데이터 센터의 묶음)을 의미합니다. 특정 리전을 선택하면 해당 리전에 속해있는 데이터 센터에 사용자의 데이터가 저장됩니다.

타임 존 * (GMT +9:00) Seoul, Tokyo, Yakutsk ▼

타임존은 알림, 보고서를 생성할 시 기준시간이 됩니다.

프로젝트 그룹 그룹 없음 ▼ 그룹 추가

여러개의 프로젝트를 그룹으로 묶어서 관리할 수 있습니다. 생성할 프로젝트가 속할 그룹이 있으면 선택해주세요.

프로젝트 설명

5. 모든 설정을 완료한 다음에는 [프로젝트 생성하기](#) 버튼을 선택하세요.

ⓘ • [데이터 서버 지역](#)은 리전(클라우드 서비스를 제공하기 위해 설치한 데이터 센터의 묶음)을 의미합니다. 특정 리전을

- ① 선택하면 해당 리전에 속한 데이터 센터에 사용자의 데이터를 저장합니다.
 - **타임 존**은 알림, 보고서를 생성하는 기준 시간입니다.
 - 여러 개의 프로젝트를 그룹으로 묶어 관리하려면 **프로젝트 그룹**에서 그룹을 선택하거나 그룹을 추가하세요. 그룹에 대한 자세한 설명은 **다음 문서**를 참조하세요.
 - 조직을 선택한 상태에서 프로젝트를 추가할 경우 **조직 하위 그룹**을 필수로 설정해야 합니다.

액세스 키 확인

액세스 키는 와탭 서비스 활성화를 위한 고유 ID입니다.

설치 안내 섹션에서 **프로젝트 액세스 키 발급받기** 버튼을 선택하세요. **액세스 키**를 자동으로 발급받은 후 다음 단계를 진행합니다.

- ⓘ **액세스 키**는 다른 프로젝트에 중복 사용할 수 없으며 **네임스페이스 프로젝트**에도 별도 **액세스 키**가 필요합니다.

- ① 프로젝트를 생성한 다음에는 자동으로 **에이전트 설치** 페이지로 이동합니다. **에이전트 설치** 페이지로 이동하지 않는다면 화면 왼쪽 메뉴에서 **관리 > 에이전트 설치**를 선택하세요.

마스터 및 노드 에이전트 설치

다음과 같이 마스터 및 노드 에이전트 설치 과정을 안내합니다. [설치 안내](#) 화면과 동일한 과정입니다.

ⓘ EKS Fargate는 지원 예정입니다.

사전 환경 점검 및 설정

사용자의 쿠버네티스 환경에 따라 사전 환경 설정이 필요할 수 있습니다. Istio, GKE, OpenShift 환경에 해당하지 않을 경우 [다음 단계](#)로 넘어가세요.

Istio 환경

GKE 환경

OpenShift 환경

Istio 환경일 경우 다음 명령을 실행해 와탭 수집서버와 통신하는 자원인 ServiceEntry를 생성하세요.

SH

```
kubectl create namespace whatap-monitoring
kubectl label namespace whatap-monitoring istio-injection=enabled
kubectl apply -f - <<EOF
apiVersion: networking.istio.io/v1alpha3
kind: ServiceEntry
metadata:
  name: whatap-external
spec:
  hosts:
  - whatap-proxy1
  - whatap-proxy2
  addresses:
  - 13.124.11.223
  - 13.209.172.35
  ports:
  - name: proxy
    number: 6600
    protocol: tcp
    location: MESH_EXTERNAL
EOF
```

Google Kubernetes Engine(GKE)의 경우 사용자 클러스터 생성 후 권한 설정을 위해 다음 명령을 실행하세요.

```
ACCOUNT=$(gcloud info --format='value(config.account)')
kubectl create clusterrolebinding owner-cluster-admin-binding \
--clusterrole cluster-admin \
--user $ACCOUNT
```

❗ gcloud 명령 설치에 대한 자세한 내용은 [구글 클라우드 문서](#)를 참조하세요.

설치 안내

- ▶ 1. 액세스 키 확인
- ▼ 2. 사전 환경 점검 및 설정
 - ▶ Istio 환경일 경우 추가 자원 생성
 - ▶ GKE 환경일 경우 추가 설정
 - ▼ OpenShift 환경일 경우 추가 설정

yaml 파일을 다운로드합니다.

↓ whatap_openshift_scc.yaml

명령어를 실행합니다.

oc create -f whatap_openshift_scc.yaml
복사

위 3가지 환경에 해당하지 않을 경우 다음 단계로 넘어가세요.

설치 안내 화면에서 [OpenShift 환경일 경우 추가 설정](#) 섹션을 열어 [whatap_openshift_scc.yaml](#) 버튼을 선택하세요. 파일을 다운로드 후 다음 명령을 실행하세요.

```
oc create -f whatap_openshift_scc.yaml
```

쿠버네티스 에이전트 생성

와탭 쿠버네티스 에이전트 설치 시 [helm](#) 차트 기준 안내와 [yaml](#) 파일 기준 안내를 다음과 같이 제공합니다.

- Helm Chart
- YAML

Helm 버전 확인 및 와탭 리포지토리 네트워크 연결 상태 확인

1. 다음 명령어를 통해 [helm](#) 버전을 확인하세요. 3.x 버전 이상을 지원합니다.

```
helm version
# version.BuildInfo{Version:"v3.x", GitCommit:"xxx", GitTreeState:"clean", GoVersion:"go1.20.8"}
```

2. 와탭 Helm 리포지토리에 대한 네트워크 연결 상태를 확인하세요.

```
curl https://whatap.github.io/helm/ping
# pong
```

Helm Chart 기준 설치

와탭 Helm 리포지토리 추가 후 사용자의 Container Runtime에 맞는 기본 설정 파일을 생성해 에이전트를 설치하세요.

1. 다음 명령어를 실행해 와탭 Helm 리포지토리를 추가하세요.

```
helm repo add whatap https://whatap.github.io/helm/
helm repo update
```

2. 다음 명령어를 실행해 사용자 **CONTAINER-RUNTIME**을 확인하세요.

```
kubectl get nodes -o wide
```

```
+ ~ kubectl get nodes -o wide
NAME      STATUS   ROLES    AGE     VERSION   INTERNAL-IP   EXTERNAL-IP   OS-IMAGE             KERNEL-VERSION   CONTAINER-RUNTIME
k1        Ready    control-plane  117d   v1.28.4   192.168.1.41   <none>         Ubuntu 20.04.3 LTS   5.4.0-173-generic containerd://1.7.9
k2        NotReady <none>    117d   v1.28.4   192.168.1.51   <none>         Ubuntu 20.04.3 LTS   5.4.0-167-generic containerd://1.7.9
k3        NotReady <none>    117d   v1.28.4   192.168.1.47   <none>         Ubuntu 20.04.3 LTS   5.4.0-167-generic containerd://1.7.9
```

3. CONTAINER-RUNTIME 확인 후 설치에 필요한 기본 설정 파일(values.yaml)을 다음과 같이 생성하세요.

```
containerRuntime: #CONTAINER-RUNTIME
whatap:
  license: #WHATAP-LICENSE-KEY
  host: "13.124.11.223/13.209.172.35"
  port: "6600"
```

4. 다음 명령어를 실행해 에이전트 애플리케이션을 설치하세요.

```
kubectl delete ns whatap-monitoring
kubectl delete clusterrole whatap
kubectl delete clusterrolebinding whatap
helm install whatap-kube-agent whatap/kube -f values.yaml
```

values.yaml 주요 설정 항목

와탭 쿠버네티스 에이전트 설정을 수정하기 위해서는 기본 설정 파일(values.yaml)을 수정해야 합니다. values.yaml 파일 내 주요 설정 항목을 다음과 같이 안내합니다.

키	타입	기본값	설명
whatap.license	String	# <license-key>	Whatap 라이선스 키
whatap.host	String	# <whatap-server-host>	Whatap 서버의 호스트 주소
whatap.port	Int	# <whatap-server-port>	Whatap 포트 번호

키	타입	기본값	설명
containerRuntime	String	"docker"	사용 중인 컨테이너 런타임. "docker", "containerd", "crio" 중 선택
daemonSet.name	String	whatap-node-agent	DaemonSet의 이름
daemonSet.label	String	whatap-node-agent	DaemonSet에 지정할 라벨
daemonSet.initContainers.nodeDebugger.enabled	Bool	true	Whatap 노드 디버거 컨테이너 활성화 여부
daemonSet.containers.nodeHelper.image	String	whatap/kube_mon	nodeHelper 컨테이너의 이미지
daemonSet.containers.nodeHelper.requests.memory	String	100Mi	nodeHelper MEMORY request
daemonSet.containers.nodeHelper.requests.cpu	String	100m	nodeHelper CPU request
daemonSet.containers.nodeHelper.limits.memory	String	350Mi	nodeHelper MEMORY limit
daemonSet.containers.nodeHelper.limits.cpu	String	200m	nodeHelper CPU limit
daemonSet.containers.nodeAgent.image	String	whatap/kube_mon	nodeAgent 컨테이너의 이미지

키	타입	기본값	설명
<code>daemonSet.containers.nodeAgent.requests.memory</code>	String	300Mi	nodeAgent MEMORY request
<code>daemonSet.containers.nodeAgent.requests.cpu</code>	String	100m	nodeAgent CPU request
<code>daemonSet.containers.nodeAgent.limits.memory</code>	String	350Mi	nodeAgent MEMORY limit
<code>daemonSet.containers.nodeAgent.limits.cpu</code>	String	200m	nodeAgent CPU limit
<code>deployment.name</code>	String	whatap-master-agent	Deployment의 이름
<code>deployment.label</code>	String	whatap-master-agent	Deployment에 지정할 라벨
<code>deployment.replicas</code>	Int	1	Deployment의 replica 수
<code>deployment.containers.controlPlaneHelper.enabled</code>	Boolean	true	whatap-control-plane-helper 컨테이너를 활성화 여부
<code>deployment.containers.controlPlaneHelper.image</code>	String	"whatap/kube_mon"	controlPlaneHelper 컨테이너의 이미지
<code>deployment.containers.controlPlaneHelper.port</code>	String	9496	controlPlaneHelper 컨테이너의 포트 번호입니다

키	타입	기본값	설명
deployment.containers.controlPlaneHelper.resources.requests.memory	String	"300Mi"	controlPlaneHelper 컨테이너 MEMORY request
deployment.containers.controlPlaneHelper.resources.requests.cpu	String	"100m"	controlPlaneHelper 컨테이너 CPU request
deployment.containers.controlPlaneHelper.resources.limits.memory	String	"350Mi"	controlPlaneHelper 컨테이너 MEMORY limit
deployment.containers.controlPlaneHelper.resources.limits.cpu	String	"200m"	controlPlaneHelper 컨테이너 CPU limit
deployment.containers.masterAgent.name	String	whatap-master-agent	masterAgent 컨테이너 이름
deployment.containers.masterAgent.image	String	"whatap/kube_mon"	masterAgent 컨테이너 이미지
deployment.containers.masterAgent.port	Int	6600	masterAgent 컨테이너 포트 번호
deployment.containers.masterAgent.resources.requests.memory	String	"300Mi"	masterAgent 컨테이너 MEMORY request
deployment.containers.masterAgent.resources.requests.cpu	String	"100m"	masterAgent 컨테이너 CPU request
deployment.containers.masterAgent.resources.limits.memory	String	"350Mi"	masterAgent 컨테이너 MEMORY

키	타입	기본값	설명
			limit
deployment.containers.masterAgent.resources.limits.cpu	String	"200m"	masterAgent 컨테이너 CPU request

ⓘ Helm 차트 기준 설치 문제 해결에 대한 안내는 [다음 문서](#)를 참조하세요.

YAML 파일 기준 설치

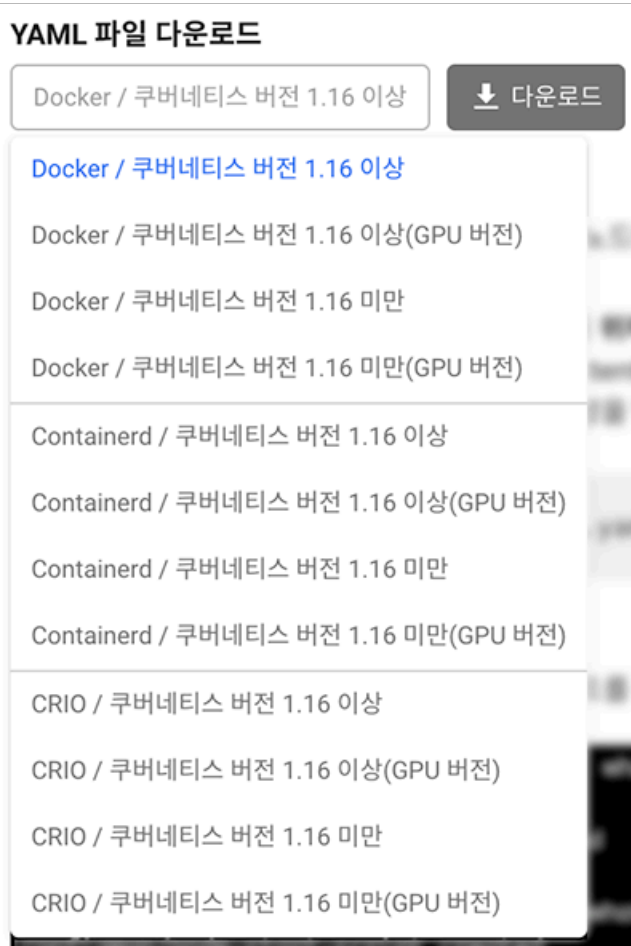
사용자의 쿠버네티스 버전과 Container Runtime에 맞는 **YAML** 파일을 다운로드해 에이전트를 생성하세요.

1. 다음 명령어를 실행해 **VERSION**과 **CONTAINER-RUNTIME**을 확인하세요.

```
kubectl get node -o wide
```

```
[root@k8s-master ~]# kubectl get node -o wide
NAME          STATUS    ROLES    AGE   VERSION   INTERNAL-IP   EXTERNAL-IP   OS-IMAGE             KERNEL-VERSION   CONTAINER-RUNTIME
k8s-master   Ready    control-plane,master   161d   v1.23.1   172.17.177.128 <none>         CentOS Linux 7 (Core) 3.10.0-1160.el7.x86_64 docker://20.10.12
k8s-worker1   Ready    <none>   161d   v1.23.1   192.168.170.143 <none>         CentOS Linux 7 (Core) 3.10.0-1160.el7.x86_64 docker://20.10.12
k8s-worker2   Ready    <none>   161d   v1.23.1   192.168.170.144 <none>         CentOS Linux 7 (Core) 3.10.0-1160.el7.x86_64 docker://20.10.12
[root@k8s-master ~]#
```

2. 에이전트 설치 화면에서 **VERSION**과 **CONTAINER-RUNTIME**에 맞는 **YAML** 파일을 선택하고 **다운로드** 버튼을 클릭하세요.



3. 다운로드한 **YAML** 파일을 마스터 노드에 업로드하세요.
4. 에이전트를 생성하기 위해 다음 명령어를 실행하세요.

```
# {whatap_kube_X.YZ.yaml} 파일명을 다운로드한 파일명과 일치하도록 수정하세요.
kubectl apply -f {whatap_kube_X.YZ.yaml}
```

5. 다음과 같이 에이전트를 정상적으로 생성(create)한 것을 확인하세요.

```
[root@kubernetes ~]# kubectl apply -f whatap_kube_1.16.yaml
namespace/whatap-monitoring created
persistentvolume/task-pv-volume created
serviceaccount/whatap created
clusterrole.rbac.authorization.k8s.io/whatap created
configmap/node-start-script created
clusterrolebinding.rbac.authorization.k8s.io/whatap created
daemonset.apps/whatap-node-agent created
configmap/master-start-script created
persistentvolumeclaim/whatap-confbase created
deployment.apps/whatap-master-agent created
service/whatap-master-agent created
[root@kubernetes ~]#
```

❗ 마스터 에이전트가 사용하는 자원량은 사용자 환경의 노드 수에 어느 정도 비례합니다. 즉 클러스터 내에 노드 수가 많을 경우 기본값으로 설정된 메모리 Limit(350MB) 이상의 자원을 사용해 OOM(Out Of Memory)이 발생할 수 있습니다. 마스터 에이전트에 할당된 메모리 Limit 수치는 에이전트 설치 [YAML](#) 파일을 통해 변경할 수 있습니다.

쿠버네티스 에이전트 설치 확인

쿠버네티스 에이전트 설치가 정상적으로 완료되었을 경우 다음 명령어를 실행해 마스터 에이전트와 노드 에이전트의 상태를 확인할 수 있습니다.

```
kubectl get pod -n whatap-monitoring
```

Running 상태인 경우

다음과 같이 **STATUS** 항목이 **Running**인 경우 설치를 완료한 상태입니다. 마스터 및 노드 에이전트의 설치를 완료하면 쿠버네티스 환경의 자원 모니터링을 시작합니다. [와탭 모니터링 서비스](#)로 이동한 다음 **프로젝트 목록**과 **대시보드 > 컨테이너 맵** 메뉴에서 성능 정보를 수집하는 것을 확인하세요.

```
[root@kubernetes ~]# kubectl get pod -n whatap-monitoring
NAME                                READY   STATUS    RESTARTS   AGE
whatap-master-agent-b4d7cbcc8-wblzt 1/1     Running  0          5m5s
whatap-node-agent-qw9d1             2/2     Running  0          5m5s
whatap-node-agent-rgjk8             2/2     Running  0          5m5s
whatap-node-agent-xvwt2             2/2     Running  0          5m5s
[root@kubernetes ~]#
```

Running 상태가 아닌 경우

다음과 같이 **STATUS** 항목이 **Running**이 아닌 경우 설치가 제대로 이루어지지 않은 상태입니다.

```
[root@kube-master ~]# kubectl get pod -n whatap-monitoring
NAME                                READY   STATUS    RESTARTS   AGE
whatap-master-agent-b4d7cbcc8-q79bw 0/1     Pending   0           31s
whatap-node-agent-6zc1n             2/2     Running   0           31s
whatap-node-agent-dsfxf             2/2     Running   0           31s
whatap-node-agent-n2vnq             2/2     Running   0           31s
[root@kube-master ~]#
```

1. 마스터 노드에서 다음 명령어를 실행해 Pod 생성의 실패 원인을 파악하세요.

```
kubectl describe pod {POD_NAME} -n whatap-monitoring
```

2. 실행 결과 하단에 **Events:** 항목을 확인하세요.

```
memory: 400Mi
Environment:
  WHATAP_LICENSE: x415g23t0qt5a-xecoppt4t37o6-xajnmdkvtjomo
  WHATAP_HOST: 13.124.11.223/13.209.172.35
  WHATAP_PORT: 6600
Mounts:
  /bin/entrypoint.sh from start-script-volume (ro,path="/bin/entrypoint.sh")
  /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-hsnzf (ro)
  /whatap_conf from whatap-config-volume (rw)
Conditions:
  Type           Status
  PodScheduled   False
Volumes:
  start-script-volume:
    Type:          ConfigMap (a volume populated by a ConfigMap)
    Name:          master-start-script
    Optional:      false
  whatap-config-volume:
    Type:          PersistentVolumeClaim (a reference to a PersistentVolumeClaim in the same namespace)
    ClaimName:    whatap-confbase
    ReadOnly:     false
  kube-api-access-hsnzf:
    Type:          Projected (a volume that contains injected data from multiple sources)
    TokenExpirationSeconds: 3607
    ConfigMapName: kube-root-ca.crt
    ConfigMapOptional: <nil>
    DownwardAPI: true
QoS Class:       Burstable
Node-Selectors:  <none>
Tolerations:    node.kubernetes.io/not-ready:NoExecute op=Exists for 300s
                 node.kubernetes.io/unreachable:NoExecute op=Exists for 300s
Events:
  Type           Reason             Age           From           Message
  ----           -
  Warning        FailedScheduling   2m14s        default-scheduler  0/3 nodes are available: 3 pod has unbound immediate PersistentVolumeClaims.
  Warning        FailedScheduling   54s          default-scheduler  0/3 nodes are available: 3 pod has unbound immediate PersistentVolumeClaims.
[root@kube-master ~]# kubectl describe pod whatap-master-agent-b4d7cbcc8-q79bw -n whatap-monitoring
```

✔ `kubectl` 명령어 사용 시 `-n {네임스페이스}` 옵션을 사용하면 특정 네임스페이스를 대상으로 작업을 수행합니다. 와탭 에이전트는 기본적으로 `whatap-monitoring` 이라는 네임스페이스에 설치되므로 `-n whatap-monitoring` 이라는 옵션을 추가합니다.

- ① • 애플리케이션 에이전트를 설치를 진행하려면 화면 오른쪽 아래에 [애플리케이션 에이전트 설치](#) 버튼을 선택하세요. 자세한 내용은 [다음 문서](#)를 참조하세요.
- 에이전트를 최신 버전으로 업데이트하려면 [다음 문서](#)를 참조하세요.

애플리케이션 에이전트 설치

컨테이너 내 Java, Python, Node.js 등의 애플리케이션을 모니터링하기 위한 에이전트 설치 단계입니다.

마이크로 애플리케이션 에이전트의 설치와 와탭의 애플리케이션 모니터링 에이전트 설치 절차와 동일합니다. 다만 컨테이너화(Containerization) 작업이 추가되는 것이 다릅니다.

애플리케이션 에이전트를 설치를 진행하려면 [관리 > 에이전트 설치](#) 메뉴의 화면 오른쪽 아래에 [애플리케이션 에이전트 설치](#) 버튼을 선택하세요.

설치 전 유의사항

쿠버네티스 에이전트는 **whatap-monitoring 네임스페이스**에 설치됩니다. 에이전트가 설치되는 네임스페이스는 변경할 수 없습니다.
WhaTap Server 127.0.0.1로 데이터를 전송하기 위해 **6606포트**가 열려 있어야 합니다. (TCP Outbound)

설치 안내

▼ 1. 액세스 키 확인

액세스 키는 와탭 서비스 활성화를 위한 고유 ID입니다.
 액세스 키가 더 이상 유효하지 않은 경우 정상적인 서비스 이용이 제한되며, 오른쪽 상단의 채널 문의를 통해 갱신하거나 연장할 수 있습니다.
 액세스 키는 **다른 프로젝트에 중복 사용할 수 없으며, 네임스페이스 프로젝트에도 별도 액세스 키가 필요합니다.**

O+ ID 복사

▶ 2. 사전 환경 점검 및 설정

▶ 3. 쿠버네티스 에이전트 생성

▶ 4. 쿠버네티스 에이전트 설치 확인

▶ 5. 에이전트 업데이트

▼ 펼치기

채널 문의

[애플리케이션 에이전트 설치 >](#)

액세스 키 확인

설치 안내 ▼ 펼치기 🔔 채널 문의

[Docker Java](#) [Docker Node JS](#) [Docker Python](#)

▼ **액세스 키 확인**

액세스 키는 와탭 서비스 활성화를 위한 고유 ID입니다.
 액세스 키가 더 이상 유효하지 않은 경우 정상적인 서비스 이용이 제한되며, 오른쪽 상단의 채널 문의를 통해 갱신하거나 연장 할 수 있습니다.
 액세스 키는 [다른 프로젝트에 중복 사용할 수 없으며, 클러스터\(마스터\) 프로젝트 인증키와 다릅니다.](#)

🔑 프로젝트 액세스 키 발급받기

▶ **와탭 에이전트 다운로드**

▶ **에이전트 설치 확인**

설치 안내 페이지의 **액세스 키 확인** 섹션에서 **프로젝트 액세스 키 발급받기** 버튼을 선택하세요. 이미 발급 받은 경우 액세스 키가 표시됩니다.

! **액세스 키**는 다른 프로젝트와 중복 사용할 수 없으며 **클러스터 프로젝트** 인증키와 다릅니다.

애플리케이션 에이전트 컨테이너화

다음 문서를 참조하여 애플리케이션 에이전트 컨테이너화를 진행하세요.

📖 Docker Java 설치

컨테이너 내 Java 애플리케이션을 모니터링하기 위한 에이전트 설치 단계입니다.

📖 Docker Node.js 설치

컨테이너 내 Node.js 애플리케이션을 모니터링하기 위한 에이전트 설치 단계입니다.

Docker Python 설치

컨테이너 내 Python 애플리케이션을 모니터링하기 위한 에이전트 설치 단계입니다.

Docker PHP 설치

컨테이너 내 PHP 애플리케이션을 모니터링하기 위한 에이전트 설치 단계입니다.

Docker Go 설치

컨테이너 내 Go 애플리케이션을 모니터링하기 위한 에이전트 설치 단계입니다.

Docker Java 설치

관리 > 에이전트 설치 > 하단의 **애플리케이션 설치** 클릭 > 설치 안내 중 **Docker Java** 탭 선택

Docker 컨테이너 기반으로 실행하는 Java 애플리케이션의 JVM Option에 에이전트 적용을 위한 설정을 추가하고 컨테이너 이미지를 패키징하는 과정을 다음과 같이 안내합니다.

- ❗ • EKS Fargate는 지원 예정입니다.
- Java 애플리케이션 연동 과정에 대한 이해를 돕고자 Git 예시 코드를 제공합니다. [다음 문서](#)를 참조하세요.

에이전트 다운로드

Docker Install 직접 설치

와탭 애플리케이션 에이전트 설치를 위해 **whatap.conf** 파일을 생성하세요. `{YOUR_PROJECT_ROOT}` 는 에이전트를 설치하려는 프로젝트의 최상위 경로로 변경하세요.

```
cat >{YOUR_PROJECT_ROOT}/whatap.conf <<EOL
whatap.server.host={proxyServer}
EOL
```

❗ **whatap.conf** 파일은 사용자 쿠버네티스 환경의 ConfigMap 기능을 통해 관리할 수 있습니다.

최신 버전 에이전트를 이미지에 포함 지정한 버전 에이전트를 이미지에 포함

최신 버전 에이전트를 애플리케이션 이미지에 포함시킬 수 있도록 안내합니다.

1. 애플리케이션 이미지에 와탭 에이전트를 포함시키려면 먼저 Dockerfile 최종 이미지 빌드 단계에서 다음 내용을 추가하세요. `{YOUR_PROJECT_ROOT}` 를 에이전트를 설치하려는 프로젝트의 최상위 경로로 변경하세요.

```
# 와탭 에이전트용 디렉토리를 사용자 컨테이너에 생성
RUN mkdir -p /whatap
```

```
# 와탭 Java 에이전트를 사용자 컨테이너에 복사
COPY --from=whatap/kube_mon /data/agent/micro/whatap.agent.kube.jar /whatap

# 생성한 whatap.conf 설정 파일을 사용자 컨테이너에 복사
COPY {YOUR_PROJECT_ROOT}/whatap.conf /whatap/
```

2. Dockerfile의 시작 명령어에 다음 내용을 추가하세요.

```
-javaagent:/whatap/whatap.agent.kube.jar -Dwhatap.micro.enabled=true
```

다음 Dockerfile의 정의 예시를 참고하세요.

Dockerfile

```
1 FROM openjdk:8-jdk-slim
2 RUN mkdir -p /app && mkdir /whatap
3 WORKDIR /app
4 COPY --from=whatap/kube_mon /data/agent/micro/whatap.agent.kube.jar /whatap
5 COPY ./whatap.conf /whatap
6 COPY ./target/myApp.jar /app/
7 CMD ["java", "-javaagent:/whatap/whatap.agent.kube.jar", "-Dwhatap.micro.enabled=true", "-jar", "/app/myApp.jar"]
8 EXPOSE 8080
```

3. Docker를 빌드하세요. {YOUR_DOCKERFILE_DIR} 를 Dockerfile이 위치한 경로로 변경하세요.

```
cd {YOUR_DOCKERFILE_DIR}
docker build -t {IMAGE_NAME} .
```

지정한 버전 에이전트를 애플리케이션 이미지에 포함시킬 수 있도록 안내합니다.

1. 애플리케이션 이미지에 와탭 에이전트를 포함시키려면 먼저 Dockerfile 최종 이미지 빌드 단계에서 다음 내용을 추가하세요. {YOUR_PROJECT_ROOT} 를 에이전트를 설치하려는 프로젝트의 최상위 경로로 변경하세요.

```
# 와탭 에이전트용 디렉터를 사용자 컨테이너에 생성
RUN mkdir -p /whatap
```

```
# 와탭 Java 에이전트를 사용자 컨테이너에 복사
COPY --from=whatap/kube_mon /data/agent/micro/whatap.agent-*.jar /whatap

# 생성한 whatap.conf 설정 파일을 사용자 컨테이너에 복사
COPY {YOUR_PROJECT_ROOT}/whatap.conf /whatap/
```

2. 현재 와탭 에이전트 버전을 확인하세요.

```
docker run whatap/kube_mon ls /data/agent/micro | grep -E 'whatap\.agent-(.*?\.)(.*?\.)(.*?\.)*jar' | sort | tail -1

# 출력 예시
## whatap.agent-X.Y.Z.jar
```

3. Dockerfile의 시작 명령어에 다음 내용을 추가하세요. X.Y.Z 부분에 앞서 확인한 에이전트 버전을 입력하세요.

```
-javaagent:/whatap/whatap.agent-X.Y.Z.jar -Dwhatap.micro.enabled=true
```

다음 Dockerfile의 정의 예시를 참조하세요.

```
Dockerfile
1 FROM openjdk:8-jdk-slim
2 RUN mkdir -p /app && mkdir /whatap
3 WORKDIR /app
4 COPY --from=whatap/kube_mon /data/agent/micro/whatap.agent-*.jar /whatap
5 COPY ./whatap.conf /whatap
6 COPY ./target/myApp.jar /app/
7 CMD ["java", "-javaagent:/whatap/whatap.agent-X.Y.Z.jar", "-Dwhatap.micro.enabled=true", "-jar", "/app/myApp.jar"]
8 EXPOSE 8080
```

4. Docker를 빌드하세요. {YOUR_DOCKERFILE_DIR} 를 Dockerfile이 위치한 경로로 변경하세요.

```
cd {YOUR_DOCKERFILE_DIR}
docker build -t {IMAGE_NAME} .
```

1. 설치 파일(whatap.agent.java.tar.gz)을 다운로드하려면 [다운로드](#) 버튼을 선택하세요.
2. Dockerfile을 작성할 수 있는 서버에 업로드한 다음 압축을 해제하세요.

❗ 설치할 서버에 직접 다운로드하려면 다음 명령어를 실행하세요.

```
wget https://api.whatap.io/agent/whatap.agent.java.tar.gz
```

3. 압축 해제 시 **whatap** 디렉터리가 생성됩니다. **whatap** 디렉터리 내부의 **whatap.conf** 파일에서 설정 내용을 확인 후 다음과 같이 수정하세요.

```
license={licenseKey}
whatap.server.host={proxyServer}
```

4. Dockerfile의 시작 명령어에 다음 내용을 추가하세요. **whatap** 디렉터리 내부에서 에이전트 파일을 확인 후 **X.Y.Z** 부분에 해당 버전을 입력하세요.

```
-javaagent:/whatap/whatap.agent-X.Y.Z.jar -Dwhatap.micro.enabled=true
```

다음 Dockerfile의 정의 예시를 참조하세요.

Dockerfile

```
FROM openjdk:8-jdk-alpine
RUN mkdir -p /app && mkdir /whatap
WORKDIR /app
COPY ./whatap.agent-X.Y.Z.jar /whatap/
COPY ./whatap.conf /whatap/
COPY ./target/myApp.jar myApp.jar
COPY ./paramkey.txt /whatap/
CMD ["java", "-javaagent:/whatap/whatap.agent-X.Y.Z.jar", "-Dwhatap.micro.enabled=true", "-jar", "/app/myApp.jar"]
EXPOSE 8080
```

- ❗ • **whatap-virtual-X.Y.Z.jar**: 모니터링 대상(샘플 애플리케이션)
- **whatap.agent-X.Y.Z.jar**: 와탭 에이전트
- 에이전트의 파일명이 **whatap.agent-1.2.3.jar**이라면 **X.Y.Z** 부분을 **1.2.3**으로 변경하세요.
- JVM 옵션 추가에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

✔ Java 17 버전 이상의 경우 reflection 관련 `--add-opens=java.base/java.lang=ALL-UNNAMED` 옵션을 추가하세요.

보안키 설정하기

SQL 변수, HTTP 쿼리를 조회하거나 Thread 중지 기능 이용하려면 보안키를 설정하세요.

- Java 에이전트 2.2.2 버전 이상

`$WHATAP_HOME` 경로에 `security.conf` 파일을 생성한 후 다음과 같이 6자리 영어, 숫자로 구성된 암호를 입력하세요.

security.conf

```
paramkey=ABCDEF # SQL 변수 및 HTTP 쿼리 조회
threadkill=ABCDEF # Thread 중지 기능
```

- Java 에이전트 2.2.2 버전 미만

`$WHATAP_HOME` 경로에 `paramkey.txt` 파일을 생성한 후 다음과 같이 6자리 영어, 숫자로 구성된 암호를 입력하세요.

paramkey.txt

```
ABCDEF # SQL 변수 및 HTTP 쿼리 조회, Thread 중지 기능
```

✔ Java 에이전트 2.2.2 버전 미만에서 2.2.2 버전으로 업데이트하는 경우 기존 `paramkey.txt` 파일의 키값은 `seuciry.conf` 파일의 키값으로 자동 적용됩니다. 예를 들어, `paramkey.txt`에 `FEDCBA` 를 사용한 상태에서 2.2.2 버전으로 업데이트했다면, `security.conf` 파일에는 다음과 같이 적용됩니다.

```
paramkey=FEDCBA
threadkill=FEDCBA
```

`paramkey.txt` 파일이 존재하지 않을 경우 `security.conf` 파일의 키값은 `WHATAP`이라는 키값으로 자동 생성됩니다.

- ⓘ • SQL 파라미터 정보 기록과 관련한 에이전트 설정은 [다음 문서](#)를 참조하세요.
- HTTP 파라미터 정보 기록과 관련한 에이전트 설정은 [다음 문서](#)를 참조하세요.

컨테이너 환경변수 설정

Docker 빌드 후 쿠버네티스 환경 내 컨테이너 환경변수를 설정하세요. 애플리케이션 배포 `yaml` 파일에 다음 내용을 추가하세요.

```
env:
- name: NODE_IP
  valueFrom: {fieldRef: {fieldPath: status.hostIP}}
- name: NODE_NAME
  valueFrom: {fieldRef: {fieldPath: spec.nodeName}}
- name: POD_NAME
  valueFrom: {fieldRef: {fieldPath: metadata.name}}
- name: OKIND
  value: {YOUR_OKIND_NAME}
- name: license
  value: {licenseKey}
- name: whatap_server_host
  value: {proxyServer}
- name: whatap_micro_enabled
  value: "true"
```

다음 예시를 참조하세요.

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: #DeploymentName
labels:
  app: #AppLabel
spec:
  replicas: 3
  selector:
    matchLabels:
      app: #AppLabel
  template:
    metadata:
      labels:
        app: #AppLabel
    spec:
      containers:
      - name: #ContainerName
        image: nginx
        ports:
        - containerPort: 80
```

```
env:
- name: NODE_IP
  valueFrom: {fieldRef: {fieldPath: "status.hostIP"}}
- name: NODE_NAME
  valueFrom: {fieldRef: {fieldPath: "spec.nodeName"}}
- name: POD_NAME
  valueFrom: {fieldRef: {fieldPath: "metadata.name "}}
- name: OKIND
  value: #DeploymentName
- name: license
  value: #licenseKey
- name: whatap_server_host
  value: #proxyServer
- name: whatap_micro_enabled
  value: "true"
```

ⓘ 환경변수 역할

- `NODE_IP` : 현재 Pod가 호스팅된 노드(Node)의 IP 주소를 수집합니다.
- `NODE_NAME` : 현재 Pod가 실행 중인 노드의 이름을 수집합니다.
- `POD_NAME` : 현재 Pod의 이름을 수집합니다.
- `OKIND` (선택사항): Pod에 해당하는 애플리케이션들을 그룹화합니다. 디플로이먼트(Deployment) 이름으로 설정하면 해당하는 Pod들을 하나로 그룹화합니다.
- `license` : 에이전트 인증 확인을 위한 키입니다.
- `whatap_server_host` : 와탭 수집 서버 호스트 IP입니다.
- `whatap_micro_enabled` : 컨테이너와의 연동을 활성화합니다.

에이전트 설치 확인

- 에이전트를 제대로 설치했는지 확인하려면 [대시보드](#) > [애플리케이션 서비스 대시보드](#) 메뉴로 이동하세요.
- 다운로드한 파일을 직접 설치한 다음 [대시보드](#) 메뉴에서 에이전트를 확인할 수 없다면 다음 사항을 확인하세요.
 - 컨테이너에서 `ps -ef | grep whatap` 명령어를 실행해 에이전트 옵션을 적용했는지 확인하세요.
 - 컨테이너의 `/whatap/logs` 경로의 내용을 확인하세요. 에이전트 로그는 `logs/{whatap 설정파일명}-yyyymmdd.log` 형식의 파일명으로 출력됩니다.

Docker Node.js 설치

관리 > 에이전트 설치 > 하단의 애플리케이션 설치 클릭 > 설치 안내 중 Docker Node.js 탭 선택

Docker 컨테이너 기반으로 실행하는 Node.js 애플리케이션에 와탭 모니터링 에이전트를 적용하고 컨테이너 이미지를 패키징하는 과정을 다음과 같이 안내합니다.

- ⓘ • EKS Fargate는 지원 예정입니다.
- Node.js 애플리케이션 연동 과정에 대한 이해를 돕고자 Git 예시 코드를 제공합니다. [다음 문서](#)를 참조하세요.

에이전트 다운로드

다음 명령어를 통해 whatap을 설치하세요.

```
npm install --save whatap
```

업데이트를 진행하려면 다음 명령어를 실행하세요.

```
npm update whatap
```

에이전트 설정

애플리케이션 메인 모듈의 첫 줄에 다음 코드를 추가하세요.

Example

```
const WhatapAgent = require('whatap').NodeAgent;
WhatapAgent.NodeAgent;
```

ECMAScript(ES) 활용 시 다음 코드를 추가하세요.

```
import WhatapAgent from 'whatap';
```

```
WhatapAgent.NodeAgent;
```

컨테이너 환경변수 설정

Docker 빌드 후 쿠버네티스 환경 내 컨테이너 환경변수를 설정하세요. 애플리케이션 배포 **yaml** 파일에 다음 내용을 추가하세요.

```
env:
- name: NODE_IP
  valueFrom: {fieldRef: {fieldPath: status.hostIP}}
- name: NODE_NAME
  valueFrom: {fieldRef: {fieldPath: spec.nodeName}}
- name: POD_NAME
  valueFrom: {fieldRef: {fieldPath: metadata.name}}
- name: OKIND
  value: {YOUR_OKIND_NAME}
- name: license
  value: {licenseKey}
- name: whatap_server_host
  value: {proxyServer}
- name: whatap_micro_enabled
  value: "true"
```

다음 예시를 참조하세요.

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: #DeploymentName
  labels:
    app: #AppLabel
spec:
  selector:
    matchLabels:
      app: #AppLabel
  replicas: 3
  template:
    metadata:
      labels:
        app: #AppLabel
    spec:
      containers:
```

```

- name: #ContainerName
image: nginx
ports:
- containerPort: 80
  env:
  - name: NODE_IP
    valueFrom: {fieldRef: {fieldPath: status.hostIP}}
  - name: NODE_NAME
    valueFrom: {fieldRef: {fieldPath: spec.nodeName}}
  - name: POD_NAME
    valueFrom: {fieldRef: {fieldPath: metadata.name}}
  - name: OKIND
    value: #DeploymentName
  - name: license
    value: {licenseKey}
  - name: whatap_server_host
    value: {proxyServer}
  - name: whatap_micro_enabled
    value: "true"

```

❗ 환경변수 역할

- `NODE_IP` : 현재 Pod가 호스팅된 노드(Node)의 IP 주소를 수집합니다.
- `NODE_NAME` : 현재 Pod가 실행 중인 노드의 이름을 수집합니다.
- `POD_NAME` : 현재 Pod의 이름을 수집합니다.
- `OKIND` (선택사항): Pod에 해당하는 애플리케이션들을 그룹화합니다. 디플로이먼트(Deployment) 이름으로 설정하면 해당하는 Pod들을 하나로 그룹화합니다.
- `license` : 에이전트 인증 확인을 위한 키입니다.
- `whatap_server_host` : 와탭 수집 서버 호스트 IP입니다.
- `whatap_micro_enabled` : 컨테이너와의 연동을 활성화합니다.

에이전트 설치 확인

- 에이전트를 제대로 설치했는지 확인하려면 [대시보드](#) > [애플리케이션 서비스 대시보드](#) 메뉴로 이동하세요.
- 다운로드한 파일을 직접 설치한 다음 [대시보드](#) 메뉴에서 에이전트를 확인할 수 없다면 다음 사항을 확인하세요.
 - 컨테이너에서 `ps -ef | grep whatap` 명령어를 실행해 에이전트 옵션을 적용했는지 확인하세요.

- 컨테이너의 `/whatap/logs` 경로의 내용을 확인하세요.

Docker Python 설치

관리 > 에이전트 설치 > 하단의 **애플리케이션 설치** 클릭 > 설치 안내 중 **Docker Python** 탭 선택

Docker 컨테이너 기반으로 실행하는 Python 애플리케이션에 와탭 모니터링 에이전트를 적용하고 컨테이너 이미지를 패키징하는 과정을 다음과 같이 안내합니다. 쿠버네티스 애플리케이션을 배포하기 위해 도커 이미지가 필요합니다. 다음 과정을 통해 whatap-python 패키지가 설치된 도커 이미지를 생성합니다.

ⓘ • EKS Fargate는 지원 예정입니다.

- Python 애플리케이션 연동 과정에 대한 이해를 돕고자 Git 예시 코드를 제공합니다. [다음 문서](#)를 참조하세요.

에이전트 다운로드

Python 애플리케이션 도커 이미지 빌드 시 whatap-python 패키지를 설치하세요.

```
RUN pip3 install --upgrade whatap-python
```

다음 Dockerfile의 정의 예시를 참조하세요.

```
python ver 3.10

# python3.10을 도커환경에 설치합니다.
FROM python:3.10

# 작업 디렉터리를 /app 으로 설정합니다.
WORKDIR /app

# 현재 디렉터리의 모든 파일과 폴더를 컨테이너 내의 /app 디렉터리에 복사합니다.
ADD ./app/

# Python에서는 pip를 이용하여 외부 라이브러리를 설치할 수 있습니다.
# 와탭 Python 에이전트를 도커 이미지 빌드시 설치합니다.
RUN pip3 install --upgrade whatap-python
```

에이전트 설정 및 실행

애플리케이션 실행 전 Python 에이전트 작업 디렉터리를 정하고 기본 설정 파일을 생성하세요. 설정 파일(entrpoint.sh)을 통해 에이전트에 인증 정보를 전달하고 로그 추적 여부 등을 설정합니다.

> 설정 파일(entrpoint.sh) 완성 예시

```

entrypoint.sh

#!/bin/bash

# 컨테이너의 작업 디렉터리를 와탭 경로로 설정하세요. 해당 경로에 에이전트 로그 및 설정 파일을 생성합니다.
export WHATAP_HOME=${PWD}

# 권한 오류 발생 시 다음 주석을 제거 후 진행하세요.
#chmod -R 777 $WHATAP_HOME

# 다음은 에이전트 구성에 필수적인 설정 값입니다. 애플리케이션 배포 yaml 파일을 통해 설정합니다.
whatap-setting-config \
--host $whatap_server_host \
--license $license \
--app_name $app_name \
--app_process_name $app_process_name

# 다음 주석은 에이전트 그룹화 및 로그 수집 활성화 등의 추가 설정입니다. 필요한 경우에만 사용하세요.

# 에이전트 그룹화
#echo "okind=$okind" >> whatap.conf

# 로그 수집 활성화
#echo "logsink_enabled=true" >> whatap.conf
#echo "logsink_trace_enabled=true" >> whatap.conf
#echo "trace_logging_enabled=true" >> whatap.conf

# 다음과 같이 whatap-start-agent를 애플리케이션 시작 명령어 앞에 추가해 에이전트를 실행하세요.
whatap-start-agent uvicorn server:app --host 0.0.0.0 --port 8000
    
```

1. 컨테이너 작업 디렉터리를 `WHATAP_HOME` 환경변수로 설정하세요. 해당 경로에 에이전트 로그 및 설정 파일을 생성합니다.

```

export WHATAP_HOME=${PWD}
    
```

2. `WHATAP_HOME` 으로 설정한 경로에 `whatap.conf` 파일을 생성하고 설정하도록 다음 명령어를 실행하세요. `$` 로 표시된 변수는 에이전트 구성에 필수적인 설정 값으로 애플리케이션 배포 `yaml` 파일을 통해 설정합니다.

```
whatap-setting-config \
--host $whatap_server_host \
--license $license \
--app_name $app_name \
--app_process_name $app_process_name
```

3. 애플리케이션 시작 명령어 앞에 다음과 같이 `whatap-start-agent` 를 추가해 에이전트를 실행하세요.

```
whatap-start-agent unicorn server:app --host 0.0.0.0 --port 8000
```

❗ 권한 문제가 발생하는 경우 다음과 같이 `$WHATAP_HOME` 에 권한을 부여하세요.

```
echo `sudo chmod -R 777 $WHATAP_HOME`
```

- ① `yaml` 작성 예제는 다음 [컨테이너 환경변수 설정](#) 을 참고하세요.
- 라이선스 (`license`)는 와탭의 사용자 인증 정보입니다. **외부에 노출해서는 안됩니다.**

스크립트 실행 예시

다음 Dockerfile 정의 예시를 참조하세요. `entrypoint.sh` 스크립트를 실행하는 완성된 예시입니다.

```
FROM python:3.10
WORKDIR /app
ADD ./app/
RUN pip3 install --upgrade whatap-python

# entrypoint.sh 스크립트를 컨테이너에서 실행할 수 있도록 권한을 부여합니다.
RUN chmod +x ./entrypoint.sh

# 컨테이너 생성시 entrypoint.sh 스크립트를 실행합니다.
CMD ["./entrypoint.sh"]
```

추가 설정

다음 설정은 선택사항으로 필요한 경우에만 사용하세요. 설정 파일([entrypoint.sh](#))에 다음 추가 설정 외에도 로그 및 트랜잭션 관련 설정이 가능합니다. 기타 추가 설정의 경우 [다음 문서](#)를 참조하세요.

- 에이전트 그룹화

```
echo "okind=$okind" >> whatap.conf
```

- 로그 수집 활성화

```
echo "logsink_enabled=true" >> whatap.conf
echo "logsink_trace_enabled=true" >> whatap.conf
echo "trace_logging_enabled=true" >> whatap.conf
```

컨테이너 환경변수 설정

Docker 빌드 후 쿠버네티스 환경 내 컨테이너 환경변수를 설정하세요. 애플리케이션 배포 [yaml](#) 파일에 다음 내용을 추가하세요.

```
env:
  - name: license
    value: XXXXXXXXXXXXXXXXXXXX-XXXXXXXXXXXXXXXX-XXXXXXXXXXXXXXXX
  - name: whatap_server_host
    value: XXX.XXX.XXX.XXX
  - name: app_name
    value: {YOUR_APP_NAME}
  - name: app_process_name
    value: {YOUR_PROCESS_NAME}
  - name: okind
    value: {YOUR_GROUP_NAME}
  - name: NODE_IP
    valueFrom: {fieldRef: {fieldPath: status.hostIP}}
  - name: NODE_NAME
    valueFrom: {fieldRef: {fieldPath: spec.nodeName}}
  - name: POD_NAME
    valueFrom: {fieldRef: {fieldPath: metadata.name}}
```

다음 예시를 참조하세요.


```

apiVersion: apps/v1
kind: Deployment
metadata:
  name: python-fastapi-deployment
spec:
  replicas: 3
  selector:
    matchLabels:
      app: python-fastapi-pod
  template:
    metadata:
      labels:
        app: python-fastapi-pod
  containers:
    - name: agent-python-fastapi
      image: whatap/agent-python-fastapi
      env:
        - name: license
          value: XXXXXXXXXXXXXXXX-XXXXXXXXXXXXXXXX-XXXXXXXXXXXXXXXX
        - name: whatap_server_host
          value: XXX.XXX.XXX.XXX
        - name: app_name
          value: "myapp-python-fastapi"
        - name: app_process_name
          value: "uvicorn"
        - name: NODE_IP
          valueFrom: {fieldRef: {fieldPath: status.hostIP}}
        - name: NODE_NAME
          valueFrom: {fieldRef: {fieldPath: spec.nodeName}}
        - name: POD_NAME
          valueFrom: {fieldRef: {fieldPath: metadata.name}}

```

⚠ 환경변수 역할

- `license` : 에이전트 인증 확인을 위한 키입니다.
- `whatap_server_host` : 와탭 수집 서버 호스트 IP입니다.
- `app_name` : 애플리케이션의 에이전트를 식별하는 이름입니다. 에이전트 식별에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.
- `app_process_name` : CPU, Heap Memory 등 수집할 대상 프로세스를 설정합니다. 예로 uwsgi, gunicorn 등이 있습니다.
- `okind` (선택사항) : Pod에 해당하는 애플리케이션들을 그룹화합니다. 디플로이먼트(Deployment) 이름으로 설정 시



해당하는 Pod들을 하나로 그룹화합니다.

- `NODE_IP` : 현재 Pod가 호스팅된 노드(Node)의 IP 주소를 수집합니다.
- `NODE_NAME` : 현재 Pod가 실행 중인 노드의 이름을 수집합니다.
- `POD_NAME` : 현재 Pod의 이름을 수집합니다.

에이전트 설치 확인

- 에이전트를 제대로 설치했는지 확인하려면 [대시보드](#) > [애플리케이션 서비스 대시보드](#) 메뉴로 이동하세요.
- 컨테이너에서 `ps -ef | grep whatap_python` 명령어를 실행해 와탭 Python 서비스가 정상적으로 실행되는지 확인하세요.

Docker PHP 설치

관리 > 에이전트 설치 > 하단의 애플리케이션 설치 클릭 > 설치 안내 중 **Docker PHP** 탭 선택

Docker 컨테이너 기반으로 실행하는 PHP 애플리케이션에 와탭 모니터링 에이전트를 적용하고 컨테이너 이미지를 패키징하는 과정을 다음과 같이 안내합니다.

❗ EKS Fargate는 지원 예정입니다.

에이전트 다운로드

PHP 애플리케이션 도커 이미지 빌드 시 whatap-php 패키지를 설치하세요.

Red Hat/CentOS

Debian/Ubuntu

Amazon Linux

Alpine Linux

Red Hat/CentOS

```
RUN rpm -Uvh https://repo.whatap.io/centos/5/noarch/whatap-repo-1.0-1.noarch.rpm
RUN yum install -y whatap-php
```

Debian/Ubuntu

```
RUN wget https://repo.whatap.io/debian/release.gpg -O - | apt-key add -
RUN wget https://repo.whatap.io/debian/whatap-repo_1.0_all.deb
RUN dpkg -i whatap-repo_1.0_all.deb
RUN apt-get update
RUN apt-get install -y whatap-php
```

Amazon Linux

```
RUN rpm --import https://repo.whatap.io/centos/release.gpg
```

```

RUN echo "[whatap]" | tee /etc/yum.repos.d/whatap.repo > /dev/null
RUN echo "name=whatap packages for enterprise linux" | tee -a /etc/yum.repos.d/whatap.repo > /dev/null
RUN echo "baseurl=https://repo.whatap.io/centos/latest/^\$basearch" | tee -a /etc/yum.repos.d/whatap.repo > /dev/null
RUN echo "enabled=1" | tee -a /etc/yum.repos.d/whatap.repo > /dev/null
RUN echo "gpgcheck=0" | tee -a /etc/yum.repos.d/whatap.repo > /dev/null
RUN yum install -y whatap-php
    
```

Alpine Linux

```

RUN wget https://s3.ap-northeast-2.amazonaws.com/repo.whatap.io/alpine/x86_64/whatap-php.tar.gz
RUN tar -xvzf whatap-php.tar.gz -C /
    
```

에이전트 설정

- 다음 명령어를 이용해 설치 스크립트를 설정하세요. 와탭 액세스 키, 서버 IP 정보를 입력하면 PHP 환경 정보를 자동으로 수집합니다.

```

RUN (echo "[액세스 키]"; echo "[서버 IP]")|/usr/whatap/php/install.sh
    
```

- 다음과 같이 PHP 환경 정보를 직접 입력할 수 있습니다.

```

# RUN /usr/whatap/php/install.sh -l [액세스 키] -s [서버 IP] -v [PHP 버전 x.x] -i [php.ini의 전체 경로] -p [프로세스명] -z [thread safety]
RUN /usr/whatap/php/install.sh -l xxxxx -s 1.1.1.1/2.2.2.2 -v 7.0 -i /etc/php.ini -p httpd [-z]
    
```

ⓘ 액세스 키가 Docker Public 상태의 이미지에 노출되지 않도록 주의하세요.

컨테이너 환경변수 설정

Docker 빌드 후 쿠버네티스 환경 내 컨테이너 환경변수를 설정하세요. 애플리케이션 배포 **yaml** 파일에 다음 내용을 추가하세요.

```
env:
- name: NODE_IP
  valueFrom: {fieldRef: {fieldPath: status.hostIP}}
- name: NODE_NAME
  valueFrom: {fieldRef: {fieldPath: spec.nodeName}}
- name: POD_NAME
  valueFrom: {fieldRef: {fieldPath: metadata.name}}
```

다음 예시를 참조하세요.

```
apiVersion: v1
kind: Pod
metadata:
  name: # Pod Name
  labels: # Pod Labels
spec:
  containers:
    env:
      - name: NODE_IP
        valueFrom: {fieldRef: {fieldPath: status.hostIP}}
      - name: NODE_NAME
        valueFrom: {fieldRef: {fieldPath: spec.nodeName}}
      - name: POD_NAME
        valueFrom: {fieldRef: {fieldPath: metadata.name}}
```

ⓘ 환경변수 역할

- `NODE_IP` : 현재 Pod가 호스팅된 노드(Node)의 IP 주소를 수집합니다.
- `NODE_NAME` : 현재 Pod가 실행 중인 노드의 이름을 수집합니다.
- `POD_NAME` : 현재 Pod의 이름을 수집합니다.

에이전트 실행

```
Command | Supervisor
```

애플리케이션 시작 명령어 앞에 추가로 whatap-php 시작 명령어를 추가하세요.

```
sh -c "/usr/whatap/php/whatap-php start && [애플리케이션 시작 명령어]"
```

Supervisor로 애플리케이션을 시작하는 경우 다음 코드를 참조해 whatap-php 서비스를 추가하세요.

```
RUN echo "[program:whatap-php]" >> /etc/supervisor/conf.d/whatap.conf
RUN echo "command = /usr/whatap/php/whatap-php start" >> /etc/supervisor/conf.d/whatap.conf
RUN echo "user = root" >> /etc/supervisor/conf.d/whatap.conf
RUN echo "startsecs = 0" >> /etc/supervisor/conf.d/whatap.conf
RUN echo "autostart = true" >> /etc/supervisor/conf.d/whatap.conf
RUN echo "autorestart = false" >> /etc/supervisor/conf.d/whatap.conf
RUN echo "stdout_logfile = /dev/stdout" >> /etc/supervisor/conf.d/whatap.conf
RUN echo "stdout_logfile_maxbytes=0" >> /etc/supervisor/conf.d/whatap.conf
RUN echo "stderr_logfile = /dev/stderr" >> /etc/supervisor/conf.d/whatap.conf
RUN echo "stderr_logfile_maxbytes=0" >> /etc/supervisor/conf.d/whatap.conf
```

다음 명령어를 실행해 와탭 서비스가 정상 실행되었는지 확인하세요. 애플리케이션 서버가 실행되면 에이전트가 모니터링 정보를 수집하기 시작합니다.

```
ps -ef | grep whatap_php
```

에이전트 설치 확인

에이전트를 제대로 설치했는지 확인하려면 [대시보드](#) > [애플리케이션 서비스 대시보드](#) 메뉴로 이동하세요.

다운로드한 파일을 직접 설치한 다음, [대시보드](#) 메뉴에서 에이전트를 확인할 수 없다면 다음 사항을 확인하세요.

- 컨테이너에서 `ps -ef | grep whatap` 명령어를 실행해 에이전트 옵션을 적용했는지 확인하세요.
- 컨테이너의 `/usr/whatap/php/logs` 경로의 내용을 확인하세요. 에이전트 로그는 `logs/whatap-{boot or install}--yyyymmdd.log` 형식의 파일명으로 출력됩니다.

Docker Go 설치

관리 > 에이전트 설치 > 하단의 애플리케이션 설치 클릭 > 설치 안내 중 Docker Go 탭 선택

Docker 컨테이너 기반으로 실행하는 Go 애플리케이션에 와탭 모니터링 에이전트를 적용하고 컨테이너 이미지를 패키징하는 과정을 다음과 같이 안내합니다.

ⓘ EKS Fargate는 지원 예정입니다.

에이전트 다운로드

Go 애플리케이션 도커 이미지 빌드 시 whatap-agent 패키지를 설치하세요.

- Red Hat/CentOS
- Debian/Ubuntu
- Amazon Linux
- Alpine Linux

Red Hat/CentOS

```

RUN rpm -Uvh https://repo.whatap.io/centos/5/noarch/whatap-repo-1.0-1.noarch.rpm
RUN yum install -y whatap-agent
    
```

Debian/Ubuntu

```

RUN wget https://repo.whatap.io/debian/release.gpg -O - | apt-key add -
RUN wget https://repo.whatap.io/debian/whatap-repo_1.0_all.deb
RUN dpkg -i whatap-repo_1.0_all.deb
RUN apt-get update
RUN apt-get install -y whatap-agent
    
```

Amazon Linux

```

RUN rpm --import https://repo.whatap.io/centos/release.gpg
RUN echo "[whatap]" | tee /etc/yum.repos.d/whatap.repo > /dev/null
RUN echo "name=whatap packages for enterprise linux" | tee -a /etc/yum.repos.d/whatap.repo > /dev/null
RUN echo "baseurl=https://repo.whatap.io/centos/latest/\${basearch}" | tee -a /etc/yum.repos.d/whatap.repo > /dev/null
RUN echo "enabled=1" | tee -a /etc/yum.repos.d/whatap.repo > /dev/null
RUN echo "gpgcheck=0" | tee -a /etc/yum.repos.d/whatap.repo > /dev/null
    
```

```
RUN yum install -y whatap-agent
```

```
Alpine Linux
```

```
RUN wget https://s3.ap-northeast-2.amazonaws.com/repo.whatap.io/alpine/x86_64/whatap-agent.tar.gz
RUN tar -xvzf whatap-agent.tar.gz -C /
```

에이전트 설정

사용자 애플리케이션 실행 경로에 와탭의 액세스 키, 서버 IP 주소 정보를 입력한 `whatap.conf` 파일을 생성하세요. 특정 경로를 사용할 경우 `WHATAP_HOME` 환경변수로 설정할 수 있습니다.

```
whatap.conf

RUN echo "accesskey={액세스 키}" >> whatap.conf
RUN echo "whatap.server.host={수집 서버 IP 주소}" >> whatap.conf
```

ⓘ 액세스 키가 Docker Public 상태의 이미지에 노출되지 않도록 주의하세요.

WHATAP_HOME 환경변수 설정하기

`whatap.conf` 파일 경로를 `WHATAP_HOME` 환경변수로 설정할 수 있습니다. `WHATAP_HOME` 경로를 우선 생성하세요. 애플리케이션 실행 문구 앞에 `WHATAP_HOME` 설정을 추가합니다.

```
mkdir ./whatap_home
echo "license={액세스 키}" >> ./whatap_home/whatap.conf
echo "whatap.server.host={수집 서버 IP 주소}" >> ./whatap_home/whatap.conf

# run application
WHATAP_HOME=./whatap_home ./app
```

Go 라이브러리 설정하기

Go 애플리케이션 소스 코드에 github.com/whatap/go-api 패키지를 추가하세요.


```
go get github.com/whatap/go-api
```

`trace.Init()`, `trace.Shutdown()` 함수로 초기화 및 종료를 설정하세요. `trace.Start()`, `trace.End()` 함수로 트랜잭션의 시작 종료를 설정하세요.

Go

```
import "github.com/whatap/go-api/trace"

func main(){
    trace.Init(nil)
    //It must be executed before closing the app.
    defer trace.Shutdown()

    ctx, _ := trace.Start(context.Background(), "Start Tx")

    ...

    trace.End(ctx, err)
}
```

ⓘ 라이브러리 설정에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

컨테이너 환경변수 설정

Docker 빌드 후 쿠버네티스 환경 내 컨테이너 환경변수를 설정하세요. 애플리케이션 배포 `yaml` 파일에 다음 내용을 추가하세요.

```
env:
- name: NODE_IP
  valueFrom: {fieldRef: {fieldPath: status.hostIP}}
- name: NODE_NAME
  valueFrom: {fieldRef: {fieldPath: spec.nodeName}}
- name: POD_NAME
  valueFrom: {fieldRef: {fieldPath: metadata.name}}
```

다음 예시를 참조하세요.

```
apiVersion: v1
kind: Pod
metadata:
  name: # Pod Name
  labels: # Pod Labels
spec:
  containers:
    env:
      - name: NODE_IP
        valueFrom: {fieldRef: {fieldPath: status.hostIP}}
      - name: NODE_NAME
        valueFrom: {fieldRef: {fieldPath: spec.nodeName}}
      - name: POD_NAME
        valueFrom: {fieldRef: {fieldPath: metadata.name}}
```

ⓘ 환경변수 역할

- `NODE_IP` : 현재 파드가 호스팅된 노드(Node)의 IP 주소를 수집합니다.
- `NODE_NAME` : 현재 파드가 실행 중인 노드의 이름을 수집합니다.
- `POD_NAME` : 현재 파드의 이름을 수집합니다.

에이전트 실행

```
Command | Supervisor
```

애플리케이션 시작 명령어 앞에 추가로 whatap-agent 시작 명령어를 추가하세요.

```
sh -c "/usr/whatap/agent/whatap-agent start && [애플리케이션 시작 명령어]"
```

Supervisor로 애플리케이션을 시작하는 경우 다음 코드를 참조해 whatap-agent 서비스를 추가하세요.

```
RUN echo "[program:whatap-agent]" >> /etc/supervisor/conf.d/whatap.conf
RUN echo "command = /usr/whatap/agent/whatap-agent start" >> /etc/supervisor/conf.d/whatap.conf
RUN echo "user = root" >> /etc/supervisor/conf.d/whatap.conf
RUN echo "startsecs = 0" >> /etc/supervisor/conf.d/whatap.conf
RUN echo "autostart = true" >> /etc/supervisor/conf.d/whatap.conf
RUN echo "autorestart = false" >> /etc/supervisor/conf.d/whatap.conf
RUN echo "stdout_logfile = /dev/stdout" >> /etc/supervisor/conf.d/whatap.conf
```

```

RUN echo "stdout_logfile_maxbytes=0" >> /etc/supervisor/conf.d/whatap.conf
RUN echo "stderr_logfile = /dev/stderr" >> /etc/supervisor/conf.d/whatap.conf
RUN echo "stderr_logfile_maxbytes=0" >> /etc/supervisor/conf.d/whatap.conf
    
```

다음 명령어를 실행해 와탭 서비스가 정상 실행되었는지 확인하세요. 애플리케이션 서버가 실행되면 에이전트가 모니터링 정보를 수집하기 시작합니다.

```
ps -ef | grep whatap_agent
```

에이전트 설치 확인

에이전트를 제대로 설치했는지 확인하려면 [대시보드](#) > [애플리케이션 서비스 대시보드](#) 메뉴로 이동하세요.

다운로드한 파일을 직접 설치한 다음, [대시보드](#) 메뉴에서 에이전트를 확인할 수 없다면 다음 사항을 확인하세요.

- 컨테이너에서 `ps -ef | grep whatap_agent` 명령어를 실행해 에이전트 옵션을 적용했는지 확인하세요.
- 컨테이너의 `WHATAP_HOME` `/logs` 또는 애플리케이션 실행 경로의 `logs` 디렉터리를 확인하세요. 에이전트 로그는 `logs/whatap-{boot or install}-yyyymmdd.log` 형식의 파일명으로 출력됩니다.

네임스페이스 프로젝트 만들기

다음과 같은 상황일 때 클러스터 프로젝트에서 네임스페이스 프로젝트 분리를 고려합니다.

- 단일 클러스터(Master) 프로젝트에 존재하는 리소스 수가 많고, 쿠버네티스 Namespace 기준 분리 가능한 경우
- 쿠버네티스 Namespace마다 부서 또는 담당자가 달라 특정 Namespace만 모니터링해야 할 경우

클러스터 프로젝트에 있는 각 쿠버네티스 Namespace를 별도의 와탭 네임스페이스로 분리하려면 [관리 > 네임스페이스 관리](#) 메뉴를 통해 프로젝트를 생성합니다.

The screenshot shows the 'Kubernetes Monitoring Demo(New)' interface. At the top, there are summary statistics for various resources:

컨테이너	Namespace	Pod	Deployment	ReplicaSet	에이전트 노드	노드 CPU 코어	노드 메모리	애플리케이션
38	5	34	0	0	4	8	18.8GiB	10

Below the summary, there is a '네임스페이스 목록' (Namespace List) table:

이름	그룹	컨테이너	Namespace	Pod	Deployment	ReplicaSet	에이전트 노드	노드 CPU 코어	노드 메모리	애플리케이션	설정
Kubernetes.Namespace.Virtual	Demo Project	4	1	2	2	2	1	2	3.7GiB	4	⊙
default											+ 프로젝트 추가
kube-flannel											+ 프로젝트 추가
kube-node-lease											+ 프로젝트 추가
virtual-main											+ 프로젝트 추가

ⓘ [관리 > 네임스페이스 관리](#) 메뉴는 클러스터 프로젝트 선택 시에만 표시됩니다. 네임스페이스 프로젝트에서는 해당 메뉴로 진입할 수 없습니다.

네임스페이스 목록에 **+ 프로젝트 추가** 버튼이 표시되는 경우 해당 버튼을 눌러 와탭 네임스페이스 프로젝트를 생성할 수 있습니다. 네임스페이스 이름을 입력한 다음 **확인** 버튼을 선택하세요.

프로젝트 추가
✕

네임스페이스

그룹 그룹을 선택해주세요. ▾

⚠ 네임스페이스 프로젝트를 추가하면, 클러스터 프로젝트에서 별도의 와탭 프로젝트로 분리됩니다. 따라서 네임스페이스 프로젝트 추가 후 삭제 하더라도 클러스터 프로젝트에서는 확인할 수 없습니다.

취소
확인

네임스페이스 프로젝트를 생성하면 새로운 별도의 프로젝트로 분리되기 때문에 [네임스페이스 목록](#)에서 더 이상 클러스터 프로젝트에서는 확인이 불가능합니다.

클러스터 프로젝트에서 네임스페이스 프로젝트로 분리되어 추가된 Namespace는 다음과 같이 현황 정보를 표시합니다.

Kubernetes Monitoring Demo(New) ▾ 네임스페이스 관리
🔔 🗄️ ⋮ 👤

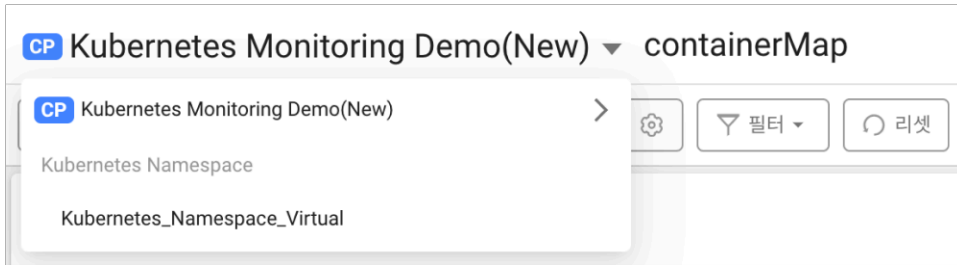
CP Kubernetes Monitoring Demo(New)

컨테이너 38
Namespace 5
Pod 34
Deployment 0
ReplicaSet 0
에이전트 노드 4
노드 CPU 코어 8
노드 메모리 18.8GiB
애플리케이션 10

네임스페이스 목록 🔄

이름	그룹	컨테이너	Namespace	Pod	Deployment	ReplicaSet	에이전트 노드	노드 CPU 코어	노드 메모리	애플리케이션	설정
Kubernetes_Namespace_Virtual	Demo Project	4	1	2	2	2	1	2	3.7GiB	4	⊙
default	Demo Project	2	1	3	2	2	1	2	1.2GiB	3	⊙
kube-flannel		+ 프로젝트 추가									
kube-node-lease		+ 프로젝트 추가									
virtual-main		+ 프로젝트 추가									

클러스터 프로젝트 하위에 네임스페이스 프로젝트를 생성한 경우 클러스터 프로젝트를 포함한 프로젝트 목록 확인 및 프로젝트 간 이동이 가능합니다. [대시보드](#) > [컨테이너 맵](#) 메뉴 화면 위에 프로젝트 목록 박스를 선택하세요.



설치 문제 해결

Helm 차트 기준 설치 문제 해결

이름이 `whatap` 다른 Helm 리포지토리가 있을 경우

"whatap" already exists with the same configuration, skipping

이미 동일한 이름의 다른 `whatap` 리포지토리가 존재할 경우 다음 명령어를 통해 기존 `whatap` 리포지토리 삭제 후 설치를 진행하세요.

```
helm repo remove whatap
```

이름이 `whatap-monitoring` 인 다른 네임스페이스가 클러스터에 있을 경우

Error: INSTALLATION FAILED: Unable to continue with install: Namespace "whatap-monitoring" in namespace "" exists and cannot be imported into the current release

클러스터에 `whatap-monitoring` 네임스페이스가 존재한다면 사용자가 `yaml` 파일 기준으로 이미 와탭 쿠버네티스 에이전트를 설치했기 때문입니다. 이 경우 다음 명령어를 통해 기존 `whatap-monitoring` 네임스페이스 및 관련 리소스를 삭제하세요.

```
kubectl delete namespace whatap-monitoring
kubectl delete clusterrolebinding whatap
kubectl delete clusterrole whatap
```

`clusterrole` , `clusterrolebinding` 리소스가 이미 클러스터에 있을 경우

Error: INSTALLATION FAILED: Unable to continue with install: ClusterRole "whatap" in namespace "" exists and cannot be imported into the current release

클러스터에 `clusterrole`, `clusterrolebinding` 리소스가 이미 존재할 경우 다음 명령어를 통해 해당 리소스를 삭제하세요.

```
kubectl delete clusterrolebinding whatap  
kubectl delete clusterrole whatap
```


쿠버네티스 에이전트 옵션 설정

- 와탭 쿠버네티스 에이전트 배포 시 컨테이너 `env` 필드를 통해 설정할 수 있습니다.
- 컨테이너 내부 `whatap.conf` 파일을 수정해 설정할 수 있습니다.
- `whatap.conf` 파일 내 옵션이 컨테이너 `env` 필드에 설정된 옵션보다 높은 우선순위를 가집니다.
- `whatap.conf` 또는 컨테이너 `env`에 설정된 값이 없는 경우 기본값이 사용됩니다.

⚠ 컨테이너 내 애플리케이션 에이전트 옵션 설정

컨테이너 내 Java, Python, Node.js 등 애플리케이션의 모니터링을 위해 필요 설정을 `whatap.conf` 파일에 작성합니다. `whatap.conf` 파일을 통해 설정할 수 있는 옵션에 대한 자세한 내용은 다음 링크를 참조하세요.

- [Java - whatap.conf 설정하기](#)
- [Node.js - whatap.conf 설정하기](#)
- [Python - whatap.conf 설정하기](#)

컨테이너별 옵션 설정

whatap-node-helper

- `collect_volume_detail_enabled` **bool**

기본값 `true`

노드 에이전트에서 모든 컨테이너에 대한 `exec df` 명령어를 이용해 컨테이너 볼륨 정보 조회를 활성화합니다. 노드에 실행 중인 모든 컨테이너의 스토리지 사용량 및 볼륨 정보를 수집하는 기능을 활성화합니다.

- `inject_container_id_to_apm_agent_enabled` **bool**

기본값 `true`

와탭 APM 컨테이너에 `exec` 를 이용한 container ID 주입을 활성화합니다. 와탭 APM으로 식별되는 컨테이너의 ID를 자동으로 추출합니다. 이를 APM 도구에 주입해 성능 모니터링 데이터와 컨테이너를 연결할 수 있습니다.

- `debug` **bool**

기본값 `false`

디버그 모드의 활성화 여부를 설정합니다. 디버그 모드 활성화 시 로그 레벨을 `DEBUG` 로, 그렇지 않을 경우 `INFO` 로 설정합니다.

- **kube_config_path** `string`

기본값 `~/.kube/config`

Kubernetes 설정 파일 경로를 지정합니다. 미설정 시 `~/.kube/config` 를 사용합니다.

- **log_sys_out** `bool`

기본값 `true`

로그 시스템 출력을 설정합니다. 옵션 값을 `true` 로 설정 시 표준 출력(stdOut)과 파일에 동시에 로깅을 남깁니다. 변경 시 로깅 시스템에 즉시 반영됩니다.

whatap-node-agent

- **shutdown** `bool`

기본값 `false`

수집 서버와 연결을 강제로 중단합니다. 통신 중지를 의미합니다.

- **license** `string`

프로젝트에 발급된 라이선스를 사용해 에이전트를 식별합니다.

- **whatap.server.host** `string`

기본값 `13.124.11.223` 혹은 `13.209.172.35`

수집 서버의 IP를 설정합니다. `delimiter` 의 경우 `/:` 을 사용합니다.

- **whatap.server.port** `number`

기본값 `6600`

수집 서버의 데이터 수신 Port를 설정합니다.

- **ONODE_NAME** `string`

기본값 `NODE_NAME` or `NODE_IP` 환경 변수값

에이전트 노드 식별을 위한 이름을 설정합니다.

- **tcp_connection_timeout** `number`

기본값 3000

whatap-master-agent와 수집 서버 소켓 연결 시 타임아웃 시간을 설정합니다.

- **tcp_so_timeout** number

기본값 60000

소켓 연결 후 데이터 송수신 타임아웃 시간을 설정합니다.

- **debug** bool

기본값 false

디버그 모드의 활성화 여부를 설정합니다.

- **gpuEnabled** bool

기본값 false

GPU 모니터링의 활성화 여부를 설정합니다.

- **count_interval** number

기본값 5000

메트릭스(`kube_pod` , `container`)를 수집 서버에 전송하는 주기를 설정합니다. 기본값의 단위는 ms입니다.

ⓘ 해당 옵션은 화면상 데이터 갱신 주기에 영향을 줄 수 있습니다. 자세한 내용은 다음 [메트릭스별 영향을 받는 화면 구성요소](#)를 참고하세요.

- **collect_interval** number

기본값 5000

에이전트가 apiserver에 요청해 메트릭스(`kube_pod` , `container`)를 수집하는 주기를 설정합니다.

ⓘ 해당 옵션은 화면상 데이터 갱신 주기에 영향을 줄 수 있습니다. 자세한 내용은 다음 [메트릭스별 영향을 받는 화면 구성요소](#)를 참고하세요.

- **collect_apiserver_metric_by_leader** bool

기본값 false

해당 옵션이 활성화 된 경우 노드 에이전트는 메트릭 수집을 마스터 에이전트에 위임합니다. whatap-master-agent의 `collect_apiserver_metric_by_leader` 옵션 활성화 선행이 필요합니다. 메트릭 수집 위임을 통해 기존 whatap-node-agent가 발생시킨 apiserver 부하를 낮출 수 있습니다.

whatap-control-plane-helper

- **port** *string*

기본값 `9496`

whatap-control-plane-helper의 웹 서버 포트를 설정합니다.

- **cycle** *number*

기본값 `5`

control-plane-helper가 지표를 스크래핑하는 주기를 설정합니다. 기본값의 단위는 s입니다.

- **collect_control_plane_monitoring_enabled** *bool*

기본값 `false`

control-plane-monitoring 기능의 활성화 여부를 설정합니다.

- **kube_config_path** *string*

기본값 `~/.kube/config`

Kubernetes 설정 파일 경로를 지정합니다. 미설정 시 `~/.kube/config` 를 사용합니다.

- **kube_master_url** *string*

기본값 `~/.kube/config`

스크래핑을 위해 요청할 control-plane 의 url을 설정합니다. 미설정 시 `~/.kube/config` 를 사용합니다.

- **debug** *bool*

기본값 `false`

whatap-control-plane-helper의 디버깅(로깅)의 활성화 여부를 설정합니다.

whatap-master-agent

- **shutdown** *bool*

기본값 `false`

수집 서버와 연결을 강제로 중단합니다. 통신 중지를 의미합니다.

- **license** *string*

프로젝트에 발급된 라이선스를 사용해 에이전트를 식별합니다.

- **whatap.server.host** *string*

기본값 `13.124.11.223` 혹은 `13.209.172.35`

수집 서버의 IP를 설정합니다. `delimiter`의 경우 `/:`을 사용합니다.

- **whatap.server.port** *number*

기본값 `6600`

수집 서버의 데이터 수신 Port를 설정합니다.

- **tcp_connection_timeout** *number*

기본값 `3000`

whatap-master-agent와 수집 서버 소켓 연결 시 타임아웃 시간을 설정합니다.

- **tcp_so_timeout** *number*

기본값 `60000`

whatap-master-agent와 수집 서버 소켓 통신 시 타임아웃 시간을 설정합니다.

- **debug** *bool*

기본값 `false`

whatap-master-agent에서 `DEBUG` 레벨 로깅 활성화 여부를 설정합니다.

- **collect_apiserver_metric_by_leader** *bool*

기본값 `false`

whatap-master-agent가 스크래퍼의 리더로 동작 활성화 여부를 설정합니다. 활성화 시 whatap-master-agent 단독으로 apiserver에 메트릭을 요청합니다.

- **collect_control_plane_monitoring_enabled** *bool*

기본값 `false`

컨트롤 플레인 모니터링 기능의 활성화 여부를 설정합니다.

- **master_to_control_plane_connection_timeout** *number*

기본값 `5000`

whatap-master-agent가 whatap-control-plane-helper와 통신 시 connection timeout이 발생할 때까지의 시간을 설정합니다. 기본값의 단위는 ms입니다.

- **master_to_control_plane_read_timeout** `number`

기본값 `20000`

whatap-master-agent가 whatap-control-plane-helper와 통신 시 read timeout이 발생할 때까지의 시간을 설정합니다. 기본값의 단위는 ms입니다.

- **control_plane_debug** `bool`

기본값 `false`

컨트롤 플레인 모니터링에 대한 whatap-master-agent에서의 모든 디버그 로그 기능의 활성화 여부를 설정합니다.

- **store_k8s_object_manifest_enabled** `bool`

기본값 `true`

오브젝트 매니페스트 수집 기능의 활성화 여부를 설정합니다. 활성화 시 하루에 한번 클러스터 오브젝트들의 매니페스트 정보를 수집합니다.

- **store_k8s_object_manifest_timezone** `string`

기본값 `UTC`

오브젝트 매니페스트 수집을 위한 타임존을 설정합니다. 별도 설정이 없을 경우 기본값 UTC 기준 자정에 수집을 시작합니다.

예, 한국 시간 기준으로 자정에 실행하고 싶을 경우 `GMT+9` 혹은 `Asia/Seoul` 과 같이 타임존 설정

- **store_k8s_object_manifest_max_fetch_limit** `number`

기본값 `50`

한 번에 수집할 오브젝트의 조회 개수를 설정합니다. 별도 설정이 없을 경우 기본값 `50` 기준으로 1회 요청당 50개씩 요청합니다.

예, 전체 Pod 개수가 90개인 클러스터에서 해당 옵션 `30` 설정 시 Pod 정보를 30개씩 3회 요청

메트릭스별 영향을 받는 화면 구성요소

`kube_pod`

- [대시보드 > 컨테이너 맵](#)

다음 예시 이미지의 Pod 뷰 기준 데이터는 `kube_pod` 메트릭스 수집 주기 및 수집 서버 전송 주기에 영향을 받습니다.



• 워크로드 > Pod 목록

다음 예시의 데이터는 kube_pod 메트릭스 수집 주기 및 수집 서버 전송 주기에 영향을 받습니다.

	Pod ↓	PodIp	Namespace	Node	ConRestartCnt	Age	CpuByLimit	CpuLimitByNode
✓	whatap-virtual-...	2406.da12.817:106.f...	virtual-main	ip-10-21-13-168.ap-n...	0	27d 02h 54m 54s	4.69%	15.00%
✓	whatap-virtual-...	2406.da12.817:106.f...	virtual-main	ip-10-21-13-168.ap-n...	0	7d 19h 36m 52s	0%	15.00%
✓	whatap-virtual-...	2406.da12.817:106.f...	virtual-main	ip-10-21-13-168.ap-n...	0	22d 06h 13m 11s	3.29%	15.00%

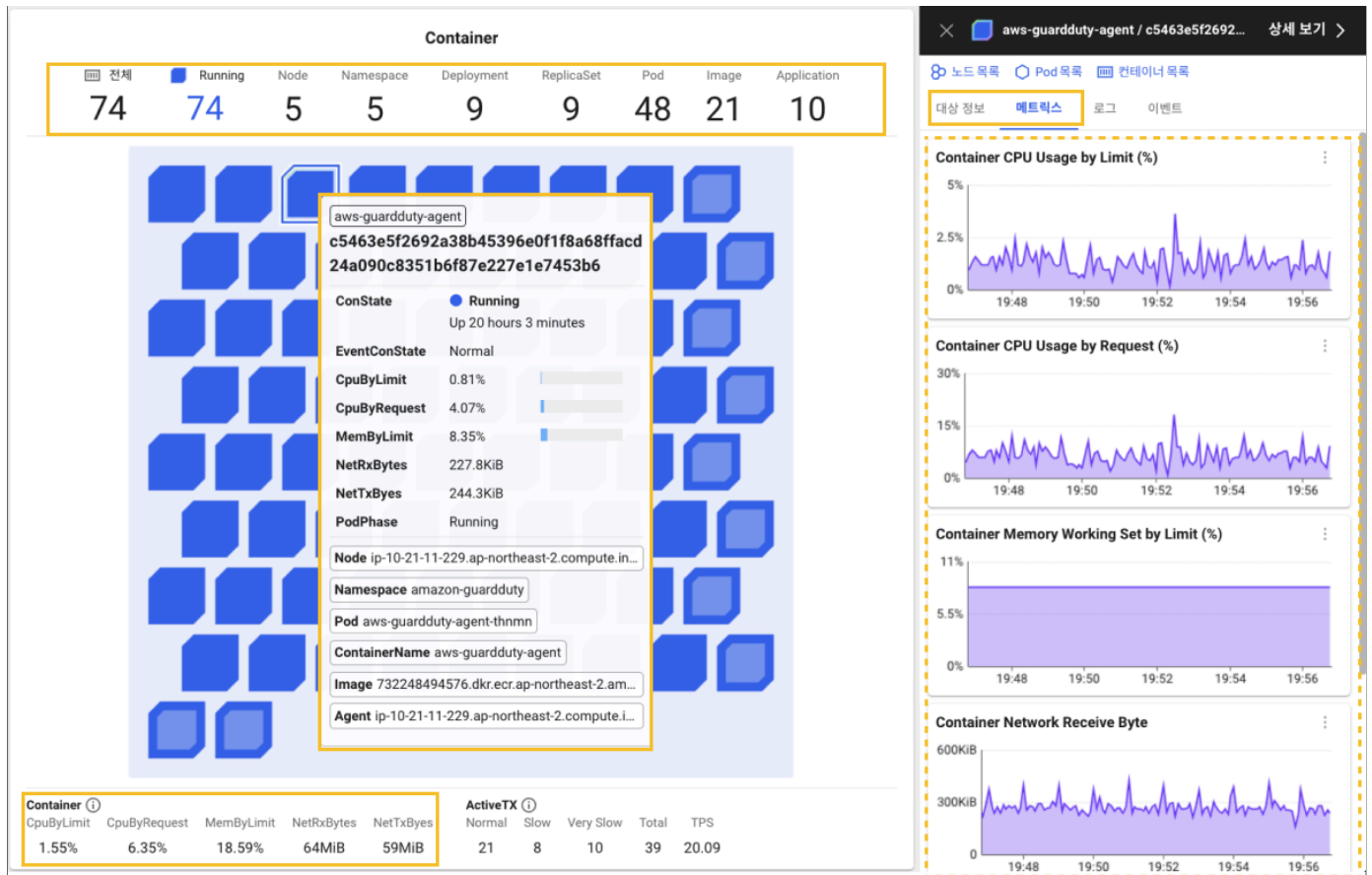
• 대시보드 > 퍼포먼스 요약

kube_pod 메트릭스를 이용하는 에이전트 설치 노드에서 실행 중인 Pod 수 차트 갱신 주기에 영향을 받습니다.

container

• 대시보드 > 컨테이너 맵

다음 예시의 컨테이너 뷰 기준 화면 데이터 갱신 주기는 `container` 메트릭스의 영향을 받습니다.



• 컨테이너 > 컨테이너 이미지

다음 예시의 데이터는 `container` 메트릭스의 영향을 받습니다.

Registry	Name	Tag	Repo Digest	Containers	Total Container CPU	Total Container Memory
732248494576.dkr.ecr.ap-nor...	aws-guardduty-agent	v1.5.0	sha256:0393d044ad76b911d...	5	544m	605.55MiB
602401143452.dkr.ecr.ap-nor...	eks/aws-ebs-csi-driver	v1.24.0	sha256:3dba81d0c3c8ce51a...	7	0m	81.15MiB
602401143452.dkr.ecr.ap-nor...	eks/livenessprobe	v2.10.0-eks-1-28-6	sha256:2fbd620d25bda985b...	7	4m	48.63MiB
602401143452.dkr.ecr.ap-nor...	eks/csi-node-driver-regi...	v2.9.0-eks-1-28-6	sha256:16ebda509ca6f641...	5	0m	29.07MiB

• 대시보드 > 퍼포먼스 요약

`container` 메트릭스를 이용하는 다음 차트의 갱신 주기에 영향을 받습니다.

- 에이전트 설치 노드에서 실행 중인 컨테이너 수
- 에이전트 설치 노드 수

- Total Container CPU Request
 - Total Container CPU Usage
 - Node CPU
 - Node Memory
 - Container CPU Usage by Request
 - Container CPU Usage by Node
 - Container Memory Working Set by Request
 - Container Memory Failure Count
 - Container CPU Throttling Time
 - Container Network Receive Byte
 - Container Network Transmit Byte
- [클러스터 > 노드 상세](#)

다음 예시의 [노드 상세](#) 화면에서 [컨테이너](#) 목록의 갱신 주기는 `container` 메트릭스의 영향을 받습니다.

Kubernetes Monitoring Demo(EKS) - 노드 상세

시간 (5초 인터벌) | 서버 선택 | ip-10-21-11-229.ap-northeast-2.compute.internal | 운영체제: Linux | CPU 코어: 2 | Memory Total: 1.9 GB

CPU Usage

26%

CPU System | CPU User | CPU Nice | CPU Steal | CPU IOWait | CPU IRQ | CPU Soft IRQ

Memory Usage

63.55%

Memory Used | Memory Swap

Network

886.25Kib

eth0 | eni8c4a194eaa4 | enic855b2f6676 | enicd4d7192240 | enif792f40cbb0

Disk Usage

30.69%

/

컨테이너

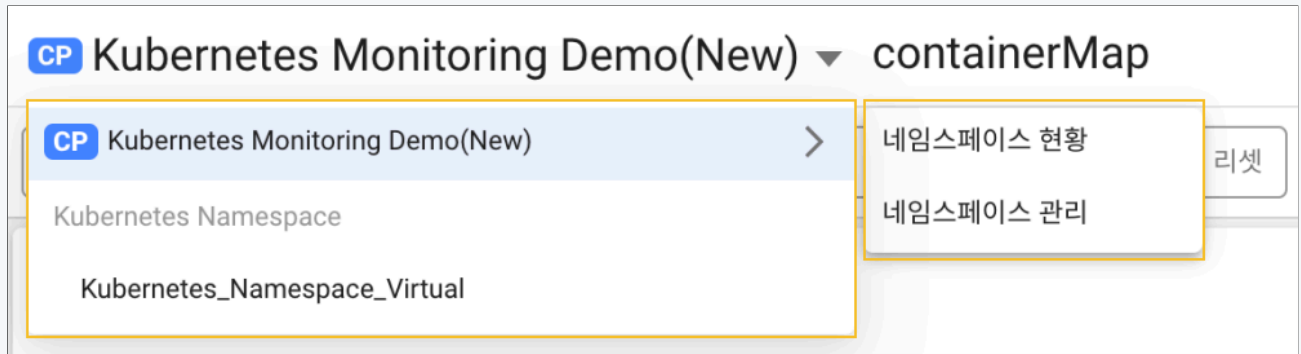
컨테이너 구성 정보						리소스 사용량		CPU		MemLimit	MemUsed
상태	컨테이너	이미지	커맨드	애플리케이션	Pod	CpuByLimit	MemWsByLimit	CpuThrottledCnt	CpuThrottledTime	MemLimit	MemUsed
✓ RUNNING	c5463e5f2692a38b453...	73224849...			aws-guardduty-agent-th...	5.31 %	9.06 %	0	0	1.00 GB	25%
✓ RUNNING	0e8191d40498705428...	60240114...			aws-node-64mtc	0.19 %	1.5 %	0	0	1.88 GB	0%
✓ RUNNING	8fe70e0faf20fc83665f...	60240114...			coredns-6d47776b78-z...	0 %	8.86 %	0	0	170.00 MB	70%
✓ RUNNING	0f62937dfada35f45d5...	60240114...			ebs-csi-node-j48qm	0 %	4.76 %	0	0	256.00 MB	40%
✓ RUNNING	f58257413cfd8827736...	60240114...			ebs-csi-node-j48qm	0 %	2.82 %	0	0	256.00 MB	40%
✓ RUNNING	af7ed81220c864a34a9...	60240114...			ebs-csi-node-j48qm	0 %	2.86 %	0	0	256.00 MB	40%
✓ RUNNING	534ba67f3cd854c0ffe3...	60240114...	kube-prox...		kube-proxy-t8zw2	0 %	0.94 %	0	0	1.88 GB	0%
✓ RUNNING	3e76bb6e0d1531f3da2...	docker.io/...		virtual	whatap-virtual-deploy...	39.88 %	69.53 %	0	0	256.00 MB	25%
✓ RUNNING	34c6958eb634601681...	docker.io/...	/bin/entry...		whatap-node-agent-p2bjl	1.93 %	72.28 %	0	0	350.00 MB	30%

네임스페이스 관리

홈 화면 > 프로젝트 선택 > 관리 > 네임스페이스 관리

클러스터 프로젝트에서 **네임스페이스 관리** 메뉴를 통해 네임스페이스 프로젝트를 관리할 수 있습니다. 네임스페이스 프로젝트에서는 해당 메뉴로 진입할 수 없습니다.

① 클러스터 프로젝트의 **대시보드** > **컨테이너 맵** 메뉴 화면 위에는 하위 네임스페이스 또는 **네임스페이스 현황** 및 **네임스페이스 관리** 메뉴로 이동할 수 있는 리스트 박스를 제공합니다.



네임스페이스 프로젝트 관리

Kubernetes Monitoring Demo(New) > 네임스페이스 관리

1

Kubernetes Monitoring Demo(New)

컨테이너: 38 | Namespace: 5 | Pod: 34 | Deployment: 0 | ReplicaSet: 0 | 에이전트 노드: 4 | 노드 CPU 코어: 8 | 노드 메모리: 18.8GiB | 애플리케이션: 10

네임스페이스 목록

2


이름	그룹	컨테이너	Namespace	Pod	Deployment	ReplicaSet	에이전트 노드	노드 CPU 코어	노드 메모리	애플리케이션	설정
Kubernetes_Namespace_Virtual	Demo Project	4	1	2	2	2	1	2	3.7GiB	4	⊙
default	Demo Project	2	1	3	2	2	1	2	1.2GiB	3	⊙
kube-flannel											+ 프로젝트 추가
kube-node-lease											+ 프로젝트 추가
virtual-main											+ 프로젝트 추가


상단의 ① 영역에서 대상 전체 개수 및 주요 리소스 개수를 확인할 수 있습니다. ② **네임스페이스 목록** 영역에서 네임스페이스 프로젝트를 조회하고 추가 및 제거할 수 있습니다.


네임스페이스 프로젝트 생성

네임스페이스 목록에서 프로젝트를 생성할 항목의 [+ 프로젝트 추가](#) 버튼을 선택하세요. [프로젝트 추가](#) 창이 나타나면 [확인](#) 버튼을 선택하세요.

네임스페이스 프로젝트 삭제

1. 네임스페이스 목록에서 삭제하려는 프로젝트 항목의 가장 오른쪽에  [설정](#) 아이콘을 선택하세요.
2. [프로젝트 관리](#) 메뉴로 이동하세요.
3. 화면 오른쪽 위에 [프로젝트 삭제](#) 버튼을 선택하세요.

 네임스페이스 프로젝트를 삭제할 경우 사용자 목록 정보와 그동안 수집한 데이터는 모두 삭제합니다. 삭제한 데이터는 다시 복구할 수 없기 때문에 주의하세요.

 프로젝트 관리에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

연계 프로젝트 관리

본 문서는 [연계 프로젝트 관리](#) 메뉴를 안내합니다. 와탭은 프로젝트간 데이터 연계를 바탕으로 통합 가시성을 제공합니다. 태그 정보를 통해 연계 프로젝트를 검색하고, 해당 프로젝트에서 문제의 원인이 될 수 있는 주요 지표를 빠르게 확인할 수 있습니다.

- 연계 프로젝트를 조회 및 추가, 삭제 등 관리 기능을 제공합니다.
- 추가한 프로젝트간 연계 분석이 가능합니다.

기본 화면 안내

홈 화면 > 프로젝트 선택 > 관리 > [연계 프로젝트 관리](#)

연계 프로젝트 추가

연계 추가할 프로젝트를 선택한 후, 추가 버튼을 눌러주세요. (지원 타입: Server, MySQL, PostgreSQL)

추가

연계 프로젝트 목록 🔄

프로젝트 코드	프로젝트 이름	그룹	타입	linkageType(Dev Only)	
20	DB-MARIADB-JINS-TEST	Jinslab	MySQL	DB_MYSQL	
1871	-	-	-	DB_POSTGRESQL	
2116	koogle-mysql	kube-dev-group	MySQL	DB_MYSQL	
2125	koogle-front-server	kube-dev-group	Server	SMS_INFRA, SMS_INFRA_IP, SMS_INFRA_TOP5	

다음은 [연계 프로젝트 관리](#) 메뉴의 기본 화면 구성을 안내합니다. 상단의 [연계 프로젝트 추가](#)와 하단의 [연계 프로젝트 목록](#) 화면으로 구성되어있습니다.

연계 프로젝트 추가

연계 프로젝트를 추가할 수 있습니다. [프로젝트 선택](#) 창 클릭 시 다음과 같이 연계 등록할 수 있는 프로젝트를 조회할 수 있습니다.

연계 프로젝트 추가
 연계 추가할 프로젝트를 선택한 후, 추가 버튼을 눌러주세요. (지원 타입: Server, MySQL, PostgreSQL)


DB-POSTGRESQL-JINS-TEST PostgreSQL 60 Jinslab x LINK-SERVER-JINS-TEST Server 1410 Jinslab x 추가

타입	linkageType(Dev Only)
MySQL	DB_MYSQL
	DB_POSTGRESQL
MySQL	DB_MYSQL
Server	SMS_INFRA, SMS_INFRA_IP, SMS_INFRA_TOP5

- 프로젝트 선택 창에서 프로젝트 코드 및 프로젝트 이름으로 검색할 수 있습니다.
- 프로젝트 선택 시 추가 버튼이 활성화됩니다. 여러 개의 프로젝트를 선택할 수 있습니다.
- 프로젝트 선택 후 추가 버튼 클릭 시 안내 메시지와 함께 하단의 연계 프로젝트 목록에 해당 프로젝트가 추가된 것을 확인할 수 있습니다.

연계 프로젝트 삭제

추가한 프로젝트를 연계 대상에서 제외하고 싶은 경우 연계 프로젝트 목록 우측의  삭제 아이콘을 선택해 삭제하세요.

- ① 현재 Server, PostgreSQL, MySQL 제품을 지원합니다.
 - 추가한 연계 프로젝트가 목록에 표시되지 않을 경우 잠시 기다린 후  새로 고침 아이콘을 선택해 새로고침 하세요.
 - 삭제의 경우 프로젝트 수정 권한이 필요합니다.

연계 분석 활용 예시

히트맵을 드래그해 트레이스 분석 창으로 이동 > 트랜잭션 항목 선택 > 오른쪽 상세 분석 화면의 테이블 뷰 선택

다음은 연계 프로젝트를 기반으로 연계 분석 활용 예시를 안내합니다.

트레이스 분석 18건 조회되었습니다. ⓘ

전체 [필터] 컬럼 설정 [다운로드] TXT 검색어를 입력하세요

No	에이전트 명 (oname)	트랜잭션	경과 시간	시작 시간
1			2,575	24/01/09 17:57:24
2			2,306	24/01/09 17:57:24
3			2,298	24/01/09 17:57:24
4			2,275	24/01/09 17:57:24
5			2,260	24/01/09 17:57:24
6			2,260	24/01/09 17:57:24
7			2,257	24/01/09 17:57:24
8			2,250	24/01/09 17:57:24
9			2,243	24/01/09 17:57:24

레코드 요약 **a** 테이블 뷰 트리 뷰 메소드 요약 SQL 요약 HTTP Call 요약 트랜잭션 ...

시작: 01/09 17:57:41.024 종료: 01/09 17:57:43.267 경과: 2,243ms
에이전트 명 (oname): 클라이언트 IP: 0.0.0.0

500ms 1s 1.5s 2s

SQL

컬럼 선택 [다운로드] TXT [다운로드] SQL [다운로드] SQL

No	시간	값	경과	내용
1	17:57:41.024		시작	2024-01-09 17:57:41.024
2	17:57:41.024	0	0	MYSQL jdbc:mysql://3.34.216... select book0_id as 'id'

b 연계 DB 주요 지표 보기

c 2024/01/09 18:09 [10분] 선택한 시간의 마지막 시간을 기준으로 10분 간의 데이터를 조회합니다.

연계 프로젝트

[20] DB-MARIADB-JINS-TEST

[2116] koogle-mysql

MySQL

Active Sessions Threads Connected InnoDB Buffer Pool Read Requests InnoDB Buffer Pool Reads Questions Lock W ...

1. **애플리케이션 서비스 대시보드**의 **히트맵**을 드래그해 **트레이스 분석** 창으로 이동하세요.
2. 조회를 원하는 트랜잭션을 선택 후 오른쪽 상세 분석 화면에서 **a** **테이블 뷰**를 선택하세요.
3. 목록에서 **b** 영역의 **연계 아이콘** 선택 시 **c** 연계 프로젝트의 주요 지표를 확인할 수 있습니다. 연계 시나리오가 다수일 경우 드롭다운 방식으로 원하는 분석 시나리오 목록을 제공합니다.

ⓘ **조회 시간**

사용자가 조회한 마지막 시간을 기준으로 10분 간의 데이터를 표시합니다.
예, **히트맵**의 34분~36분 범위 데이터 드래그 시 26분~36분간의 데이터를 표시합니다.

쿠버네티스 에이전트

쿠버네티스 내에 실행 중인 애플리케이션의 모니터링을 위해 설치한 에이전트 구성에 대해 알아봅니다.

에이전트 구성

와탭 쿠버네티스 설치 시 기본적으로 사용자 쿠버네티스 환경의 <whatap-monitoring> Namespace 하위에 에이전트용 리소스를 구성합니다.

마스터 에이전트는 Deployment, 노드 에이전트는 DaemonSet 형태입니다. 다음은 사용자 쿠버네티스 환경에 설치된 와탭 에이전트를 확인하는 방법입니다.

• 마스터 에이전트

```
kubectl get deployments -n whatap-monitoring
kubectl get pods -l=name=whatap-master-agent -n whatap-monitoring
```

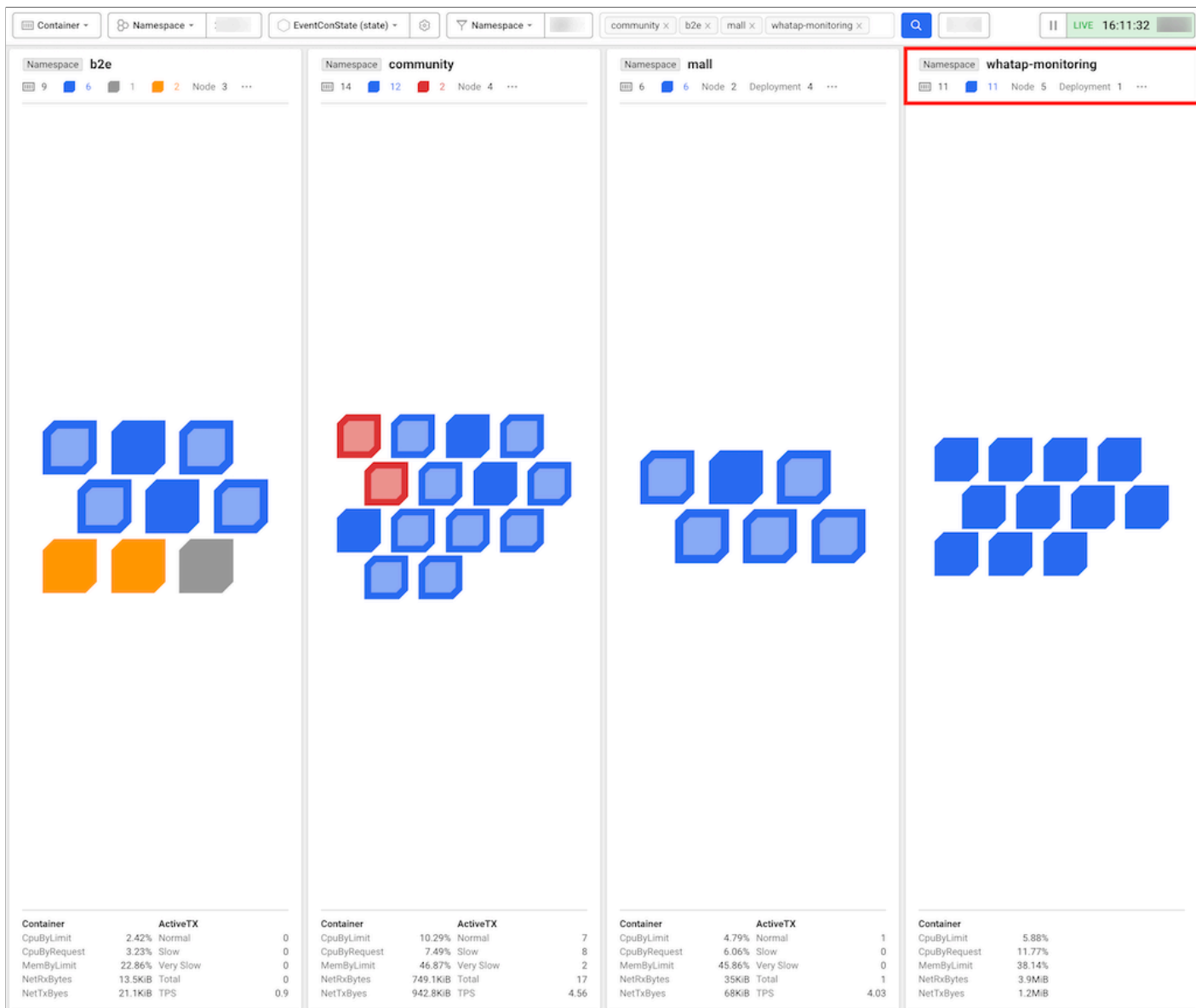
• 노드 에이전트

```
kubectl get daemonsets -n whatap-monitoring
kubectl get pods -l=name=whatap-node-agent -n whatap-monitoring
```

다음은 <whatap-monitoring> Namespace 내에서 실행 중인 마스터 에이전트, 노드 에이전트 형태의 예입니다.

```
$ kubectl get pods -n whatap-monitoring
NAME                                READY STATUS RESTARTS AGE
whatap-master-agent-57f4499567-ltbb 1/1   Running 0      3d19h
whatap-node-agent-5q8rj              2/2   Running 0      21h
whatap-node-agent-6nb2z              2/2   Running 0      30h
whatap-node-agent-8m2vk              2/2   Running 0      21h
whatap-node-agent-9thmw              2/2   Running 0      3h31m
whatap-node-agent-kpv7d              2/2   Running 0      21h
```

다음은 와탭 쿠버네티스 컨테이너 맵 상에 표시된 <whatap-monitoring> Namespace 입니다.



쿠버네티스 에이전트 목록

홈 화면 > 클러스터 프로젝트 선택 > 관리 > 쿠버네티스 에이전트 목록

클러스터 프로젝트 (CP)의 관리 메뉴 하위에서 쿠버네티스 에이전트 목록 메뉴로 이동할 수 있습니다.

Kubernetes Monitoring Demo(EKS) ▾ 쿠버네티스 에이전트 목록 🔊 🔔 🗑️ ⋮ 👤

🔄 11:39:06

마스터 에이전트 설치 현황 (현재 마스터 에이전트 CPU Limit: 200m, 노드 에이전트 수: 6) 📄 컬럼 선택 📄 다운로드

에이전트 정보				노드 정보			
State	Name	Version	OnodeName	Age	CapacityCpu	CapacityMemory	
ACTIVE	KMaster	1.5.6 20240221	ip-10-21-11-147.ap-n...	8d 00h 25m 55s	2	7.61 GiB	

노드 에이전트 설치 현황 전체 6 ACTIVE 6 Node Name, Agent Name 🔍 📄 컬럼 선택 📄 다운로드


에이전트 정보				노드 정보				프로젝트 리소스	
State	Name	Version	OnodeName	Age	CapacityCpu	CapacityMemory	PodsCount	Contain	
ACTIVE	ip-10-21-11-147.a...	1.5.6 20240221	ip-10-21-11-14...	8d 00h 25m 55s	2	7.61 GiB	20		
ACTIVE	ip-10-21-13-19.ap...	1.5.6 20240221	ip-10-21-13-19...	8d 00h 25m 40s	2	1.88 GiB	8		
ACTIVE	ip-10-21-11-47.ap...	1.5.6 20240221	ip-10-21-11-47...	8d 00h 24m 44s	2	1.88 GiB	6		
ACTIVE	ip-10-21-13-126.a...	1.5.6 20240221	ip-10-21-13-12...	8d 00h 25m 40s	2	1.88 GiB	7		
ACTIVE	ip-10-21-11-155.a...	1.5.6 20240221	ip-10-21-11-15...	7d 22h 59m 20s	2	1.88 GiB	5		
ACTIVE	ip-10-21-13-25.ap...	1.5.6 20240221	ip-10-21-13-25...	6d 04h 46m 57s	2	1.88 GiB	7		

- 쿠버네티스 에이전트의 설치 현황을 확인할 수 있습니다. 일부 에이전트가 미설치(**UNINSTALLED**) 혹은 비활성화(**INACTIVE**) 상태일 경우 데이터 수집에 왜곡이 생길 수 있습니다.
 - 예, State 컬럼의 **UNINSTALLED**, **ACTIVE**, **INACTIVE**
- 마스터 에이전트에 할당된 CPU Limit과 설치된 노드 에이전트 수를 확인할 수 있습니다.
 - 예, 노드 에이전트 수 50개 이상일 경우 마스터 에이전트 CPU Limit 500 밀리코어 이상으로 설정 권장
- 에이전트가 설치된 노드에 할당된 CPU 및 메모리 정보를 확인할 수 있습니다.
 - 예, 노드 정보 컬럼 하위의 CapacityCpu, CapacityMemory
- 에이전트가 설치되지 않은 노드들을 확인할 수 있습니다.
- 에이전트의 주요 정보를 확인할 수 있습니다.
- 노드 에이전트 버전과 관련된 안내를 확인할 수 있습니다.
 - 예, 노드간 버전 불일치 시 상단에 버전 불일치 안내

ⓘ 마스터 에이전트 자원 할당

사용자의 쿠버네티스 환경 규모 즉 노드의 수에 따라 와탭 에이전트가 수집 및 처리하는 데이터의 양이 달라지기 때문에 마스터 에이전트에 충분한 자원 할당이 필요합니다. 예를 들어 [노드 에이전트](#) 수가 많을 경우 마스터 에이전트 [CPU Limit](#) 상향을 권장합니다.

새로고침


상단의  [새로고침](#) 아이콘 선택 시 현재 시간을 기준으로 데이터를 갱신할 수 있습니다.

컬럼 선택



[컬럼 선택](#) 옵션 활용 시 원하는 에이전트 정보를 선택해 에이전트 목록을 조회할 수 있습니다.

- 선택한 컬럼 목록은 메뉴 재진입 시 유지됩니다.
- [초기화](#) 버튼 선택 시 사전 정의된 기본 주요 지표 컬럼이 선택됩니다. 이 경우 기존 컬럼 선택 내역은 삭제됩니다.
- 컬럼 선택 후 우측 하단 [적용](#) 버튼을 클릭 해 선택 내역을 적용할 수 있습니다.

다운로드

 [다운로드](#) 아이콘 선택 시 [범위](#) 및 [표시 옵션](#)을 설정해 해당 데이터를 CSV 파일로 다운로드할 수 있습니다.

상세 보기

- [에이전트 정보](#)의 [Name](#) 컬럼 하위의  [상세 보기](#) 아이콘 선택 시 해당 에이전트의 [에이전트 로그](#)를 확인할 수 있습니다.
- [OnodeName](#) 컬럼 하위의  [노드 상세 보기](#) 아이콘 선택 시 해당 노드 상세 페이지로 이동합니다.

ⓘ [노드 상세](#)와 관련한 자세한 내용은 [다음 문서](#)를 참조하세요.

에이전트 업데이트

- ⓘ • EKS Fargate는 지원 예정입니다.
- 와탭 쿠버네티스 에이전트의 최신 버전은 hub.docker.com에서 확인할 수 있습니다.

에이전트 업데이트가 필요한 경우 다음 명령을 실행하세요.

• 마스터 에이전트

```
kubectl set image -n whatap-monitoring deployment/whatap-master-agent whatap-master-agent=whatap/
kube_mon:{agent_version}
```

• 노드 에이전트

```
kubectl set image -n whatap-monitoring daemonset/whatap-node-agent whatap-node-agent=whatap/
kube_mon:{agent_version}
```

• Helm 차트 기준 설치 시 에이전트 애플리케이션 업데이트

```
# release 업그레이드
helm upgrade whatap-kube-agent whatap/kube -f values.yaml
```

에이전트 삭제

쿠버네티스 에이전트 삭제

에이전트 설치 단계에서 다운로드한 쿠버네티스 버전과 Container Runtime에 맞는 [YAML](#) 파일명을 확인하고 절차에 따라 쿠버네티스 에이전트 삭제 과정을 진행하세요.

쿠버네티스 에이전트 YAML 파일명

Docker	YAML 파일명
쿠버네티스 버전 1.16 이상	whatap_kube_1.16.yaml
쿠버네티스 버전 1.16 이상(GPU 버전)	whatap_kube_1.16_gpu.yaml
쿠버네티스 버전 1.16 미만	whatap_kube.yaml
쿠버네티스 버전 1.16 미만(GPU 버전)	whatap_kube_gpu.yaml

Containerd	YAML 파일명
쿠버네티스 버전 1.16 이상	whatap_kube_containerd_1.16.yaml
쿠버네티스 버전 1.16 이상(GPU 버전)	whatap_kube_containerd_1.16_gpu.yaml
쿠버네티스 버전 1.16 미만	whatap_kube_containerd.yaml
쿠버네티스 버전 1.16 미만(GPU 버전)	whatap_kube_containerd_gpu.yaml

CRIO YAML	파일명
쿠버네티스 버전 1.16 이상	whatap_kube_crio_1.16.yaml
쿠버네티스 버전 1.16 이상(GPU 버전)	whatap_kube_crio_1.16_gpu.yaml

CRIO YAML	파일명
쿠버네티스 버전 1.16 미만	<code>whatap_kube_crio.yaml</code>
쿠버네티스 버전 1.16 미만(GPU 버전)	<code>whatap_kube_crio_gpu.yaml</code>

YAML 파일이 있을 경우


1. 설치 단계에서 다운로드한 **YAML** 파일명으로 설치된 쿠버네티스 에이전트를 삭제합니다. 다음 명령어의 표기 부분을 사용자의 **YAML** 파일명으로 수정 후 실행하세요.

```
kubectl delete -f <사용자의 YAML 파일명 >
```

2. 쿠버네티스 에이전트 삭제가 정상적으로 완료되었을 경우 다음과 같은 메시지를 확인할 수 있습니다.

```
namespace "whatap-monitoring" deleted
serviceaccount "whatap" deleted
clusterrole.rbac.authorization.k8s.io "whatap" deleted
configmap "node-start-script" deleted
clusterrolebinding.rbac.authorization.k8s.io "whatap" deleted
daemonset.apps "whatap-node-agent" deleted
configmap "master-start-script" deleted
deployment.apps "whatap-master-agent" deleted
service "whatap-master-agent" deleted
```

YAML 파일이 없을 경우

 whatap-monitoring 네임스페이스에 직접 사용자 정의 리소스를 생성한 경우 해당 리소스도 함께 삭제됩니다.

1. 다음 명령어를 실행해 쿠버네티스 에이전트를 삭제하세요.

```
kubectl delete namespace whatap-monitoring
kubectl delete clusterrolebinding whatap
kubectl delete clusterrole whatap
```

2. 쿠버네티스 에이전트 삭제가 정상적으로 완료되었을 경우 다음과 같은 메시지를 확인할 수 있습니다.

```
namespace "whatap-monitoring" deleted  
clusterrolebinding.rbac.authorization.k8s.io "whatap" deleted  
clusterrole.rbac.authorization.k8s.io "whatap" deleted
```

Helm 차트 기준 설치 시 에이전트 삭제

다음 명령어를 실행해 와탭 쿠버네티스 에이전트 애플리케이션을 삭제하세요.

```
# release 삭제  
helm uninstall whatap-kube-agent
```

주요 메뉴 알아보기

와탭 에이전트 설치를 완료했다면 [와탭 모니터링 서비스](#)에 로그인하세요. 와탭 모니터링 서비스는 원활한 서비스 제공과 안정적 운영, 이상 징후 감시 및 파악을 위한 정보를 제공합니다.

쿠버네티스에는 마이크로 서비스를 효율적으로 운영할 수 있는 다양한 기능이 포함되어 있습니다. 컴퓨팅 리소스를 효율적으로 사용하기 위한 스케줄링 기능과 애플리케이션 업데이트를 위한 배포 기능, 컨테이너 실행에 필요한 스토리지와 네트워크 관리 등 서비스 운영에 필요한 모든 기능을 제공합니다.

모니터링 서비스는 사용자가 쿠버네티스를 운영하는 모든 과정에 도움을 주고자 합니다. 실시간으로 문제 상황을 감시하여 장애 상황을 알리고 문제를 해결에 도움을 주는 모니터링의 기본적인 기능 뿐만 아니라, 자원을 분배하고 인프라 비용을 절약하고자 하는 운영 업무 시, 또는 쿠버네티스로 통합된 여러 마이크로 서비스와 이와 관련된 업무 부서 간 협업 도구로서 모니터링 서비스를 활용할 수 있습니다.

쿠버네티스의 활용성이 다양한만큼 운영 환경과 비즈니스 상황에 따라 사용자의 모니터링 니즈도 다양합니다. 이를 위해 와탭의 모니터링 서비스는 애플리케이션 트랜잭션과 리소스 사용량, 로그와 이벤트 등 쿠버네티스가 관리하거나 생산하는 모든 정보를 수집하고 통합적으로 관제할 수 있는 기능을 제공합니다.

실시간 모니터링

- 멀티 클라우드, 하이브리드 클라우드 상의 쿠버네티스 통합 모니터링
- 컨테이너 맵 - 노드(호스트), 컨테이너, 애플리케이션, 로그, 이벤트 정보를 제공하는 동적 대시보드
 - Namespace, Deployment, ReplicaSet, Pod 등의 쿠버네티스 구성요소를 개별 / 그룹화하여 표시
 - 개별 대상 / 사용자 지정 그룹 별 정보 실시간 확인
 - 컨테이너 상태 및 핵심 리소스 지표별 실시간 상태를 3단계 색상으로 구분 (Normal, Warning, Critical)
 - 컨테이너에 할당된 CPU Limit, Request 및 실 사용량 (core)
 - 컨테이너에 할당된 Memory Limit, Request 및 실 사용량 (MiB, GiB)
 - 컨테이너가 사용 중인 Disk I/O, Network I/O 정보
 - 히트맵 기반 애플리케이션 모니터링
 - Pod / 컨테이너 내에서 발생 중인 Stdout/Stderr 정보
 - 컨테이너 / 애플리케이션 로그 기간별 검색
 - 컨테이너 / 애플리케이션에 사용자가 지정한 경고
 - 쿠버네티스 클러스터 내 이벤트 정보

- 리소스 모니터링 - 컨테이너 및 노드 상세 모니터링
 - 컨테이너
 - 전체 컨테이너 목록 및 사용률 현황
 - 개별 컨테이너의 CPU, Memory, Disk(Volume), Network 사용률
 - 컨테이너에 연결된 볼륨 정보
 - 노드
 - 전체 노드 목록 및 사용률 현황
 - 개별 노드(호스트)의 CPU, Memory, Disk, Network 사용률
- 애플리케이션 모니터링 - 쿠버네티스 상에서 컨테이너화된 애플리케이션 모니터링
 - 지원 애플리케이션 : Java, Node.js, Python
 - 필터링을 이용한 특정 애플리케이션 모니터링
 - 진행중 트랜잭션 정보, 종료된 트랜잭션 정보 모니터링
 - 애플리케이션 그룹 및 개별 애플리케이션의 TPS 및 응답시간 추이
 - 5분간 집계된 사용자 수 추이
- 쿠버네티스 마스터 메타 정보 확인
 - ETCD에 저장된 마스터 메타 정보를 와탭 콘솔 상에서 확인 가능
 - 주요 쿠버네티스 구성요소 리스트 및 각 구성요소 및 상세 스펙(YAML) 정보

성능 분석

- 쿠버네티스 Pod 피크 사용량 및 초기화 시간
- 기간별 Out of Memory 발생 컨테이너 정보 및 메모리 차트
- 애플리케이션(APM)과 동일 기능
 - 매트릭스 차트
 - MSA 분석
 - 스택

이벤트 알림

- 이메일, 문자, SNS(Telegram, Slack) 발송
- 이벤트 지속 기간 및 임계치 설정에 의한 이벤트 발생
- 무음 설정 시간 등록에 의한 동일 이벤트 알림 억제 기능
- CPU, Memory, Disk 알림 기능
- 실시간 트랜잭션 개수 임계치 설정에 의한 알림
- 트랜잭션 에러 개수 임계치 설정에 의한 알림
- 응답시간 임계치 초과 트랜잭션 개수에 의한 알림
- 응답시간 분포의 가로 및 세로 분포 패턴에 의한 알림

통계 및 보고

- 일간/주간 보고서 : 실시간 사용자, TPS, 응답시간, CPU 추이 등
- 월간 보고서
 - 실시간 사용자, TPS, 응답시간, CPU 추이 등
 - 일간 방문자 수 추이, 월간 방문자 수, 고객 충성도 지표
- 애플리케이션 단위 일간 보고서
- 애플리케이션 피크 성능 비교 보고서 : 전일, 전주 피크 타임과의 비교
- Pod 별 주간 CPU, Memory 보고서

컨테이너 맵

홈 화면 > 프로젝트 선택 > 대시보드 > 컨테이너 맵

와탭 쿠버네티스의 [컨테이너 맵](#)은 쿠버네티스 모니터링의 시작점입니다. 사용자 설정 보기에 따라 각 개체는 하나의 컨테이너, 또는 하나의 Pod를 나타냅니다.

- 컨테이너 맵을 통해 분석 대상을 선택하고 그룹을 구성할 수 있습니다.
- 상태 지표 및 임계치를 설정해 개별 컨테이너 또는 Pod의 상태를 한 눈에 볼 수 있습니다.
- 개별 컨테이너 또는 Pod를 선택해 상세 정보를 확인할 수 있습니다. [트레이스](#), [메트릭스](#), [로그](#), [이벤트](#)와 같은 핵심 모니터링 데이터를 제공합니다.

다음 동영상은 [컨테이너 맵](#) 메뉴에 대한 기본 안내입니다.

기본 화면 안내

1 CP 컨테이너 맵

2 LIVE 16:44:20 Container 그룹화 ConState (state) + 적용된 필터가 없습니다. 리셋

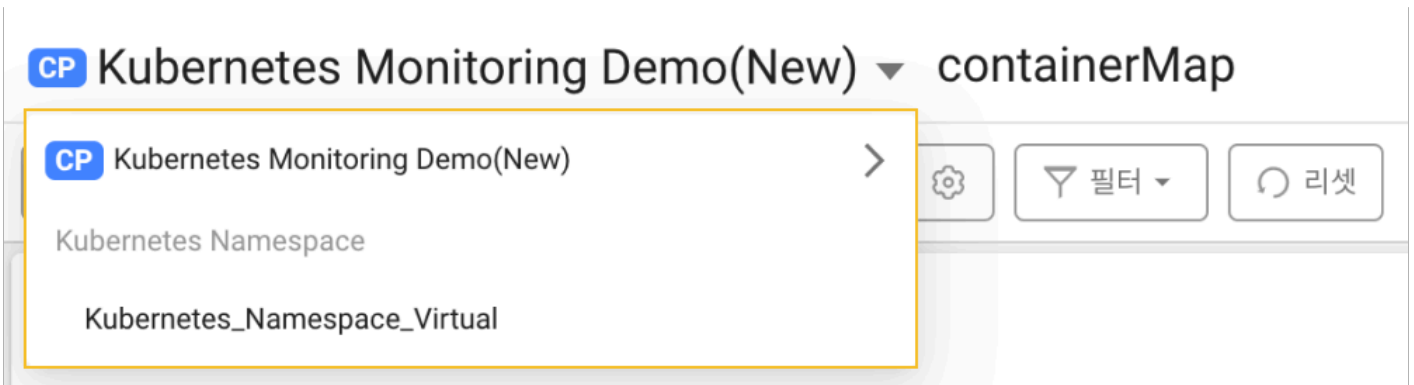
3

Container

전체	Running	Node	Namespace	Deployment	ReplicaSet	Pod	Image
33	33	3	4	13	13	30	18

Container					ActiveTX				
CpuByLimit	CpuByRequest	MemByLimit	NetRxBytes	NetTxBytes	Normal	Slow	Very Slow	Total	TPS
2.31%	4.85%	27.29%	10.5MiB	8.7MiB	1	0	0	1	0.66

1 클러스터 프로젝트 - 네임스페이스 프로젝트 이동



클러스터 프로젝트 **CP** 하위에 네임스페이스 프로젝트를 생성한 경우 컨테이너 맵 화면 위의 **더보기** 아이콘 선택 시 프로젝트 리스트 박스가 나타납니다. 프로젝트 리스트 박스를 통해 클러스터 프로젝트를 포함한 프로젝트 목록을 확인하고 프로젝트 간 이동할 수 있습니다.

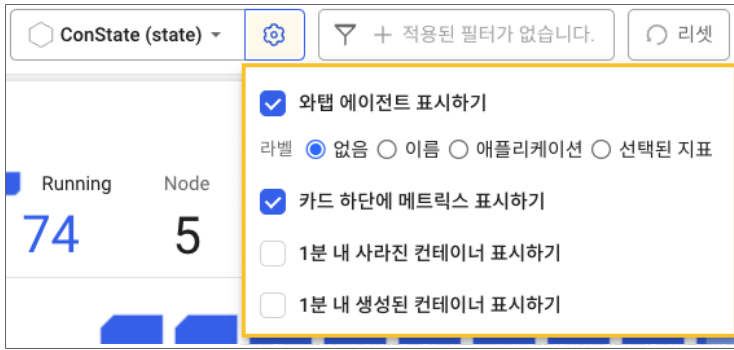
① 네임스페이스 프로젝트 생성 및 관리에 관한 자세한 내용은 [다음 문서](#)를 참조하세요.

2 사용자 설정



컨테이너 맵이 제공하는 여러 사용자 설정을 통해 원하는 화면 보기를 생성할 수 있습니다. 화면 순서대로 사용할 수 있는 기능은 다음과 같습니다.

- 카테고리: 분석 대상을 선택할 수 있습니다. **Container** 또는 **Pod**를 지정할 수 있습니다.
- **그룹화**: **그룹화**(클러스터링) 옵션을 통해 클러스터 구성 요소 간의 계층관계를 총 2단계까지 표현할 수 있습니다. 자세한 내용은 [다음 문서](#)를 참조하세요.
- **ConState(state)**: 차트 지표를 선택할 수 있습니다.
- **⚙** 버튼을 클릭하면 각 개체에 컨테이너 또는 Pod의 정보를 표시할 수 있는 옵션을 선택할 수 있습니다.



- **와탭 에이전트 표시하기:** 컨테이너 또는 Pod의 정보를 표시할 수 있습니다. 원하는 **라벨** 옵션을 선택하세요. 라벨 옵션에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.
- **카드 하단에 메트릭스 표시하기:** 화면 하단에 메트릭스 정보를 표시할 지 여부를 선택할 수 있습니다.
- **1분 내 생성된 컨테이너 또는 Pod 표시하기:** 1분 이내 생성된 컨테이너 또는 Pod를 표시하는 옵션입니다.
- **1분 내 사라진 컨테이너 또는 Pod 표시하기:** 1분 이내 사라진 컨테이너 또는 Pod를 표시하는 옵션입니다.
- **🔍 필터:** 사용자가 선택한 항목에 따라 필터링해 컨테이너 또는 Pod의 상태를 확인할 수 있습니다.
- **🔄 리셋:** 사용자가 설정한 화면 보기를 초기화합니다.

지표 선택 및 임계치 설정

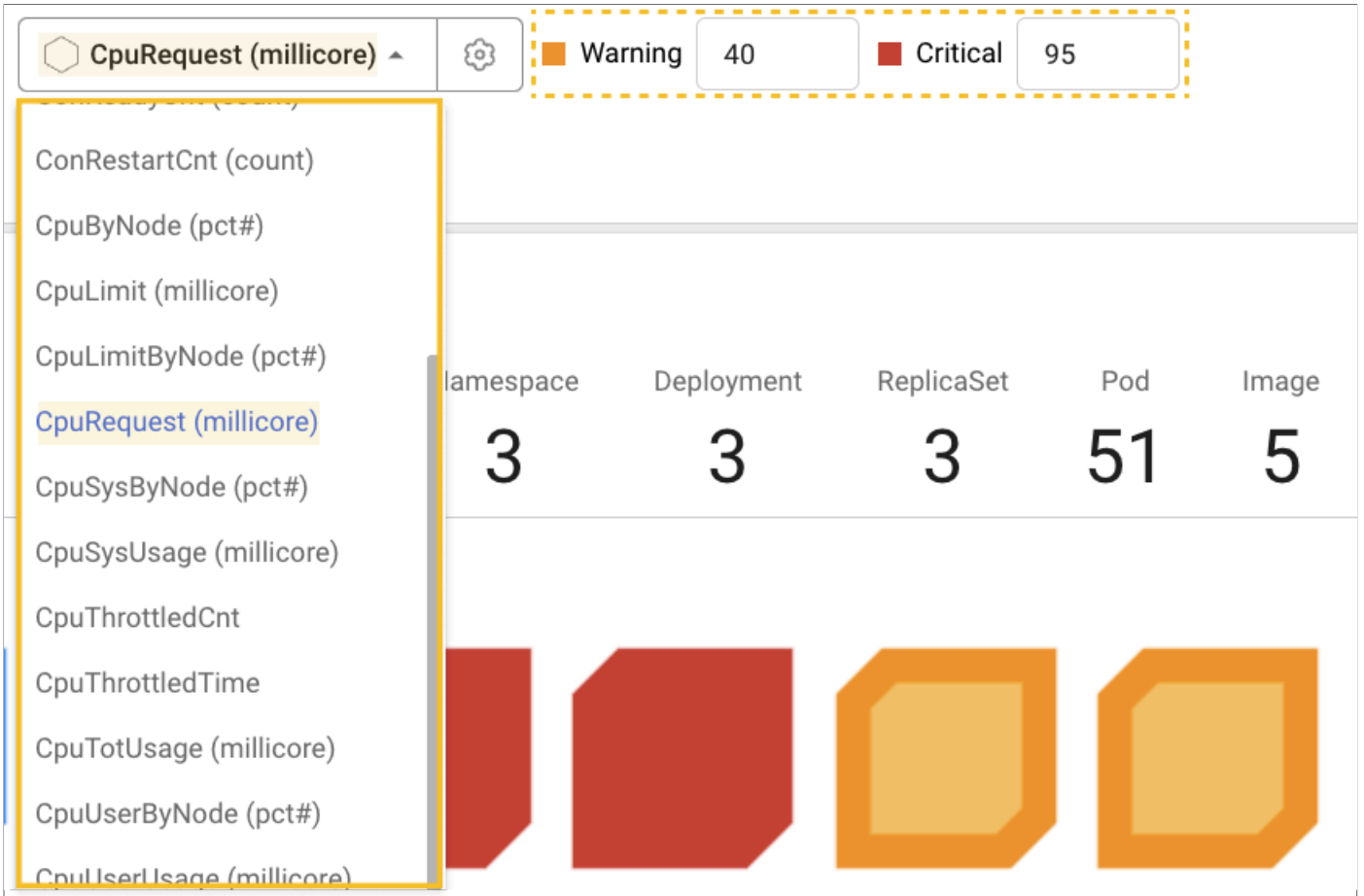


차트 지표 선택 및 임계치 설정은 컨테이너 맵의 상태 표현을 결정합니다. 선택한 지표의 단위가 숫자인 경우 임계치를 설정할 수 있습니다. 설정한 임계치는 다음 범주에 따라 색상으로 표현됩니다.

- 0 이상 **Warning** 이하: **파란색**
- **Warning** 초과 **Critical** 이하: **주황색**
- **Critical** 초과: **빨간색**

- ① 수치형 지표 단위로는 **pct**(percentage), **bps**, **millicore** 등이 있습니다. **pct#**는 100% 이상의 값이 나올 수 있는 지표 단위입니다.
- 차트 지표 선택과 임계치 설정에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

라벨 옵션

- **없음**: 블럭에 라벨을 표시하지 않습니다.
- **이름**: 컨테이너/POD의 이름을 표시합니다.
- **애플리케이션**: 컨테이너/POD 내의 애플리케이션을 표시합니다.
- **선택된 지표**: `ConState(state)` 항목에서 선택한 차트 지표를 표시합니다.

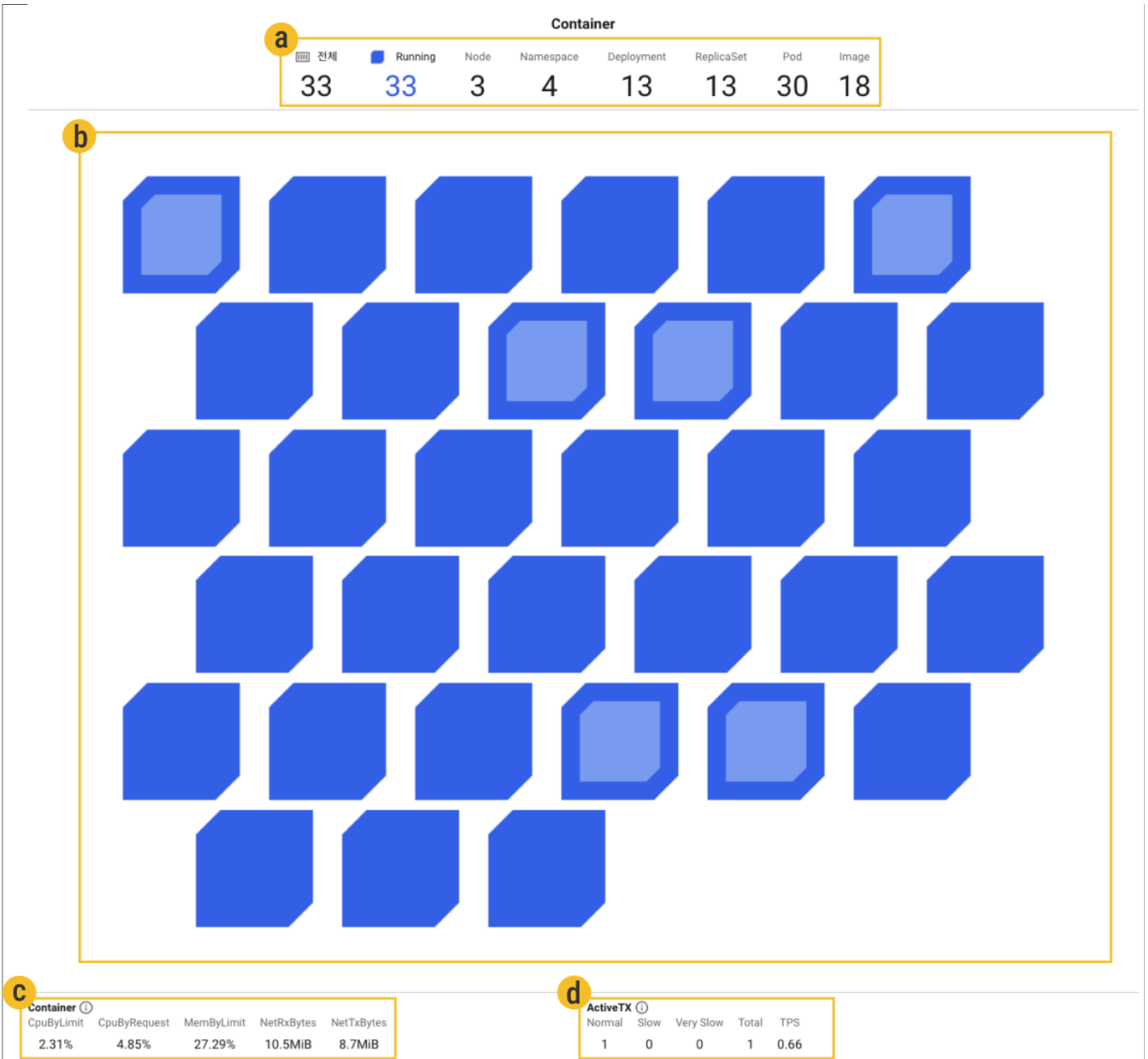
실시간 조회 및 과거 조회

컨테이너 맵은 기본적으로 실시간 조회 기반이지만 여러 시점에 따른 조회 기능을 제공합니다. 실시간 조회 변경은 화면 상단 왼쪽의 시간 선택자를 이용하세요.



조회 상태	설명
실시간 조회 (LIVE)	기본 모드 10초 인터벌로 실시간 상태 변경
일시정지 조회 (PAUSE)	실시간 상태 변경 OFF 현재 상태를 자세히 확인할 때 사용
과거시점 조회	사용자가 과거 특정 시점을 지정하여 확인 년/월/일/시/분 지정 가능

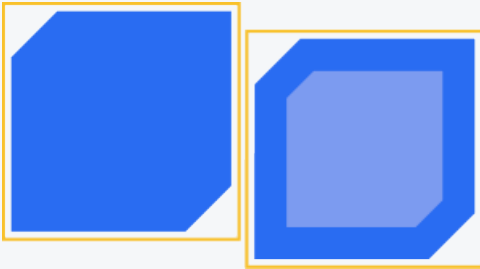
3 컨테이너 맵 차트



- **a** 그룹화된 대상 전체 개수 및 주요 리소스 개수를 나타냅니다.

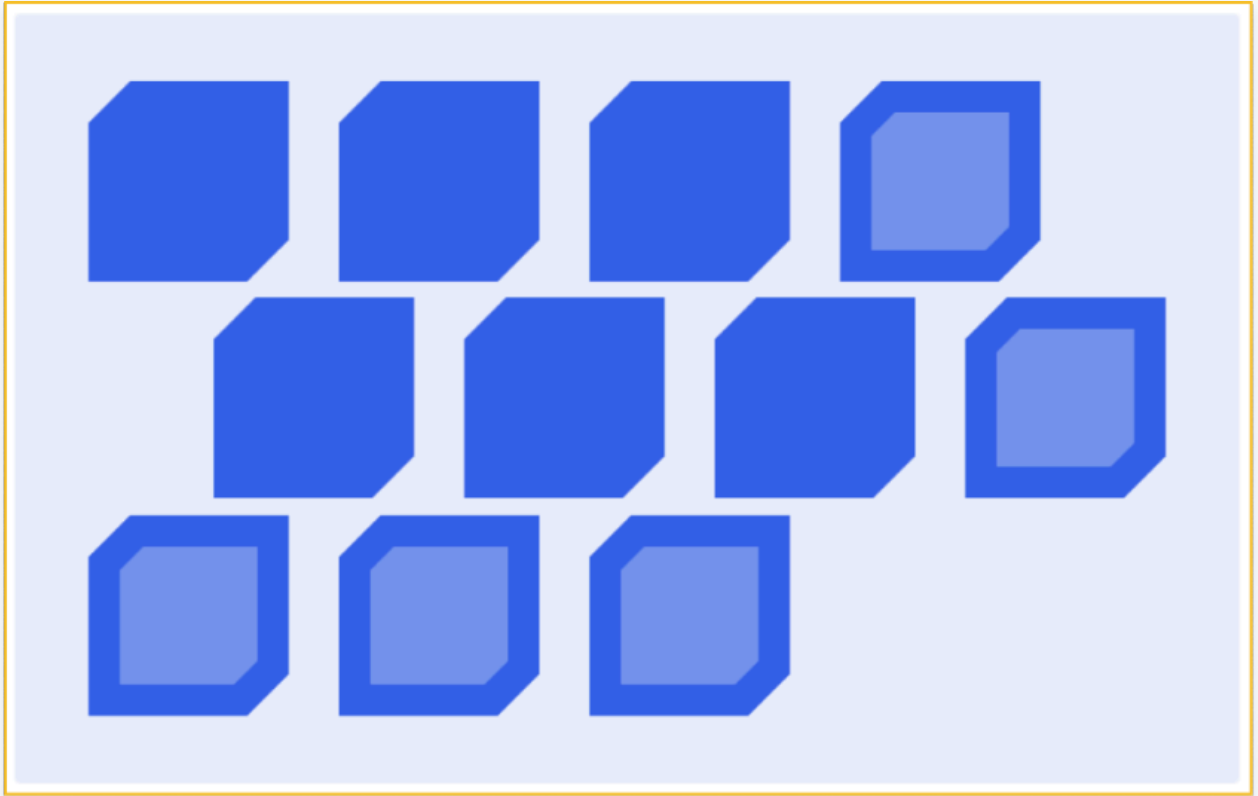
- **b** 각 개체의 모양과 색상은 컨테이너 또는 Pod의 특성과 상태를 나타냅니다.
 - 색상 : 컨테이너 또는 Pod의 상태를 의미합니다. **Normal**, **Warning**, **Critical** 외 상태에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.
 - 테두리
 - 개체 중에는 테두리가 표현된 것이 있습니다. 이는 컨테이너에 Java, Node.js, Python 등 마이크로 에이전트를 설치한 대상입니다.
 - 만약 1개 Pod 내에 2개 컨테이너가 있고, 2개 중 1개 컨테이너에만 마이크로 에이전트를 설치했더라도 테두리가 표시됩니다. 즉 Pod 내 컨테이너 중에 하나라도 마이크로 에이전트가 설치되어 있다면 테두리 표시가 됩니다.

❗ 블록과 마이크로 블록



컨테이너 맵에서는 다각형 하나를 **블록(Block)**이라고 합니다. 이때 에이전트가 있는 블록(블록 안에 블록)은 **마이크로 블록(Micro Block)**이라고 합니다.

❗ 그룹



컨테이너 맵 화면에서 전체 블록을 선택한 경우 해당 다중 블록을 그룹이라고 합니다.

- **c** 그룹화된 대상의 CPU 및 Memory 관련 지표를 나타냅니다.
- **d** 그룹화된 대상 내부 애플리케이션의 트랜잭션 정보를 나타냅니다.

요약 보기와 상세 보기

사용자는 관심있는 컨테이너 또는 Pod를 선택해 요약 정보 및 상세 정보를 확인할 수 있습니다. 해당 정보에는 다음과 같은 핵심 모니터링 데이터를 포함하고 있습니다.

- **트레이스 (Trace)**
컨테이너 상의 애플리케이션 추적을 위한 APM 도구를 의미합니다.
- **메트릭스 (Metrics)**

노드, 쿠버네티스의 다양한 지표 데이터를 의미합니다.

- **로그 (Log)**

컨테이너, 애플리케이션의 각종 로그를 의미합니다.

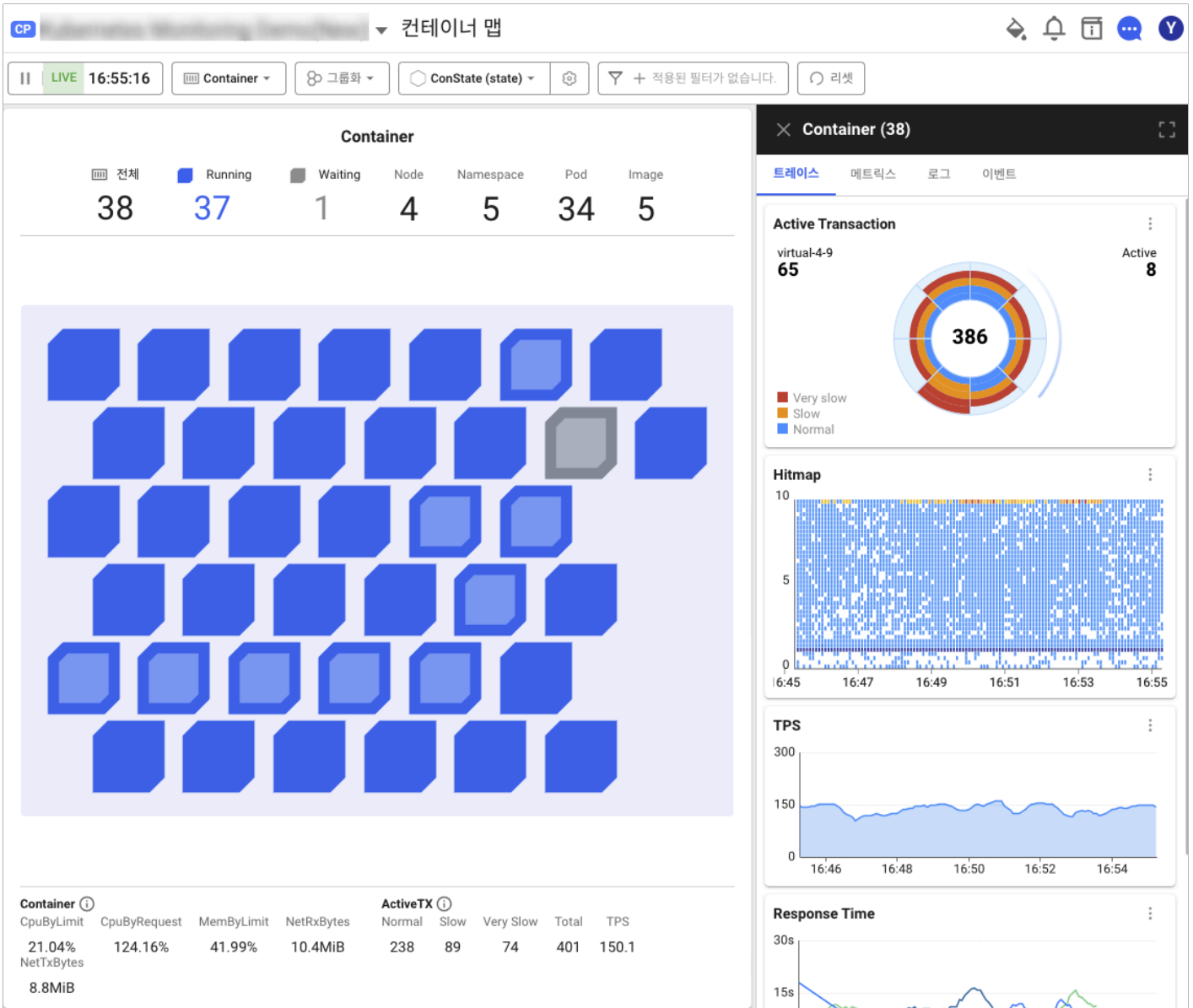
- **이벤트 (Event)**

컨테이너와 애플리케이션 실행 과정에서 발생하는 이벤트를 기록한 것을 의미합니다.

이러한 데이터들을 기반으로 사용자는 쿠버네티스 환경의 Observability를 확보할 수 있습니다. 아래 **요약 보기**, **상세 보기** 화면 모두 위 4개 항목을 중심으로 정보를 제공합니다.

요약 보기

특정 개체 혹은 개체 집합을 클릭하세요. 오른쪽에 나타나는 **요약 보기** 화면을 통해 해당 객체의 요약 정보를 확인할 수 있습니다.



• **트레이스**

컨테이너 맵의 **트레이스** 탭을 통해 컨테이너 상에서 실행되는 애플리케이션의 개별 트랜잭션을 추적할 수 있습니다.

• **메트릭스**

컨테이너 맵의 **메트릭스** 탭을 통해 각 컨테이너의 자원 할당량, 사용량 등의 각종 지표 데이터를 실시간으로 확인할 수 있습니다.

• **로그**

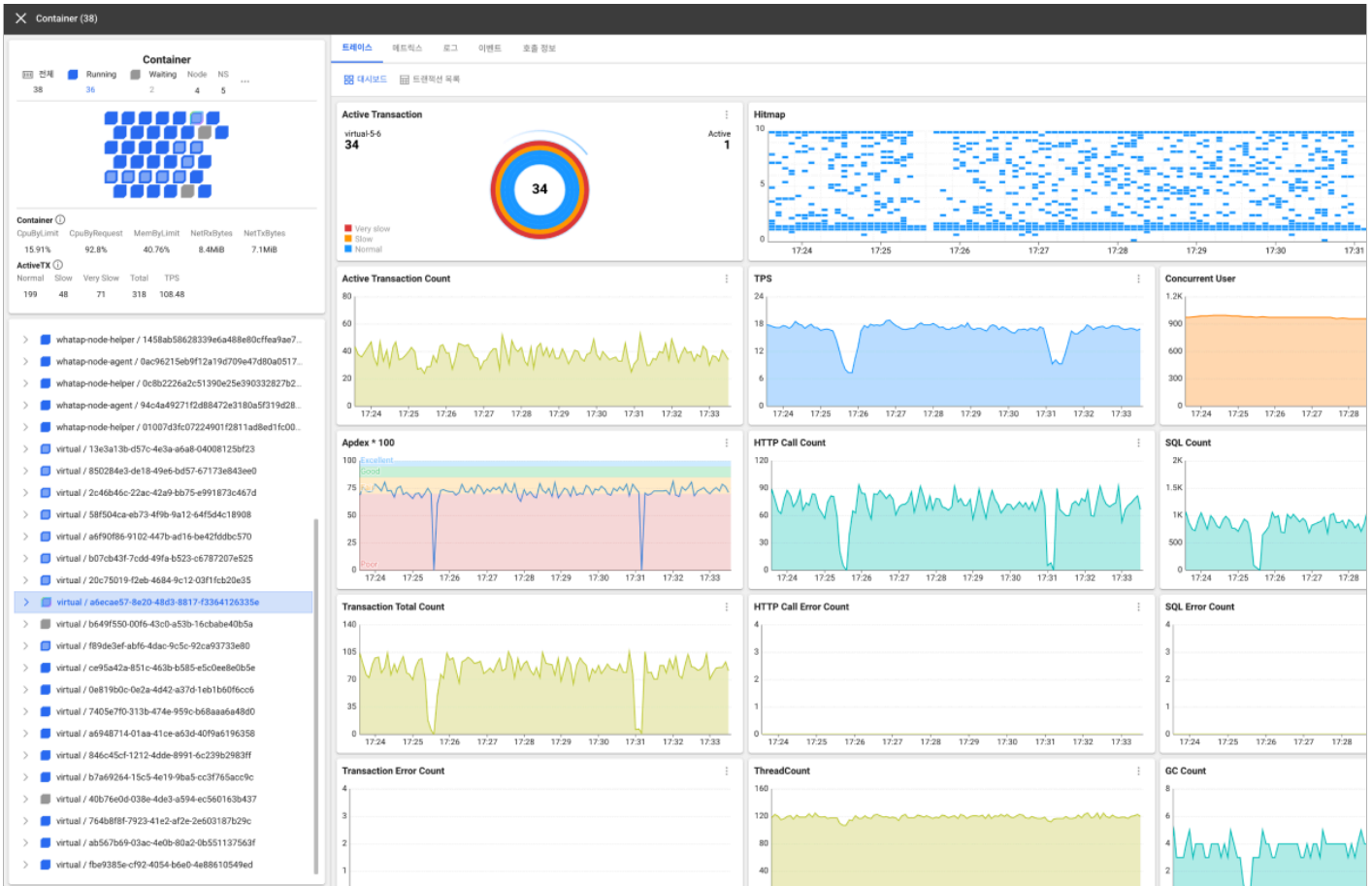
컨테이너 맵의 **로그** 탭을 통해 각 컨테이너의 표준 출력 로그 및 컨테이너 상의 애플리케이션이 생성하는 로그를 통합 관리, 조회할 수 있습니다.

• 이벤트

컨테이너 맵의 **이벤트** 탭을 통해 쿠버네티스 클러스터에서 발생한 이벤트 및 와탭의 경고 기능을 이용한 사용자 임계치 기반 이벤트의 통합 조회가 가능합니다.

상세 보기

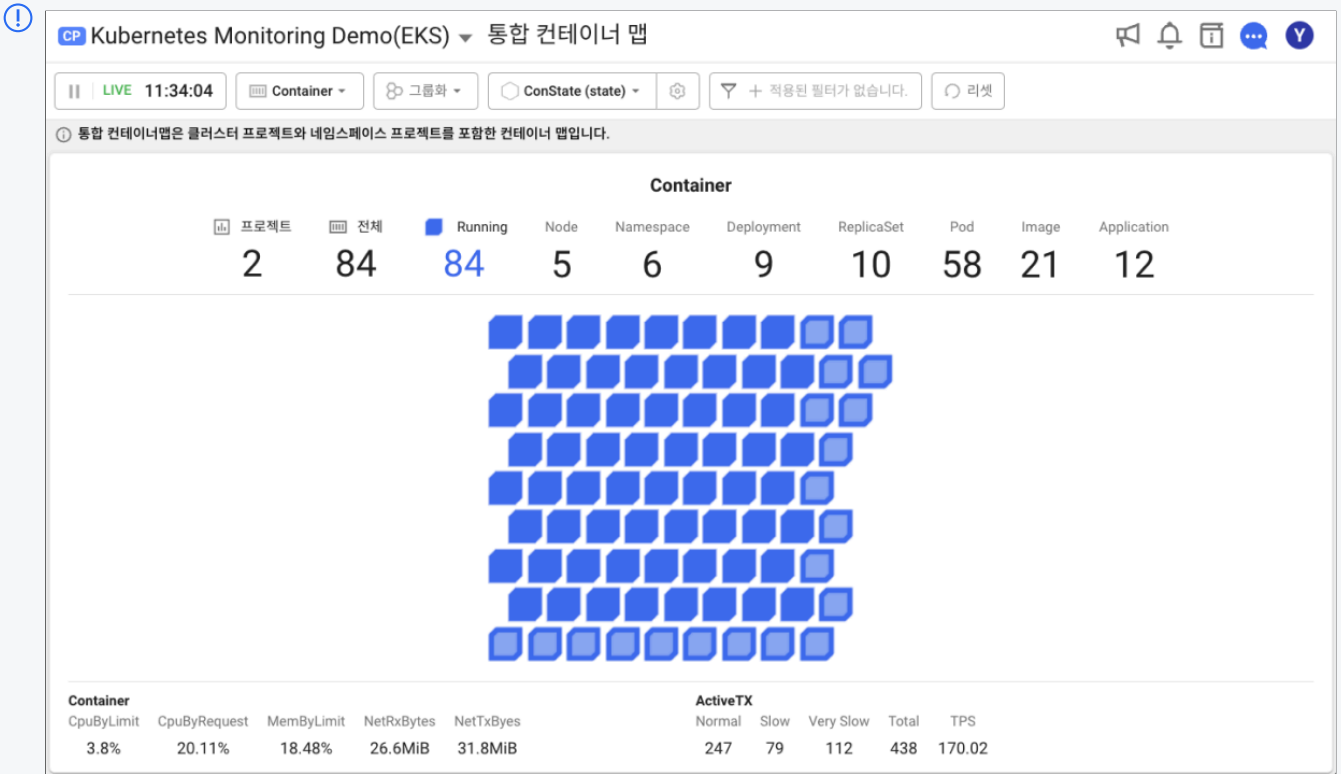
요약 보기 화면의 오른쪽 위에 [] 버튼을 클릭하세요. **상세 보기** 화면으로 전환합니다. **상세 보기** 화면 구성에 대한 자세한 안내는 [다음 문서](#)를 참조하세요.



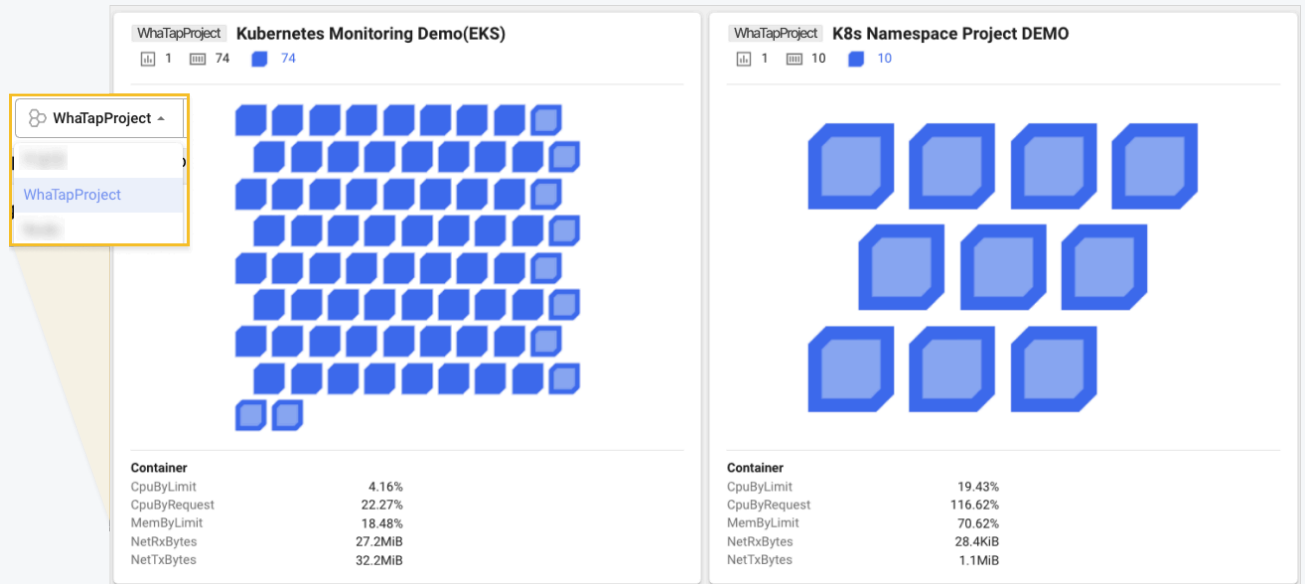
📌 통합 컨테이너 맵

홈 화면 > 프로젝트 선택 > 대시보드 > 통합 컨테이너 맵

클러스터 프로젝트(**CP**)와 네임스페이스 프로젝트(**Namespace**)를 포함한 컨테이너 맵을 조회하려면 **통합 컨테이너 맵**을 확인하세요.



통합 컨테이너 맵은 프로젝트별 그룹화 옵션을 추가로 제공합니다. 그룹화 옵션 중 **WhaTapProject** 그룹화를 통해 한 화면에서 클러스터 프로젝트 및 네임스페이스 개별 프로젝트 내 컨테이너 맵을 다음과 같이 조회할 수 있습니다.



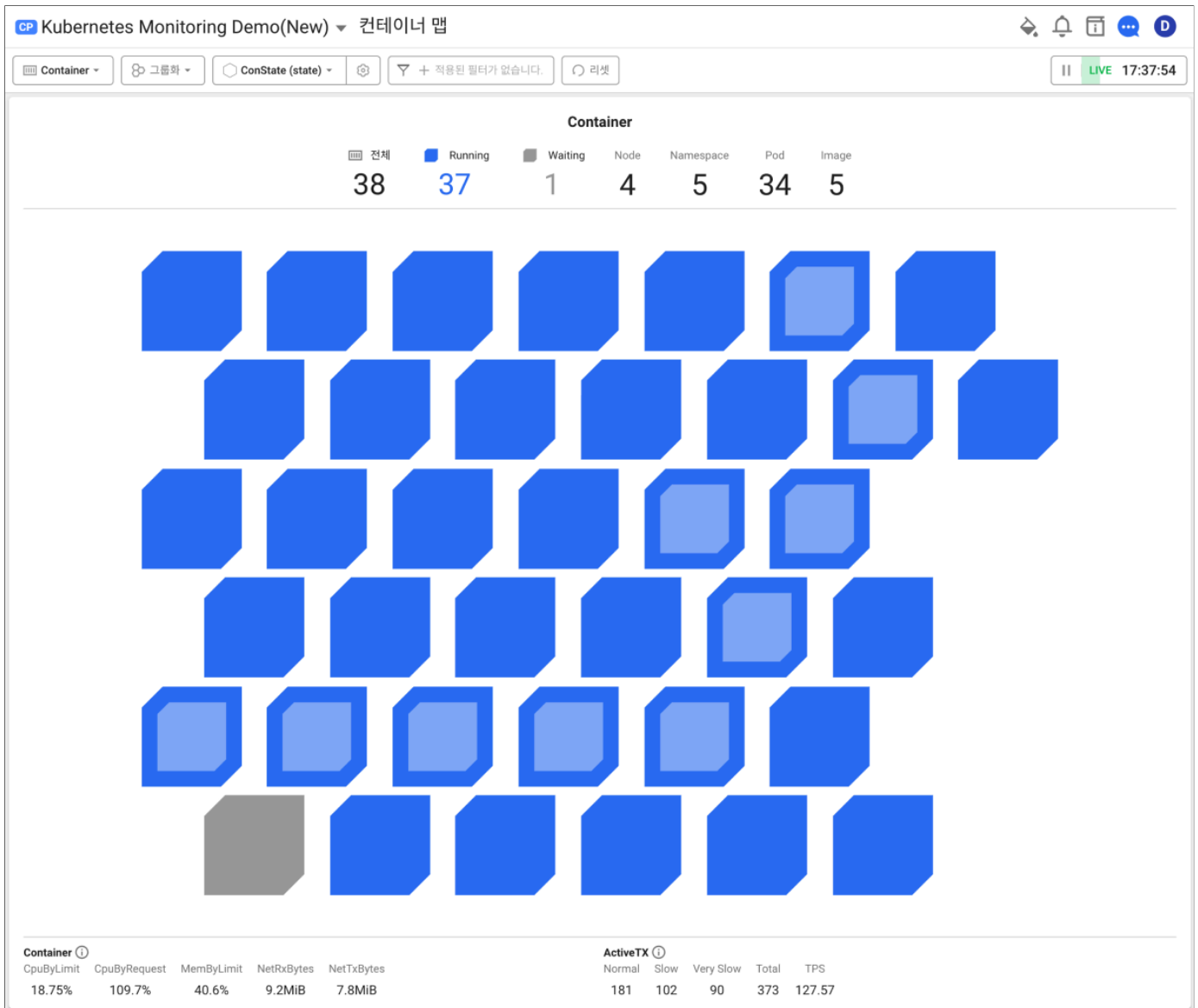
컨테이너 뷰, Pod 뷰

카테고리 선택 (컨테이너 뷰, Pod 뷰)

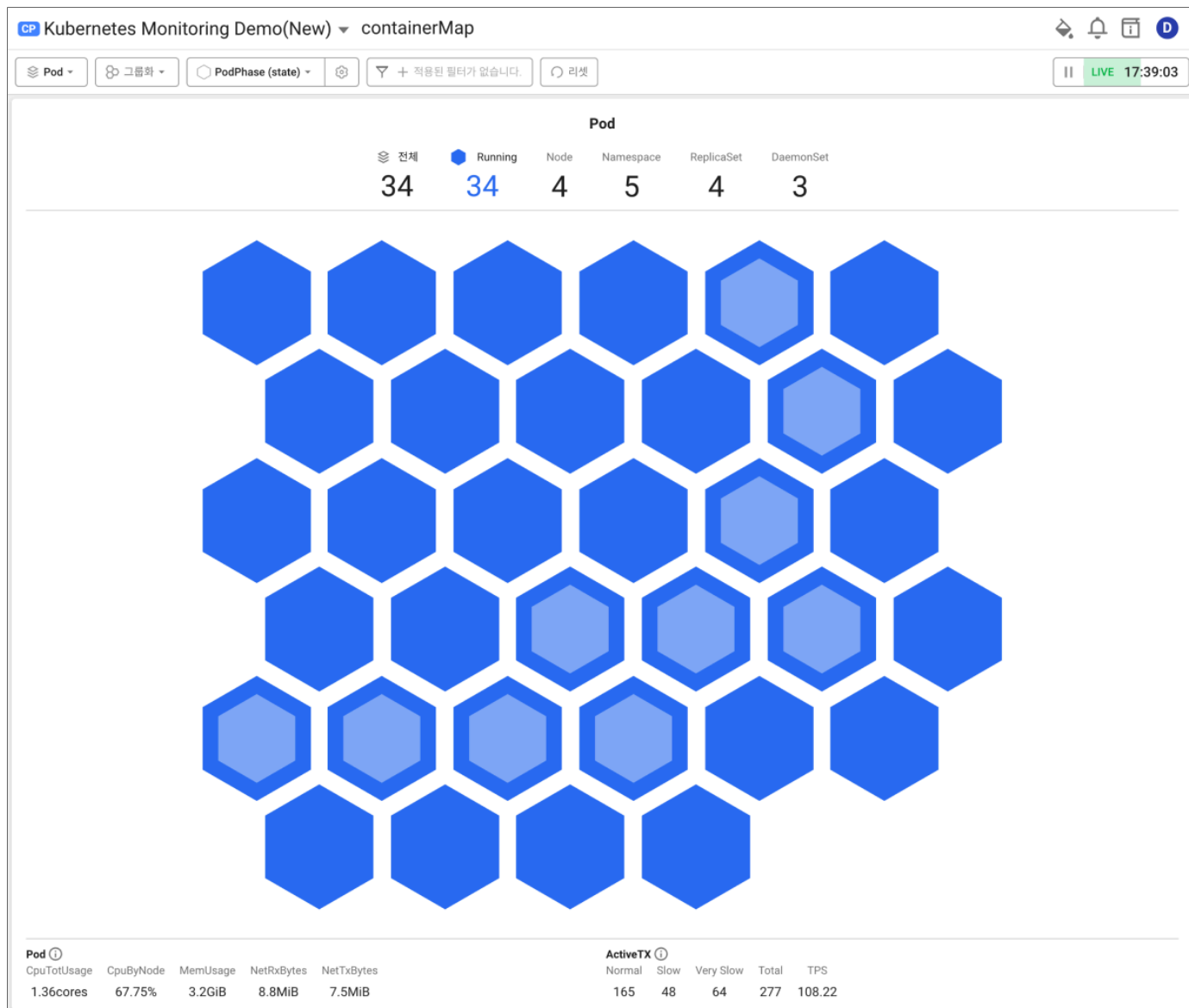
[컨테이너 맵](#) 메뉴에 처음 진입하면 사용자 클러스터 프로젝트 내의 개별 컨테이너 혹은 Pod가 모두 펼쳐진 것을 확인할 수 있습니다. 1개 Pod 내에 2개 이상의 컨테이너가 실행될 수 있기 때문에 컨테이너, Pod 각각에 대한 뷰를 가지고 있어야 합니다.

와탭 쿠버네티스의 [컨테이너 맵](#) 메뉴에서는 **컨테이너 기준 뷰**, **Pod 기준 뷰**의 두가지 모드를 모두 지원하기 때문에 컨테이너 단위의 모니터링, Pod 단위의 모니터링 등 상황에 맞는 모니터링이 가능합니다.

- 컨테이너 기준 뷰



• Pod 기준 뷰



컨테이너 기준 뷰와 Pod 기준 뷰는 컨테이너 맵 메뉴 화면 위에 사용자 설정을 통해 전환할 수 있습니다.

ⓘ 구버전 쿠버네티스 에이전트를 사용한다면 Pod 기준 뷰를 확인할 수 없습니다. 에이전트 업그레이드를 수행하세요.

Master



```
kubectl set image deployment/whatap-master-agent whatap-master-agent=whatap/kube_mon:{에이전트 버전} -n whatap-monitoring
```

Node

```
kubectl set image daemonset/whatap-node-agent whatap-node-agent=whatap/kube_mon:{에이전트 버전} -n whatap-monitoring
```

대상 그룹화

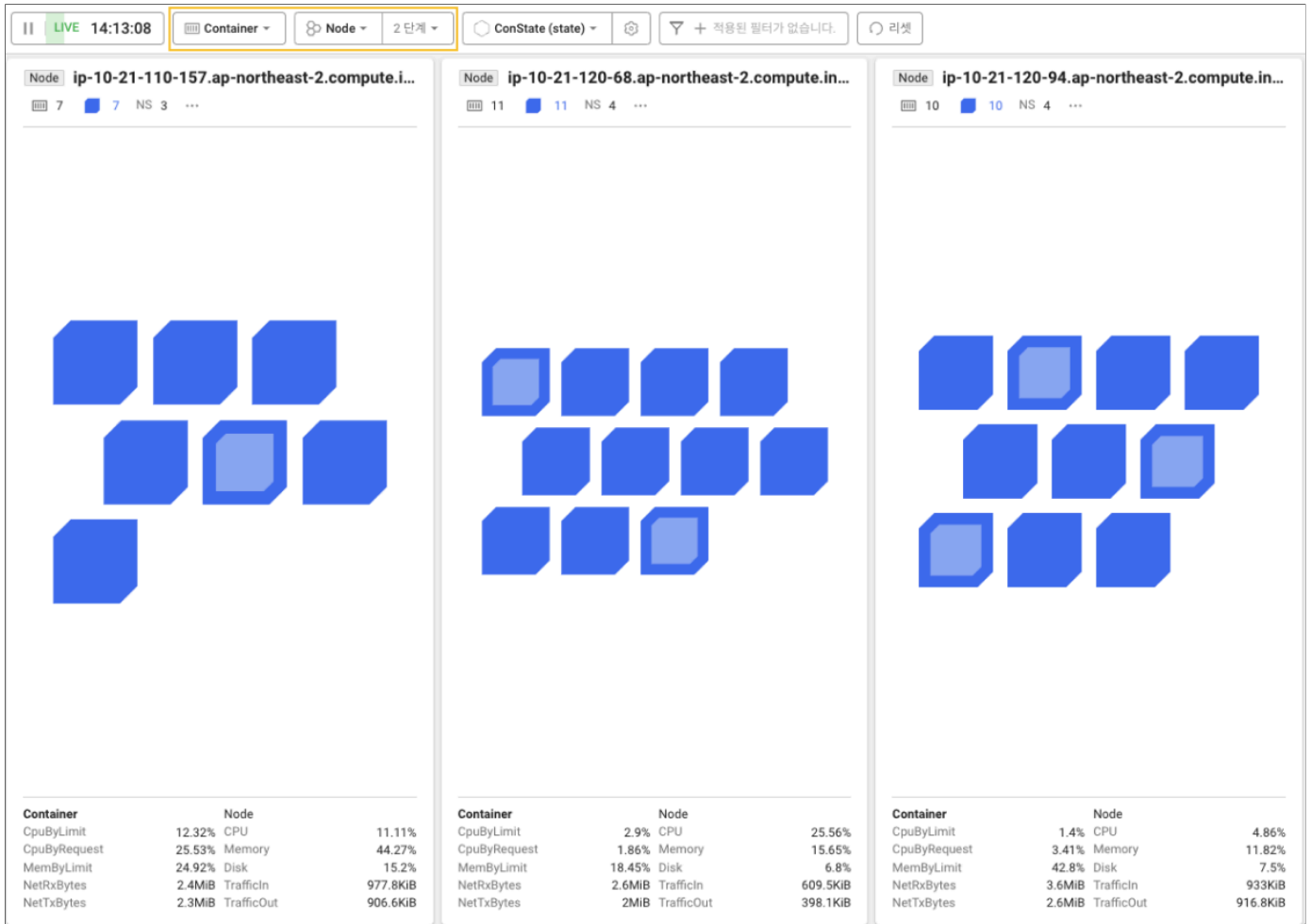
사용자 관심사에 따른 다양한 그룹화

기본 **컨테이너 맵** 메뉴의 화면을 보면 모든 컨테이너 또는 Pod가 펼쳐져 있습니다. 그런데 쿠버네티스의 컨테이너/Pod를 관찰할 때는 특정 기준으로 대상을 선별해 모니터링하는 경우가 많습니다. 물론 대상을 바라보는 기준은 매우 다양합니다. 쿠버네티스 환경을 대하는 사용자의 관심사가 모두 동일하지 않기 때문입니다.

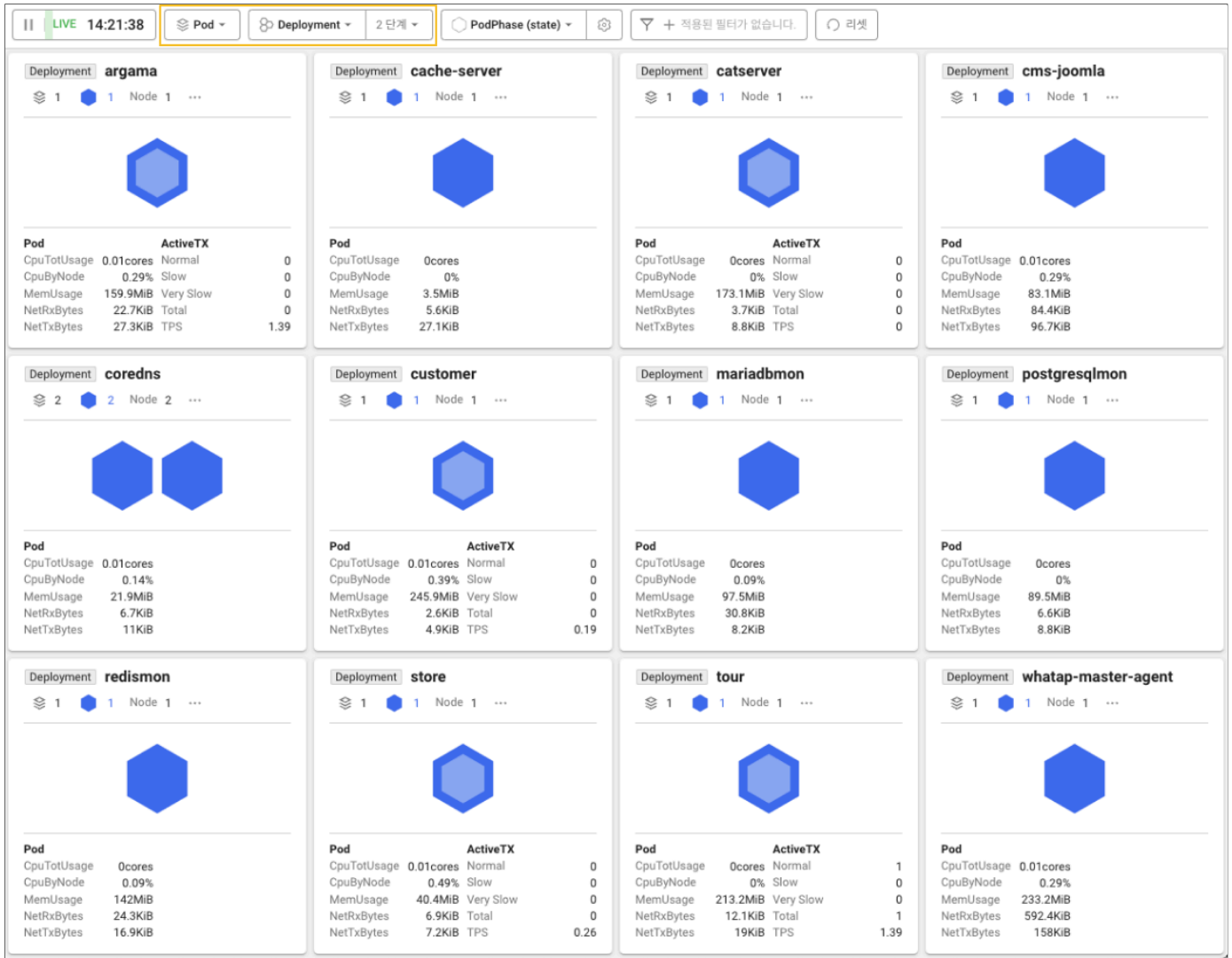
서버 담당자는 쿠버네티스 리소스를 실행하기 위한 Node 상태에 관심이 많습니다. 반면 특정 서비스의 애플리케이션 담당자는 Deployment나 개별 Pod, 혹은 컨테이너 상태에 관심이 많습니다. 만약 신규 버전 배포 이후라면 Deployment 전체적인 상태 뿐만 아니라 신규 이미지를 사용하는 컨테이너들을 모니터링하고자 할 것입니다.

그래서 이러한 다양한 사용자 니즈를 충족하고자 제공하는 기능이 **그룹화**입니다.

- 컨테이너 기준 뷰 + Node 그룹화



• Pod 기준 뷰 + Deployment 그룹화



주요 그룹화 속성

컨테이너 맵의 뷰에 따라 지원하는 그룹화 속성은 다음과 같습니다.

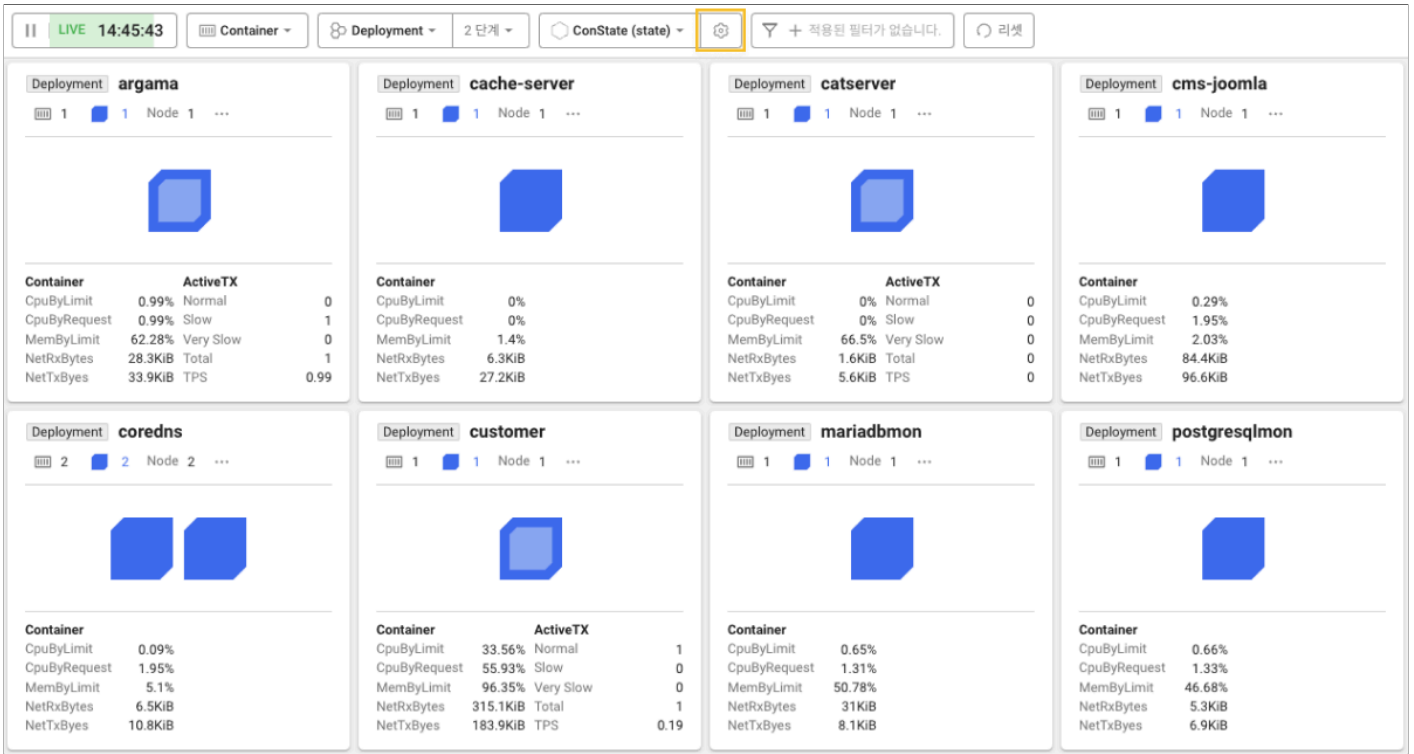
기준 뷰	그룹화 속성
Container	미설정 (All) Node

기준 뷰	그룹화 속성
	Namespace Deployment ReplicaSet ReplicationController Pod ContainerName Image
Pod	미설정 (All) Node Namespace Deployment ReplicaSet ReplicationController DemonSet StatefulSet

❗ 화면 위에 설정한 그룹화 설정(속성 필드 및 필터링 옵션 포함) 정보는 사용자 브라우저 내에 저장됩니다. 만약 저장된 설정을 초기화하고 싶다면 [리셋](#) 버튼을 선택하세요.

그룹화 & 카드

컨테이너 맵은 1개 이상의 카드로 구성되어 있습니다. 만약 그룹화를 적용하지 않은 상태라면 프로젝트 내 전체 컨테이너/Pod가 하나의 큰 카드 내에 표현됩니다. 그룹화를 적용하면 각 그룹 단위로 카드를 구성합니다.



위 화면은 Deployment를 기준 속성으로 하여 그룹화를 적용한 것입니다. 각 Deployment 단위로 카드가 나열되어 있음을 알 수 있습니다.

그룹화 적용 시 기준 속성으로 그룹화 할 수 없는 대상은 기본적으로 표시되지 않습니다. 하지만 해당 대상들도 화면에 표시할 수 있는 방법이 있습니다. 화면 위에 ⚙ 설정 버튼을 선택하세요. **알 수 없는 그룹 숨기기** 체크박스를 해제하면 그룹화에 해당하지 않는 대상들만 모아 별도의 카드로 표현합니다.

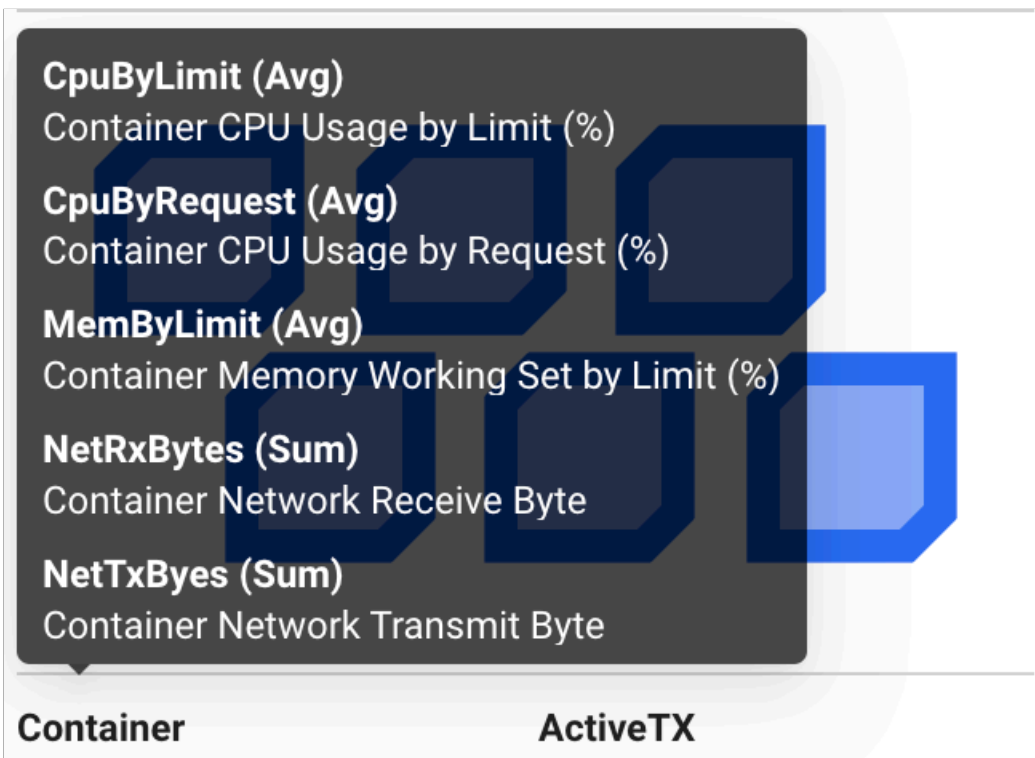
카드별 메트릭 데이터

각 카드 아래에는 그룹화된 대상들에 대한 메트릭 데이터를 표시합니다. 다음은 제공하는 지표 정보입니다.

기준 뷰	리소스 지표	애플리케이션 지표
Container	CpuByLimit CpuByRequest MemByLimit NetRxBytes	Normal TX Slow TX Very Slow TX Total TX

기준 뷰	리소스 지표	애플리케이션 지표
	NetTxBytes	TPS
Pod	CpuTotUsage CpuByNode MemUsage NetRxBytes NetTxBytes	

사용자의 커서가 **Container** (또는 **Pod**), **ActiveTX** 위에 올라가 있으면 각 메트릭에 대한 설명을 팝업으로 확인할 수 있습니다.

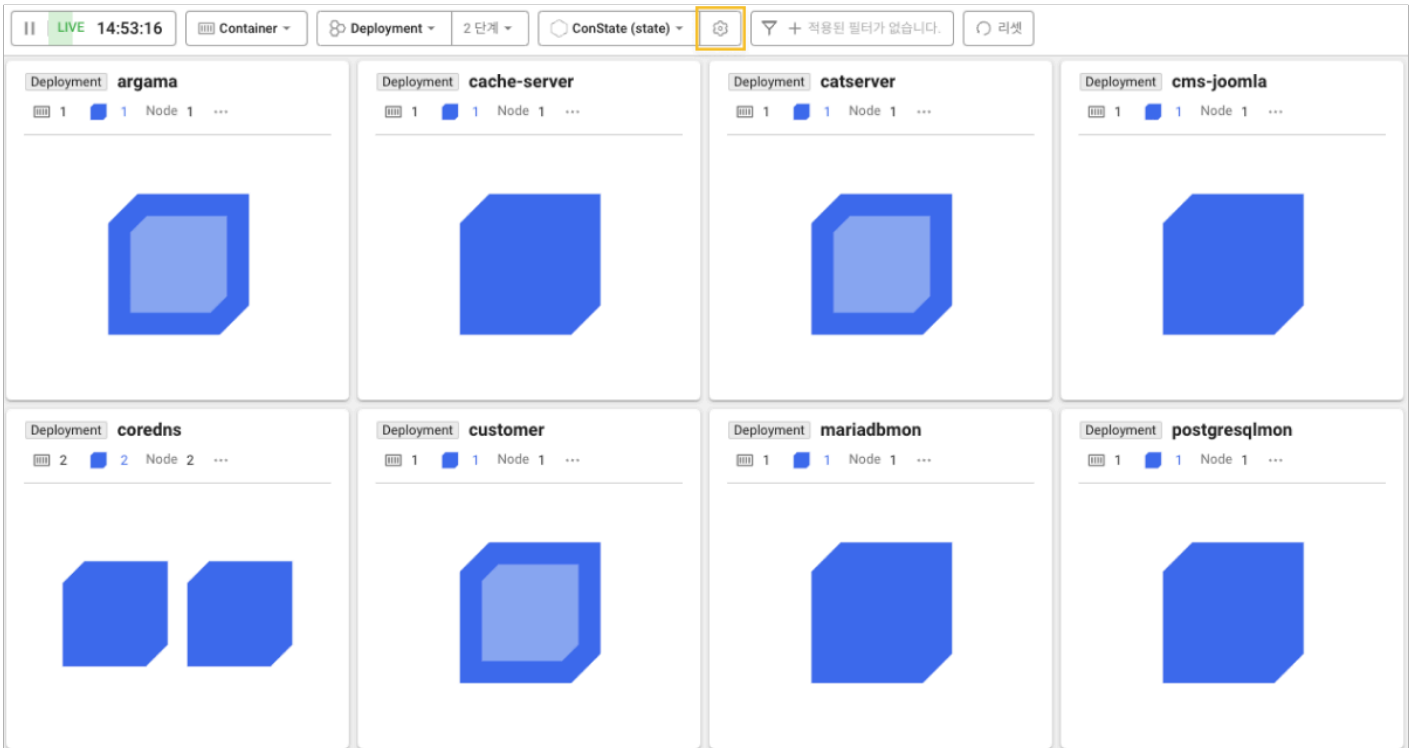


- 팝업에 표시된 지표 중 (Avg)가 붙은 항목은 카드 아래에 표시된 같은 이름의 지표 수치가 그룹 내 개별 대상이 가진 값의 평균 값이라는 의미입니다.
- 팝업에 표시된 지표 중 (Sum)이 붙은 항목은 카드 아래에 표시된 같은 이름의 지표 수치가 그룹 내 개별 대상이 가진 값의 합산 값이라는 의미입니다.

카드의 메트릭 데이터 감추기

카드 아래에 메트릭 데이터는 숨기고 오직 차트에만 집중할 수 있습니다. 화면 위에 ⚙️ 설정 버튼을 선택하세요. [카드 하단에 메트릭스 표시](#) 체크박스를 해제하세요. 이 기능은 카드 수가 많을 때 유용합니다.

다음과 같이 메트릭 데이터를 숨길 수 있습니다.

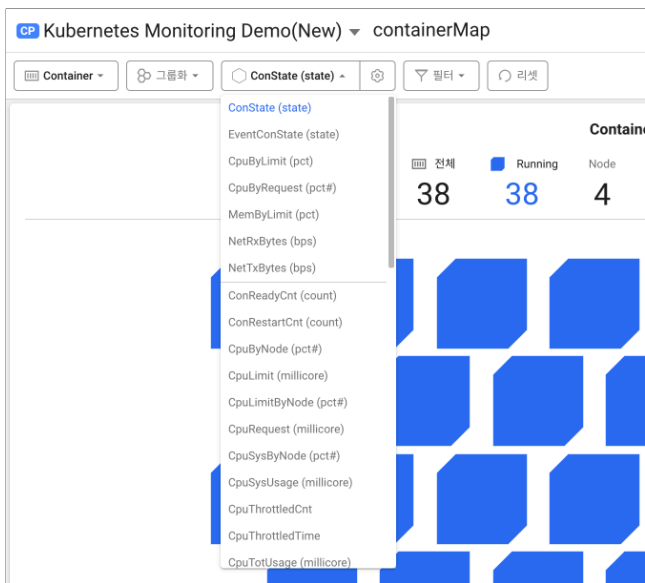


컨테이너 맵 지표와 상태

차트 지표 사용자화

와탭 쿠버네티스 모니터링 서비스의 **컨테이너 맵** 메뉴에서 제공하는 기본 차트 속성은 컨테이너 또는 Pod의 상태(State)입니다. 메트릭 이름으로는 **ConState**(Container State)입니다. 하지만 기본 상태가 아닌 다른 기준으로 맵을 구성할 필요가 있습니다. 예를 들어, 기본 State 상태가 아닌 CPU 사용량(millicore), 메모리 사용률(%), 네트워크 사용량(byte) 등을 기준으로 각 대상들의 상태를 파악해야 하는 경우도 있습니다.

와탭은 컨테이너의 차트 지표 변경 기능을 제공합니다. 변경 가능한 차트 지표 값은 와탭 쿠버네티스가 수집하는 메트릭스(지표)와 같습니다.

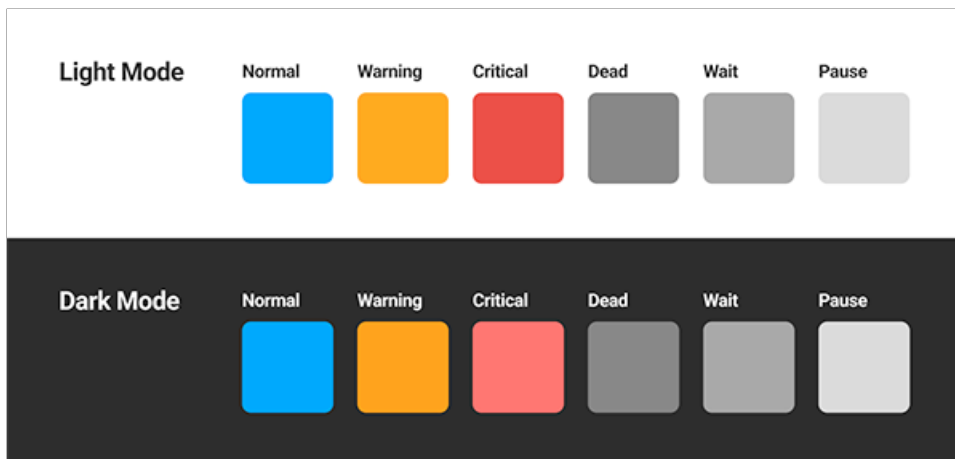


ⓘ 화면 위에 설정한 그룹화 설정(속성 필드 및 필터링 옵션 포함) 정보는 사용자 브라우저 내에 저장됩니다. 만약 저장된 설정을 초기화하고 싶다면 **리셋** 버튼을 선택하세요.

지표 임계치 설정 및 상태 색상

대상에 부여한 지표의 임계치 설정으로 각 대상의 상태를 색상으로 분류할 수 있습니다. 지표의 단위가 숫자인 경우(pct, bps, millicore)에 임계치 설정이 가능합니다.

Normal, Warning, Critical 등 상태별 색상 및 의미는 다음과 같습니다.



상태	설명
Normal	정상 상태 구간 표시 (~ Warning 미만 구간을 3등분)
Warning	Warning 임계치 구간 내에 있음을 표시 (Warning 초과 ~ Critical 이상)
Critical	Critical 임계치 구간 내에 있음을 표시 (Critical 초과 ~)
Dead	컨테이너 종료 진행 상태
Wait	컨테이너 시작 진행 상태
Pause	컨테이너 일시 중단 상태

이벤트 기반 컨테이너 상태

컨테이너 맵 메뉴에서 컨테이너 뷰는 `EventConState` 라는 상태 값을 제공합니다. `EventConState` 는 `Event-based Container State` 로 이벤트가 반영된 컨테이너 상태를 나타냅니다. `EventConState` 에 반영된 이벤트 종류는 다음과 같습니다.

이벤트	레벨
Kubernetes Event	Warning
WhaTap Event	Warning, Critical

- `Kubernetes Event` 는 쿠버네티스가 제공하는 이벤트 정보입니다. 사용자의 쿠버네티스 환경에서 `kubectl get events` 명령을 통해 확인 가능한 이벤트 정보와 동일합니다.
- `WhaTap Event` 는 사용자가 와탭 이벤트 기능을 통해 설정한 이벤트입니다.

`EventConState` 는 다음과 같은 방식으로 결정됩니다.

상태	상태값	원인
Critical	50	WhaTap 마이크로 에이전트에서 Critical 이벤트 발생 WhaTap 노드 에이전트에서 Critical 이벤트 발생 컨테이너 상태 OOMKILLED 전환
Warning	30	Kubernetes에서 Warning 이벤트 발생 WhaTap 마이크로 에이전트에서 Warning 이벤트 발생 WhaTap 노드 에이전트에서 Warning 이벤트 발생 컨테이너 상태 RESTARTING 전환
Dead	23	컨테이너 상태 DEAD 전환
Pause	22	컨테이너 상태 PAUSE 전환
Waiting	21	컨테이너 상태 WAITING 전환
Normal	10	컨테이너 상태 Normal 전환

와탭 이벤트 중에는 '이벤트 상태가 해결되면 추가 알림'을 설정한 이벤트가 있습니다. 내부적으로 두가지 유형의 이벤트를 다음과 같이 구분합니다.

- 상태 해결을 추적하지 않는 이벤트 : `stateless` 이벤트
- 상태 해결 시 추가 알림을 설정한 이벤트 : `stateful` 이벤트

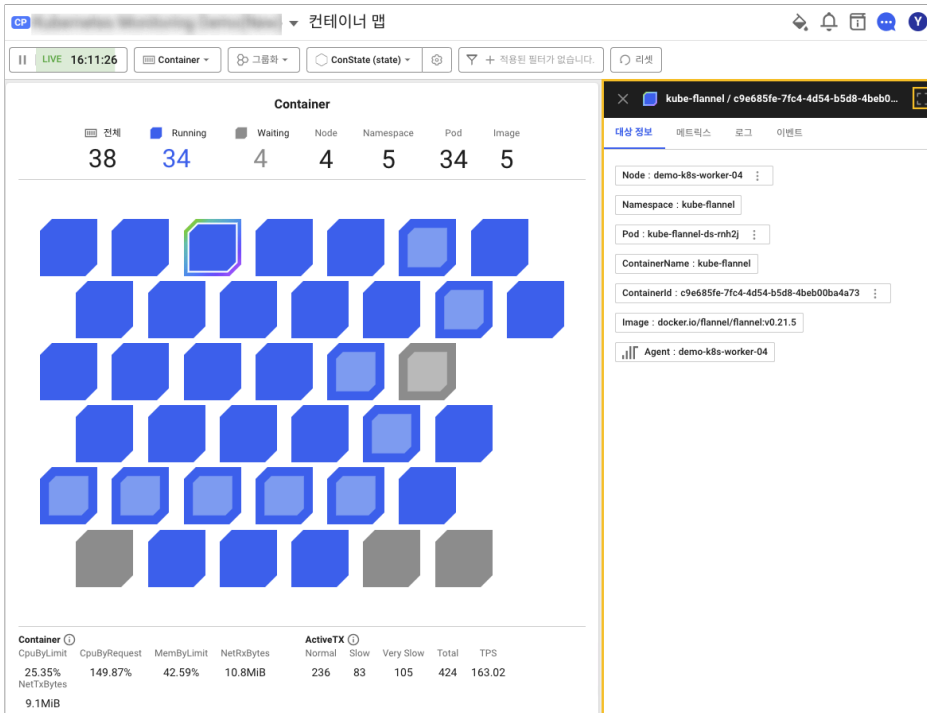
컨테이너에서 발생한 `stateless`, `stateful` 이벤트는 `EventConState` 상태를 결정하는데 다음과 같이 사용합니다.

- `stateless` 이벤트 : 컨테이너에서 발생한 최근 1분 이내의 `stateless` 이벤트를 `EventConState` 상태에 반영
- `stateful` 이벤트 : 발생 시점부터 종료 시점까지의 `stateful` 이벤트를 `EventConState` 상태에 반영

컨테이너 맵 상세 보기

컨테이너 맵 메뉴에서 선택한 대상에 대한 자세한 정보를 상세 보기 화면에서 확인할 수 있습니다.

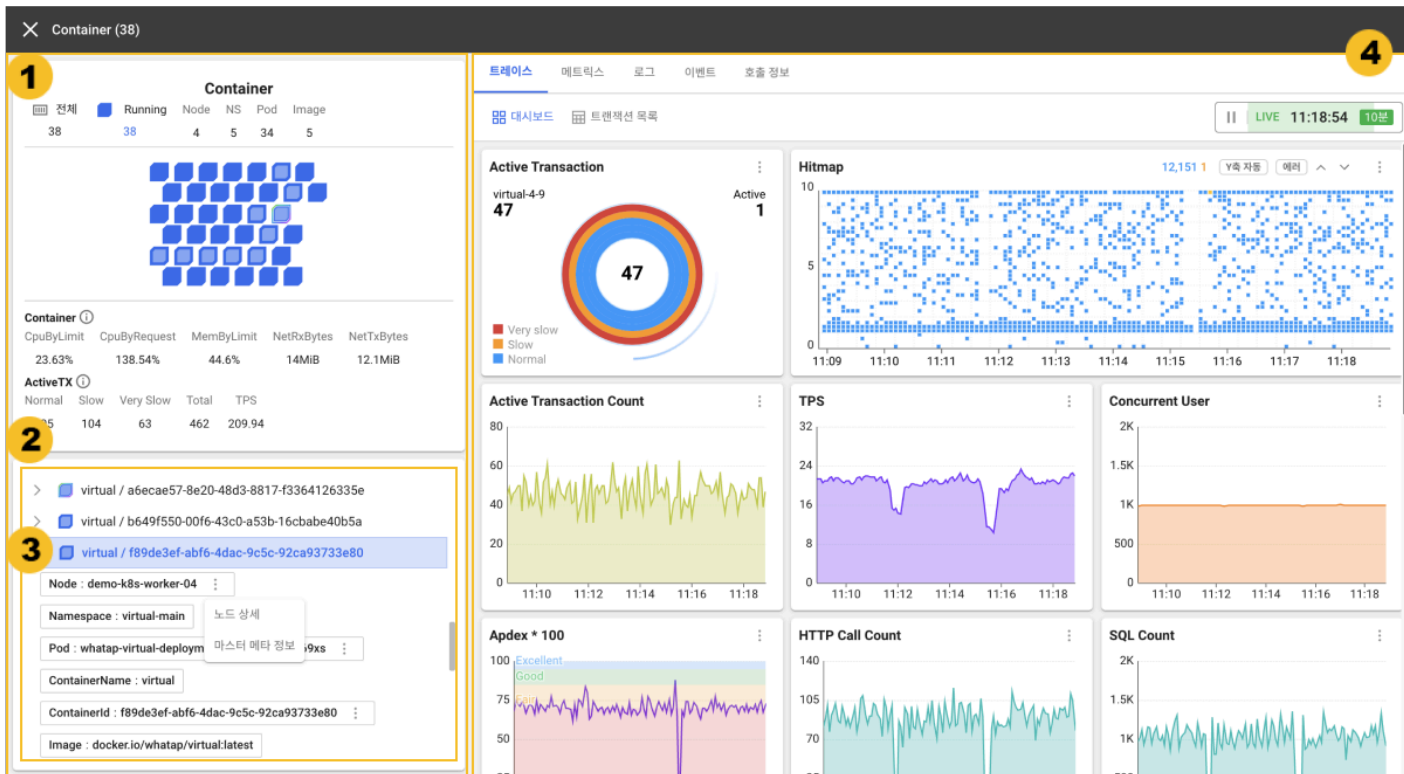
1. 컨테이너 맵 메뉴 화면에서 조회를 원하는 특정 블록(컨테이너 또는 Pod)을 선택하세요.
2. 화면 오른쪽에 다음과 같이 요약 보기 패널이 나타납니다.



3. 요약 보기 패널 오른쪽 상단에 [상세 보기] 아이콘을 선택하면 더욱 자세한 정보를 확인할 수 있습니다.

기본 화면 안내

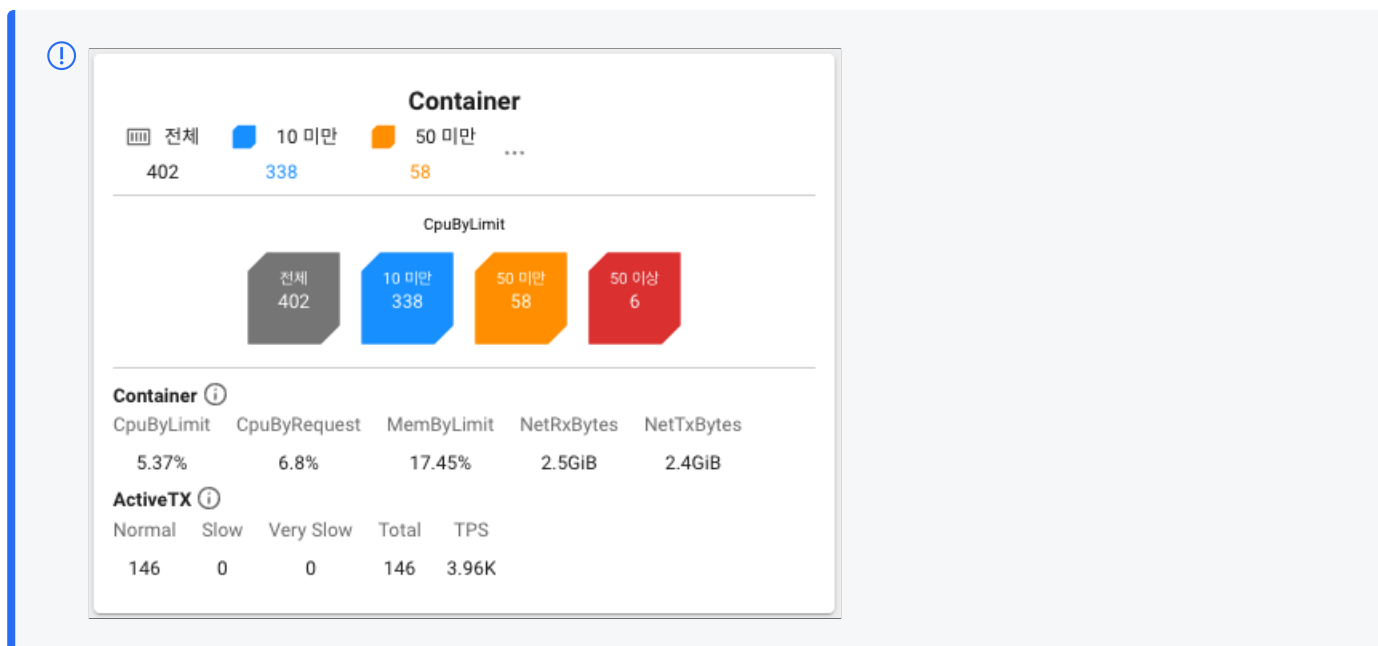
컨테이너 맵 상세 보기 화면은 다음과 같이 구성되어 있습니다. 기본 정보는 카드와 목록 형식으로 확인할 수 있습니다. 핵심 모니터링 데이터를 조회할 수 있도록 트레이스, 매트릭스, 로그, 이벤트 탭을 제공합니다.



1 요약 정보 카드 (컨테이너 또는 Pod)

선택 대상의 그룹 정보를 확인할 수 있습니다. 대상 추가 선택 시 4 영역에서 선택한 대상에 대한 지표를 확인할 수 있습니다. 컨테이너 맵에서 선택한 카드와 동일한 뷰를 가지고 있습니다. 자세한 내용은 [다음 문서](#)를 참조하세요.

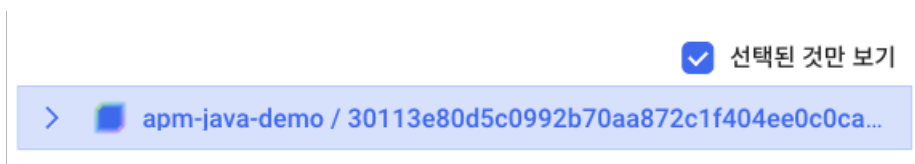
① 요약 정보 카드 조회 시 블록의 개수가 많은 경우 다음과 같이 표기 방식이 변경됩니다. 전체 블록, 정상, 경고, 위험 상태 블록 개수를 확인해 비교할 수 있습니다.



2 요약 정보 목록

선택 대상의 그룹 정보를 목록 형식으로 확인할 수 있습니다. 대상 추가 선택 시 4 영역에서 선택한 대상에 대한 지표를 확인할 수 있습니다.

- 2 목록 상단의 **선택된 것만 보기** 체크 박스 선택 시 다음과 같이 선택된 컨테이너 또는 Pod만 표시됩니다.



- 목록에서 3 선택한 대상 좌측의 > **더보기** 아이콘을 선택하면 **Node**, **Namespace**, **Deployment**, **ReplicaSet**, **Pod**, **ContainerName**, **ContainerId**, **Image**, **Agent** 등의 지표 정보를 조회할 수 있습니다.
 - Node**: 노드 상세, 마스터 메타 정보
 - Deployment**: 마스터 메타 정보
 - ReplicaSet**: 마스터 메타 정보
 - Pod**: 마스터 메타 정보
 - ContainerId**: 컨테이너 정보
- ⋮ **상세 조회**: 각 지표 안내 오른쪽의 상세 조회 아이콘을 선택하면 상세 또는 마스터 메타 정보를 조회할 수 있습니다.

- **4 트레이스, 메트릭스, 로그, 이벤트, 호출 정보**

- **트레이스**

트레이스 탭을 통해 쿠버네티스 컨테이너 상에서 실행 중인 애플리케이션의 트랜잭션 상세 정보를 모니터링할 수 있습니다. **대시보드** 뷰와 **트랜잭션 목록** 뷰를 제공합니다. 자세한 내용은 [다음 문서](#)를 참조하세요.

- **메트릭스**

메트릭스 탭을 통해 선택된 컨테이너 또는 Pod의 주요 메트릭 정보와 리소스 사용 패턴을 확인할 수 있습니다. 이를 통해 리소스 관련 문제점을 빠르게 식별하고 최적화 포인트를 찾을 수 있습니다. 자세한 내용은 [다음 문서](#)를 참조하세요.

- **로그**

로그 탭을 통해 쿠버네티스 컨테이너 및 컨테이너 애플리케이션에서 발생하는 로그를 조회할 수 있습니다. 로그 기능은 컨테이너의 실시간 로그 조회 기능과 특정 시점의 과거 로그 검색 기능으로 나누어집니다. 자세한 내용은 [다음 문서](#)를 참조하세요.

- **이벤트**

이벤트 탭을 통해 쿠버네티스 클러스터에서 발생하는 **Kubernetes** 이벤트와 사용자가 직접 설정한 조건에 의한 **WhaTap** 이벤트를 조회할 수 있습니다. 자세한 내용은 [다음 문서](#)를 참조하세요.

- **호출 정보**

호출 정보 탭을 통해 쿠버네티스 컨테이너 상에서 실행 중인 애플리케이션의 호출 흐름과 외부 호출 대상의 상태를 확인할 수 있습니다. 자세한 내용은 [다음 문서](#)를 참조하세요.

Trace

홈 화면 > 프로젝트 선택 > 대시보드 > 컨테이너 맵 > 대상 선택 후 상세 보기 화면 진입 > [트레이스](#)

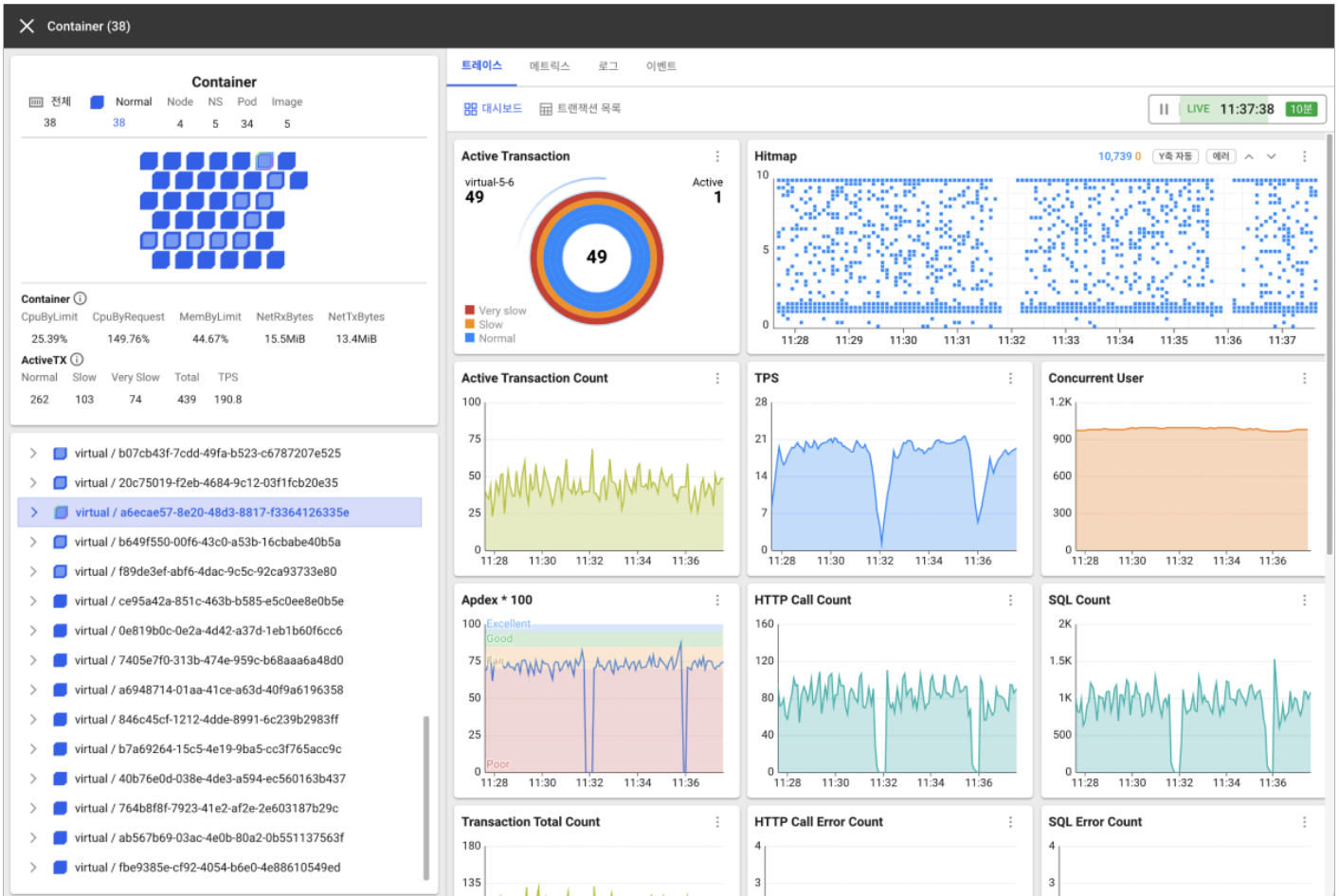
컨테이너화된 애플리케이션 트레이스

쿠버네티스 모니터링 시 쿠버네티스 환경 내부에서 실행 중인 애플리케이션에 대한 추적 없이 단지 클러스터만 모니터링해서는 현재 상태나 문제의 원인을 정확히 파악하기 힘듭니다.

문제의 원인을 추적하다보면 특정 트랜잭션의 SQL 처리 지연, 비즈니스 로직 오류 등 애플리케이션 레벨의 문제인 경우가 많기 때문입니다.

와탭 쿠버네티스의 트레이스 기능은 컨테이너 상에서 실행 중인 애플리케이션 상태 모니터링 정보를 제공합니다.

- ① 쿠버네티스 상에서 실행 중인 애플리케이션을 추적하기 위해서는 Java, Node.js, Python 등의 각 애플리케이션 컨테이너에 와탭 모니터링 에이전트가 설치되어야 합니다. 에이전트 설치에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.



트레이스 기능의 하위 메뉴는 [대시보드](#), [트랜잭션 목록](#)으로 구성되어 있습니다.

대시보드

대시보드는 다음과 같은 애플리케이션의 핵심 정보를 한눈에 제공합니다.

- **Active Transaction:** 현재 실행 중인 트랜잭션으로, 트랜잭션 실행 시간에 따라 색으로 분류
 - 빨간색 (Very slow) : 8초 이상
 - 주황색 (Slow) : 3초 ~ 8초
 - 파란색 (Normal) : 0초 ~ 3초
- **Active Transaction Count**(액티브 트랜잭션 추이)

- **Htimap**(히트맵): 종료된 트랜잭션 응답시간의 분포도
 - 가로축: 트랜잭션 종료 시간
 - 세로축: 수행 시간
 - 수행 시간이 긴 트랜잭션은 분포도 상단에 위치
 - 특정 영역 드래그 시 트랜잭션 정보 확인 가능
- **TPS**(Transaction Per Second): 초당 처리된 트랜잭션 건수
- **Concurrent User**: 동시 사용자
- **Apdex * 100**(Application Performance Index): 웹 애플리케이션의 사용자 만족도 측정
- **HTTP Call Count**: HTTP 호출 건수
- **HTTP Call Error Count**: HTTP 호출 에러 건수
- **SQL Count**
- **SQL Error Count**
- **Transaction Total Conunt**: 트랜잭션 건수
- **Transaction Error Conunt**: 트랜잭션 Error 건수
- **Response Time**: 응답시간
- **Heap Memory Usage**: Heap Memory 사용량, 사용 가능한 최대/현재 메모리
 - 시간에 따른 메모리 사용량 변화를 실시간 파악 가능
 - Python의 경우 Process Total 메모리 사용량
- **ThreadCount**: Thread 건수, Node.js, Python 제외
- **GC Count**: GC 횟수, Node.js, Python 제외
- **GC Old Generation Count**: Old 영역 GC 횟수, Node.js, Python 제외

트랜잭션 목록

트랜잭션 목록 화면은 지정된 시간대의 트랜잭션 목록 정보를 제공합니다. 만약 트랜잭션 분포도를 확인하여 특정 구간의 트랜잭션들만 확인하려면 위 트레이스 **대시보드**의 히트맵 차트에서 대상을 선택(드래그)하세요.

에이전트 명	시작 시간	경과 시간↑	URL	HTTP 호출 ...	SQL 건수	SQL 시간	SQL
virtual-5-6	2023/08/17 11:31:30.637	36,503	/account/save/division/pusan A M □	8,029	3	37	
virtual-5-6	2023/08/17 11:31:37.507	36,879	/order/pickup/unit/pusan A M □	11,594	4	57	
virtual-5-6	2023/08/17 11:31:29.531	36,912	/account/save/unit/daegu A M □	9,180	0	0	
virtual-5-6	2023/08/17 11:31:36.790	37,598	/account/delete/dept/daejun A M □	10,854	2	228	
virtual-5-6	2023/08/17 11:31:28.530	37,866	/sales/save/division/jeju A M □	7,914	4	393	
virtual-5-6	2023/08/17 11:31:29.537	38,615	/order/read/unit/daejun A M □	10,819	4	243	
virtual-5-6	2023/08/17 11:31:26.163	39,076	/account/pickup/employee/seoul A M □	10,087	3	317	
virtual-5-6	2023/08/17 11:31:24.527	39,963	/product/create/dept/daegu A M □	10,595	5	318	
virtual-5-6	2023/08/17 11:31:33.247	41,156	/account/save/employee/jeju A M □	9,443	6	966	
virtual-5-6	2023/08/17 11:31:32.407	41,251	/order/write/employee/seoul A M □	11,855	3	27	
virtual-5-6	2023/08/17 11:31:25.529	41,423	/product/save/division/kwangju A □	12,944	2	24	
virtual-5-6	2023/08/17 11:31:31.580	42,049	/product/delete/dept/daegu A M □	13,096	3	45	
virtual-5-6	2023/08/17 11:31:24.106	42,461	/product/save/dept/daejun A M □	13,590	7	45	
virtual-5-6	2023/08/17 11:31:22.526	43,252	/edu/load/unit/jeju A M □	13,642	3	396	

Metrics

홈 화면 > 프로젝트 선택 > 대시보드 > 컨테이너 맵 > 대상 선택 후 상세 보기 화면 진입 > [메트릭스](#)

쿠버네티스 메트릭스

쿠버네티스는 각 컨테이너의 리소스 사용량을 통제하고 지속적으로 모니터링하면서 최적의 상태를 유지합니다. 따라서 리소스 사용량을 확인하고, 해당 데이터를 기반으로 컨테이너 리소스 사용량이 적절히 설정되었는지를 확인하는 과정은 쿠버네티스 모니터링의 기본적인 패턴이라고 할 수 있습니다.

CPU와 Memory 메트릭

대표적인 쿠버네티스 리소스 설정은 컨테이너의 CPU Request/Limit, Memory Request/Limit입니다. 다음은 설정 예시입니다.

```

1  apiVersion: v1
2  kind: Pod
3  metadata:
4    name: <Pod명>
5  spec:
6    containers:
7    - name: <컨테이너 1명>
8      image: <컨테이너 1 이미지>
9      resources:
10     requests:
11       memory: "64Mi"
12       cpu: "100m"
13     limits:
14       memory: "128Mi"
15       cpu: "200m"
16    - name: <컨테이너 2명>
17      image: <컨테이너 2 이미지>
18      resources:
19     requests:
20       memory: "128Mi"
21       cpu: "250m"
22     limits:
23       memory: "256Mi"
24       cpu: "500m"

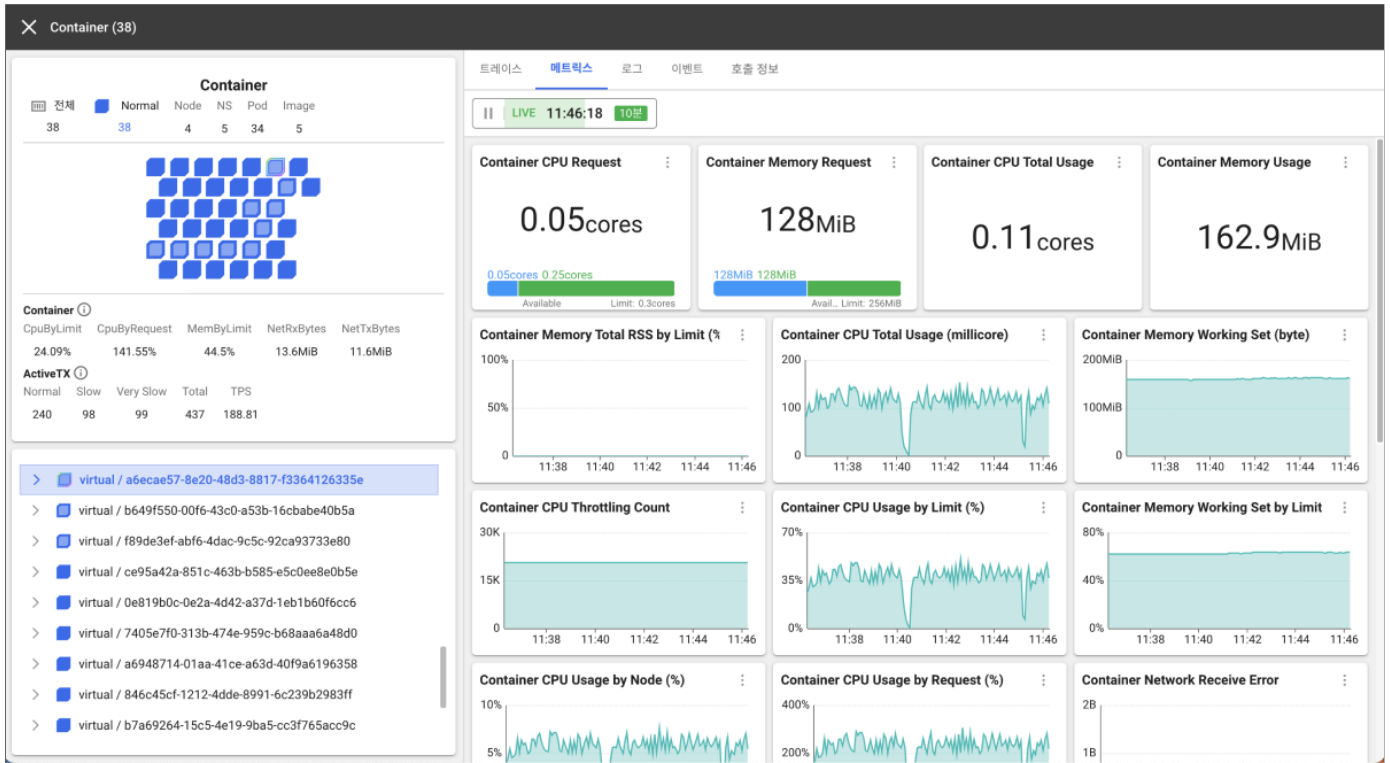
```

위 설정을 통해 해당 Pod 내에는 2개의 컨테이너가 존재합니다. 각 컨테이너의 리소스 설정은 다음과 같습니다.

대상 컨테이너	리소스 구분	리소스 설정	값
컨테이너1	CPU	Request	100m
		Limit	200m
	Memory	Request	64Mi
		Limit	128Mi
컨테이너2	CPU	Request	250m
		Limit	500m
	Memory	Request	128Mi
		Limit	256Mi

다양한 컨테이너의 리소스 설정 정보 및 현재 사용량을 쉽게 파악하기란 쉽지 않습니다. 그래서 와탭 쿠버네티스 모니터링 서비스는 특정 단일 컨테이너, 사용자가 선택한 복수의 컨테이너들, 또는 Deployment 전체를 대상으로 리소스 설정 및 현재 사용량을 쉽게 확인할 수 있는 기능을 제공합니다.

- 상세보기 화면 > [메트릭스](#) - 리소스 사용량 모니터링 및 분석



- 특정 컨테이너를 선택해 상태를 확인해 볼 수 있습니다.



위 데이터를 보면 CPU는 Request 0.2cores(200m), Limit 0.4cores(400m)로 설정되어 있습니다. 현재 CPU 사용량은 0.02cores(20m) (5초 평균)입니다. 우측의 CPU 관련 추이 그래프를 확인하면 전반적으로 여유가 있지만 간헐적으로 CPU 사용량이 증가하는 현상을 확인할 수 있습니다.

Memory는 Request 128MiB, Limit 256MiB로 설정되어 있습니다. 그런데 현재 Memory 사용량은 184MiB입니다. 우측의 Memory 관련 추이 그래프를 봐도 계속 Request 기준으로 100%를 초과하였고 Limit 기준으로도 70% 이상을 사용하고 있습니다.

CPU, Memory 관련 주요 와탭 메트릭은 다음과 같습니다.

대상 리소스	Shortname	Name	설명
CPU	CpuTotUsage	Container CPU Usage (millicore)	컨테이너 CPU 사용률
	CpuByRequest	Container CPU Usage by Request (%)	컨테이너 CPU 요청 대비 사용률
	CpuByLimit	Container CPU Usage by Limit (%)	CPU Limit 기준 CPU 전체 사용률
Memory	MemWs	Container Memory Working Set (byte)	컨테이너 메모리 working set
	MemWsByRequest	Container Memory Working Set by Request (%)	컨테이너 메모리 요청량 기준 working set 사용량
	MemWsByLimit	Container Memory Working Set by Limit (%)	컨테이너 메모리 할당량 기준 working set 사용량

① Pod가 실행 중인 노드에 사용 가능한 리소스가 충분하면 컨테이너에 설정된 Request 이상의 리소스를 사용할 수 있습니다. 와탭 메트릭 중 [Container CPU Usage by Request \(%\)](#), [Container Memory Working Set by Request \(%\)](#)의 값이 100%를 넘어설 수 있는 이유입니다.

예를 들어, 컨테이너의 Memory Request를 128MiB로 설정했다고 가정하겠습니다. 컨테이너(Pod)가 실행 중인 노드의 총 메모리가 4GiB이고, 해당 노드에 다른 Pod가 없는 경우 이 컨테이너는 128MiB이상의 메모리를 사용할 수 있습니다.

그 외 메트릭

와탭은 CPU, Memory 외에도 다양한 메트릭을 제공합니다.

- [Container Block I/O Read Byte](#) (IoReadBytes)
- [Container Block I/O Write Byte](#) (IoWriteBytes)
- [Container Network Receive Byte](#) (NetRxBytes)
- [Container Network Transmit Byte](#) (NetTxBytes)

와탭 쿠버네티스 메트릭스에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

Log

홈 화면 > 프로젝트 선택 > [대시보드](#) > [컨테이너 맵](#) > 대상 선택 후 [상세 보기](#) 화면 진입 > [로그](#)

쿠버네티스 로그

와탭 쿠버네티스 모니터링 서비스는 쿠버네티스에서 발생하는 로그를 수집하고 저장합니다.

쿠버네티스는 다음 두 종류의 로그가 있습니다.

- 각 컨테이너의 Standard Out(표준 출력) 로그
- 각 컨테이너 내부에서 실행 중인 애플리케이션이 출력하는 로그

이러한 로그는 컨테이너의 상태나 메트릭 정보로는 확인하기 힘든 모니터링의 사각지대를 보완하는 역할을 합니다. 이 로그를 확인하는 방법은 다음과 같습니다.

- 각 컨테이너의 Standard Out 로그의 경우는 컨테이너 내부에 진입하지 않아도 쿠버네티스가 제공하는 `kubectl logs` 명령을 통해 확인할 수 있습니다.
- 각 컨테이너 내부의 애플리케이션 로그는 컨테이너 내부에 진입하여 확인하거나 별도의 로그 수집 방안을 수립해야 합니다.

최근 로그 메시지 (컨테이너 Stdout/Stderr)

현재 컨테이너 로그를 바로 확인할 수 있는 기능입니다. 사용자의 클러스터 환경에서 직접 로그를 가져와 선택한 컨테이너의 실시간 로그 또는 선택한 Pod 내 애플리케이션의 실시간 로그를 확인할 수 있습니다. 일반적으로 컨테이너 로그를 확인하기 위해서는 다음 명령어를 사용합니다.

```
kubectl logs pod/<Pod명> -n <Namespace명>
```

Pod 내에 2개 이상의 컨테이너가 존재한다면 개별 컨테이너를 선택해 로그를 확인합니다.

```
kubectl logs pod/<Pod명> -c <컨테이너명> -n <Namespace명>
```

로그를 확인하기 위해 매번 터미널 화면에 접근해 커맨드 라인 도구를 실행하는 것은 매우 불편합니다. 와탭 쿠버네티스 모니터링의 [로그](#) 기능을 사용하면 편리하게 현재 로그를 확인할 수 있습니다.

The screenshot shows a container management dashboard with a 'Container (38)' header. On the left, there's a summary table for containers and a list of container details. The main area on the right displays the log for a selected container, showing various system messages and network configuration steps.

Container	전체	Normal	Node	NS	Pod	Image
	38	38	4	5	34	5

Container	CpuByLimit	CpuByRequest	MemByLimit	NetRxBytes	NetTxBytes
	22.41%	131.01%	41.5%	12.7MiB	11.2MiB

ActiveTX	Normal	Slow	Very Slow	Total	TPS
	246	94	84	424	181.81


```

I0728 08:27:40.302383 1 main.go:211] CLI flags config: {etcdEndpoints:http://127.0.0.1:4001,http://127.0.0.1:2379
etcdPrefix:/coreos.com/network etcdKeyfile: etcdCertfile: etcdCAFile: etcdUsername: etcdPassword: version:false
kubeSubnetMgr:true kubeApiUrl: kubeAnnotationPrefix:flannel.alpha.coreos.com kubeConfigFile: iface:[] ifaceRegex:[] ipMasq:true
ifaceCanReach: subnetFile:/run/flannel/subnet.env publicIP: publicIPv6: subnetLeaseRenewMargin:60 healthzIP:0.0.0.0 healthzPort:0
iptablesResyncSeconds:5 iptablesForwardRules:true netConfPath:/etc/kube-flannel/net-conf.json setNodeNetworkUnavailable:true
useMultiClusterCidr:false}
W0728 08:27:40.303396 1 client_config.go:617] Neither --kubeconfig nor --master was specified. Using the inClusterConfig.
This might not work.
I0728 08:27:40.325316 1 kube.go:144] Waiting 10m0s for node controller to sync
I0728 08:27:40.325366 1 kube.go:485] Starting kube subnet manager
I0728 08:27:40.329182 1 kube.go:506] Creating the node lease for IPv4. This is the n.Spec.PodCIDRs: [10.244.0.0/24]
I0728 08:27:40.329217 1 kube.go:506] Creating the node lease for IPv4. This is the n.Spec.PodCIDRs: [10.244.2.0/24]
I0728 08:27:40.329223 1 kube.go:506] Creating the node lease for IPv4. This is the n.Spec.PodCIDRs: [10.244.3.0/24]
I0728 08:27:40.329223 1 kube.go:506] Creating the node lease for IPv4. This is the n.Spec.PodCIDRs: [10.244.4.0/24]
I0728 08:27:41.325472 1 kube.go:151] Node controller sync successful
I0728 08:27:41.325496 1 main.go:231] Created subnet manager: Kubernetes Subnet Manager - demo-k8s-worker-02
I0728 08:27:41.325502 1 main.go:234] Installing signal handlers
I0728 08:27:41.325692 1 main.go:542] Found network config - Backend type: vxlan
I0728 08:27:41.325721 1 match.go:206] Determining IP address of default interface
I0728 08:27:41.326079 1 vxlan.go:259] Using interface with name ens5 and address 10.21.1.157
I0728 08:27:41.326103 1 match.go:281] Defaulting external address to interface address (10.21.1.157)
I0728 08:27:41.326159 1 vxlan.go:140] VXLAN config: VNI=1 Port=0 GBP=false Learning=false DirectRouting=false
I0728 08:27:41.430283 1 kube.go:506] Creating the node lease for IPv4. This is the n.Spec.PodCIDRs: [10.244.5.0/24]
W0728 08:27:41.433723 1 main.go:595] no subnet found for key: FLANNEL_SUBNET in file: /run/flannel/subnet.env
I0728 08:27:41.433750 1 main.go:481] Current network or subnet (10.244.0.0/16, 10.244.5.0/24) is not equal to previous one
(0.0.0.0/0, 0.0.0.0/0), trying to recycle old iptables rules
I0728 08:27:41.451731 1 main.go:356] Setting up masking rules
I0728 08:27:41.451964 1 main.go:407] Changing default FORWARD chain policy to ACCEPT
I0728 08:27:41.454779 1 iptables.go:290] generated 7 rules
I0728 08:27:41.455461 1 main.go:435] Wrote subnet file to /run/flannel/subnet.env
I0728 08:27:41.455523 1 main.go:439] Running backend.
I0728 08:27:41.455001 1 iptables.go:290] generated 3 rules
I0728 08:27:41.457003 1 vxlan_network.go:64] watching for new subnet leases
I0728 08:27:41.457233 1 watch.go:51] Batch elem [0] is { subnet.Event(Type:0, Lease:subnet.Lease{EnableIPv4:true,
EnableIPv6:false, Subnet:ip.IP4Net{IP:0xaf40000, PrefixLen:0x18}, IPv6Subnet:ip.IP6Net{IP:(*ip.IP6)(nil), PrefixLen:0x0},
Attrs:subnet.LeaseAttrs{PublicIP:0xa150195, PublicIPv6:(*ip.IP6)(nil), BackendType:"vxlan", BackendData:json.RawMessage{0x7b,
0x22, 0x56, 0x4e, 0x49, 0x22, 0x3a, 0x31, 0x27, 0x22, 0x56, 0x74, 0x65, 0x70, 0x4d, 0x41, 0x43, 0x22, 0x3a, 0x22, 0x36, 0x65,

```

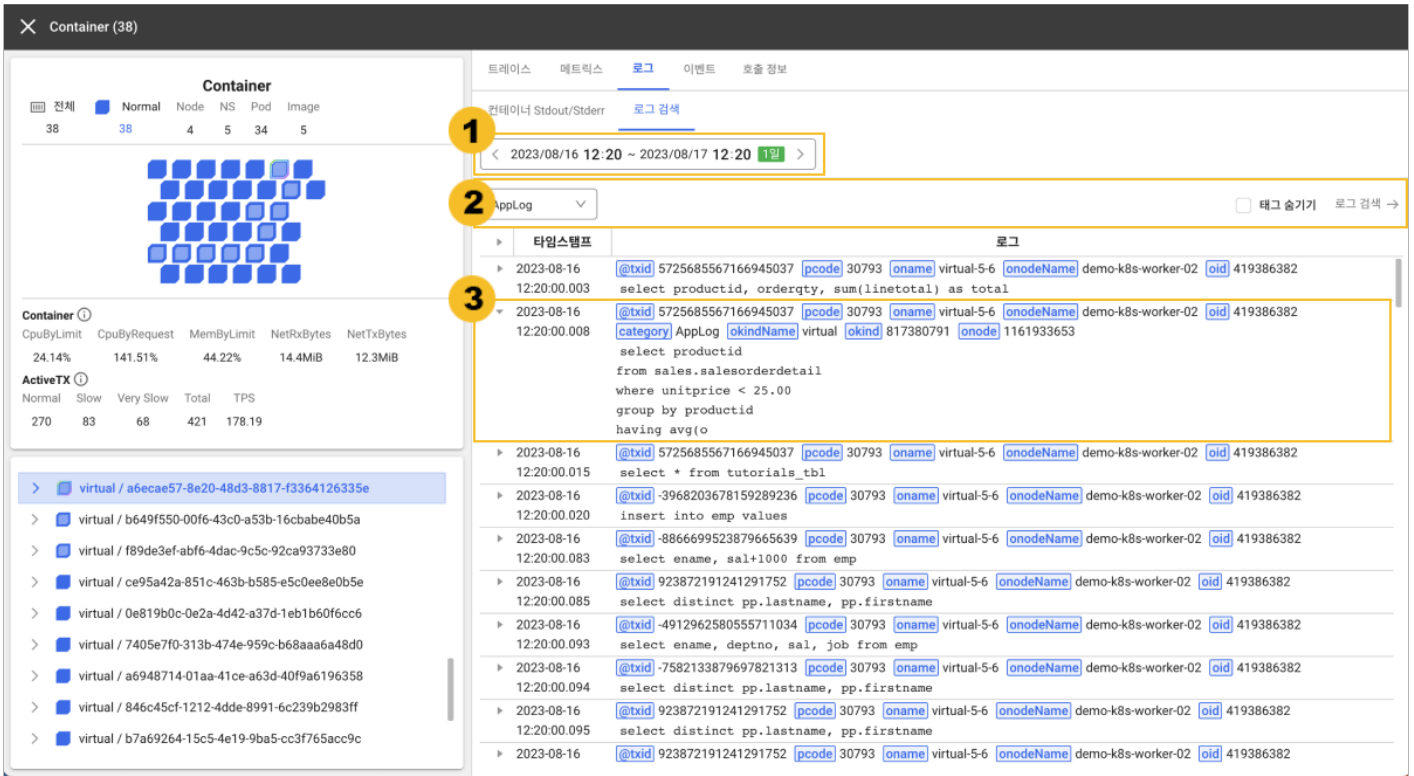
다만 생각해봐야 할 점이 있습니다. 우선 컨테이너를 종료하면 더 이상 로그를 확인할 수 없습니다. 또한 컨테이너가 계속 실행 중이더라도 일정 기간 또는 일정 용량의 로그만 확인할 수 있다는 단점이 있습니다. 와탭의 최근 로그 메시지([컨테이너 Stdout/Stderr](#)) 기능은 사용자 쿠버네티스 환경의 로그를 와탭을 통해 볼 수 있게 제공하는 기능으로 사용자 환경에서 사라진 로그는 와탭을 통해서도 확인하기 어렵습니다.

따라서 로그 데이터를 안정적이고 지속적으로 관리하기 위해서는 별도의 관리 방안이 필수적입니다. 특히 마이크로서비스 아키텍처 도입이 확산되면서 비즈니스를 구성하는 컨테이너 수량이 증가함에 따라 로그의 확인, 관리 방안은 더욱 복잡해지고 있습니다.

좋은 방법이 있을까요?

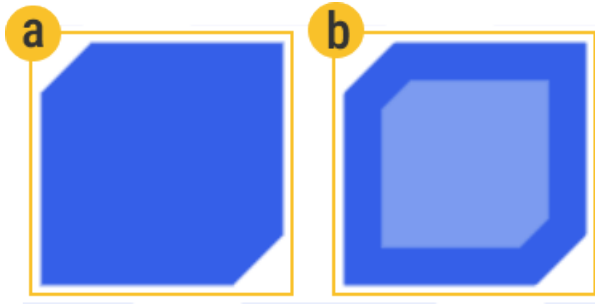
로그 검색

컨테이너 종료 시 관련 로그가 모두 사라지지만 [로그 검색](#) 기능을 통해 선택한 컨테이너 또는 Pod에서 과거에 발생한 로그를 확인할 수 있습니다. 컨테이너 Standard Out 로그 저장 시 원하는 만큼 저장 기간을 설정할 수 있고 특정 대상별 정보를 확인하거나 상호 연계 분석을 할 수 있습니다.



- 1 시간 선택자를 통해 데이터 조회를 원하는 시점을 선택할 수 있습니다.
- 2 왼쪽 카테고리 옵션을 통해 카테고리를 지정해 조회할 수 있습니다. 조회할 수 있는 카테고리에 대한 설명은 다음의 [블록](#)과 [그룹](#) 안내를 참조하세요.
- 2 오른쪽 [태그 숨기기](#)를 선택하면 로그 기록 중 태그 정보를 제외한 [타임스탬프](#)와 [메시지](#)를 확인할 수 있습니다.
- 2 오른쪽 [로그 검색](#) 버튼을 클릭하면 필터 기능 등을 통해 상세 조회할 수 있는 [로그](#) > [로그 검색](#) 메뉴로 이동합니다.
- 3 ▶ [더보기](#) 아이콘을 선택하면 해당 로그의 파싱된 키와 값 및 로그 메시지를 확인할 수 있습니다. 자세한 내용은 [다음 문서](#)를 참조하세요.

일반 블록과 마이크로 블록



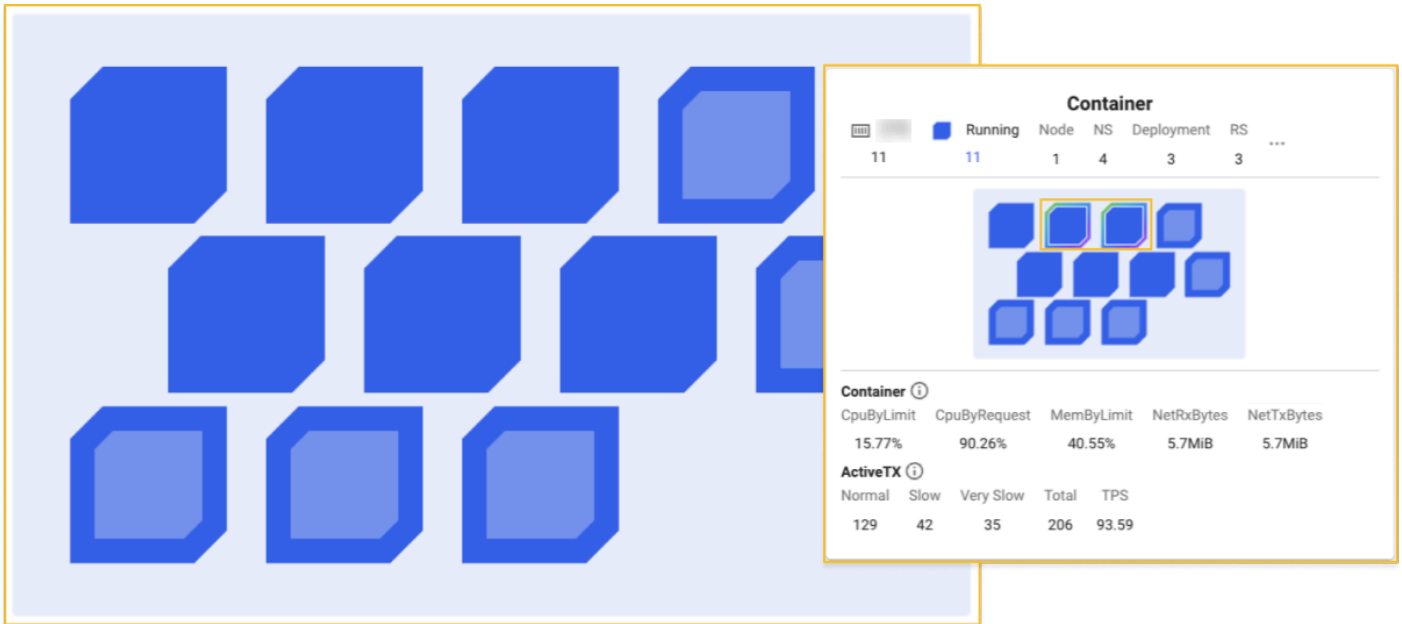
- **a 일반 블록**

컨테이너 맵의 다각형 하나를 블록이라고 합니다. 로그의 `containerStdout` 카테고리에서 `containerid` 가 일치하는 정보를 확인할 수 있습니다.

- **b 마이크로 블록**

에이전트가 설치된 블록을 마이크로 블록이라고 합니다. `containerStdout` 카테고리 뿐 아니라 `AppLog`, `AppStdOut`, `AppStdErr` 등 `App*` 관련 카테고리도 확인할 수 있습니다. 해당 카테고리를 선택할 경우 `microOid` (블록 내 에이전트 id)가 일치하는 로그 정보를 확인할 수 있습니다.

그룹



컨테이너 맵 화면에서 전체 블록을 선택한 경우 혹은 상세 보기 화면의 카드에서 블록을 다중 선택한 경우 해당 다중 블록을 그룹이라고 합니다. 그룹 선택 시 로그 검색 탭에서 `containerStdout` 카테고리 와 `App*` 카테고리의 모든 로그를 확인할 수 있습니다.

Standard Out 저장을 위한 노드 에이전트 설정 추가

와탭 로그 서비스를 통해 컨테이너의 Standard Output을 저장하기 위해서는 와탭 쿠버네티스 노드 에이전트에 `logsink_enabled=true` 옵션을 설정해야 합니다. 설정 및 와탭 로그 모니터링 서비스에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

컨테이너 내부 애플리케이션 로그 수집을 위한 애플리케이션 에이전트 설정 추가

와탭 로그 서비스를 통해 컨테이너 상에서 실행되는 애플리케이션의 로그를 수집하려면 [다음 문서](#)를 참조하세요. Java, Python, Go를 지원합니다.

⚠ 와탭 로그 서비스를 활성화하고 컨테이너의 Standard Output 로그 저장 기능을 사용하면 그에 따른 추가 비용이 청구됩니다. 관련한 요금에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

ⓘ 일부 구 버전 에이전트의 경우는 컨테이너 Standard Output 로그 저장 기능을 제공하지 않습니다. 에이전트 업그레이드

ⓘ 방법은 다음과 같습니다.

Agent update

```
kubectl set image daemonset/whatap-node-agent whatap-node-agent=whatap/kube_mon:<에이전트버전> -n whatap-monitoring
```

와탭 쿠버네티스 에이전트 버전은 [다음 문서](#)를 통해 확인할 수 있습니다.

Event

홈 화면 > 프로젝트 선택 > 대시보드 > 컨테이너 맵 > 대상 선택 후 상세 보기 화면 진입 > 이벤트 탭 선택

리소스 수명 주기 동안 발생하는 모든 이벤트에 대한 기록을 제공합니다. 와탭 쿠버네티스 모니터링에서 제공하는 이벤트 정보는 다음과 같습니다.

- 쿠버네티스 이벤트(Kubernetes)
- 와탭 이벤트(WhaTap)

기본 화면 안내

The screenshot displays the 'Container (38)' monitoring dashboard. The interface is divided into several sections:

- Container Overview:** Shows a grid of 38 containers, with 38 in 'Normal' status. Below the grid, there are summary statistics for CPU, memory, and network usage.
- Event Filter:** A dropdown menu (1) allows selecting the event source, with 'Container 이벤트' currently selected.
- Time Range:** A date and time selector (2) is set to '2023/08/16 13:34 ~ 2023/08/17 13:34'.
- Event Count Chart:** A bar chart (3) shows the number of events per hour over the selected time period.
- Event Log:** A table (4) lists individual events with columns for timestamp, status, namespace, pod name, and pod name. The log shows multiple 'DISABLED' events for 'whatap-virtual-deployment-76c678b7d7-rvr4h'.

- 1 쿠버네티스 이벤트(Kubernetes) 또는 와탭 이벤트(WhaTap)를 선택해 조회할 수 있습니다.
 - 전체 또는 다중 블록 선택

컨테이너 맵 및 왼쪽의 카드에서 전체 또는 다중 블록 선택 시 **Kubernetes 전체 이벤트** 또는 **WhaTap 전체 이벤트**를 조회할 수 있습니다.

- 클러스터링 또는 개별 블록 선택

컨테이너 맵 상단의 **2단계** 그룹화 선택 또는 개별 블록 선택 시 해당 이벤트를 조회할 수 있습니다. **그룹화(클러스터링)**에 관한 자세한 내용은 [다음 문서](#)를 참조하세요.

❗ **상세 보기** 화면에 진입하기 전 **컨테이너 맵** 상단에서 선택한 뷰 및 대상 그룹화에 따라 조회할 수 있는 이벤트가 달라집니다.

- **2** 시간 선택자를 통해 이벤트 기록을 조회하고자 하는 시간대를 지정할 수 있습니다. 기본 설정은 **1일**입니다.
- **3** 선택한 시간대의 **이벤트 수 추이**를 바 차트를 통해 살펴볼 수 있습니다.
- **4** 목록 형식으로 이벤트 기록 정보를 확인할 수 있습니다.
 - 이벤트 기록 목록 오른쪽 상단의 **태그 숨기기**를 선택하면 이벤트 기록 중 태그 정보를 제외한 **타임스탬프**와 **메시지**를 확인할 수 있습니다.
 - **Kubernetes** 이벤트
 - **Normal** 또는 **Warning** 중 조회를 원하는 이벤트 타입을 선택해 확인할 수 있습니다. 이 이벤트 타입은 쿠버네티스가 제공하는 이벤트 타입 정보와 동일합니다.
 - 각 이벤트의 왼쪽 띠 역시 이벤트 타입을 나타냅니다. 회색은 **Normal**, 주황색은 **Warning** 상태를 의미합니다.
 - **WhaTap** 이벤트
 - **Critical**, **Info**, **Warning** 중 조회를 원하는 이벤트 레벨을 선택해 확인할 수 있습니다.
 - 각 이벤트의 왼쪽 띠 역시 이벤트 레벨을 나타냅니다. 회색은 **Info**, 주황색은 **Warning**, 빨간색은 **Critical** 상태를 의미합니다.

쿠버네티스 이벤트

Kubernetes 이벤트는 쿠버네티스 자체적으로 제공하는 이벤트 정보입니다. 사용자 쿠버네티스 환경에서 `kubectrl get events` 명령을 사용하면 사용자 쿠버네티스 클러스터의 이벤트 정보를 확인할 수 있습니다.

```
$ kubectrl get events -A
```

NAMESPACE	LAST SEEN	TYPE	REASON	OBJECT	MESSAGE
community baekdusan-59554d4859	15m	Normal	ScalingReplicaSet	deployment/baekdusan	Scaled up replica set
community suraksan-66fc4bf889	12m	Normal	ScalingReplicaSet	deployment/suraksan	Scaled down replica set
community baekdusan-59554d4859	5m18s	Normal	ScalingReplicaSet	deployment/baekdusan	Scaled down replica set
community	7m10s	Normal	Created	pod/suraksan-66fc4bf889-h2cw8	Created container suraksan
community	6m56s	Normal	Pulled	pod/suraksan-66fc4bf889-h2cw8	Container image "123456789123.dkr.ecr.ap-northeast-2.amazonaws.com/suraksan:0.48" already present on machine
community	7m9s	Normal	Started	pod/suraksan-66fc4bf889-h2cw8	Started container suraksan
community	2m7s	Warning	BackOff	pod/suraksan-66fc4bf889-h2cw8	Back-off restarting failed container
community	7m12s	Normal	Started	pod/suraksan-66fc4bf889-t27rp	Started container suraksan
community	6m54s	Normal	Pulled	pod/suraksan-66fc4bf889-t27rp	Container image "123456789123.dkr.ecr.ap-northeast-2.amazonaws.com/suraksan:0.48" already present on machine
community	7m12s	Normal	Created	pod/suraksan-66fc4bf889-t27rp	Created container suraksan
community	2m46s	Warning	BackOff	pod/suraksan-66fc4bf889-t27rp	Back-off restarting failed container
community mountain-6795bfb54	7m19s	Normal	ScalingReplicaSet	deployment/mountain	Scaled up replica set
community mountain-6795bfb54	27m	Normal	ScalingReplicaSet	deployment/mountain	Scaled down replica set
community	6m53s	Normal	Pulled	pod/suraksan-66fc4bf889-mjbsj	Container image "123456789123.dkr.ecr.ap-northeast-2.amazonaws.com/suraksan:0.48" already present on machine
community	7m10s	Normal	Created	pod/suraksan-66fc4bf889-mjbsj	Created container suraksan
community	7m10s	Normal	Started	pod/suraksan-66fc4bf889-mjbsj	Started container suraksan
community	2m10s	Warning	BackOff	pod/suraksan-66fc4bf889-mjbsj	Back-off restarting failed container
community community/baekdusan-svc	5m18s	Warning	FailedToUpdateEndpoint	endpoints/baekdusan-svc	Failed to update endpoint Operation cannot be fulfilled on endpoints "baekdusan-svc": the object has been modified; please apply your changes to the latest version and try again

하지만 이러한 이벤트 정보는 지속적으로 저장되지 않습니다. 따라서 일정 시간이 지난 이벤트는 더 이상 정보를 확인할 수 없습니다.

와탭 쿠버네티스는 사용자 클러스터 환경에서 발생한 쿠버네티스 이벤트 정보를 모두 수집하여 저장합니다. 그리고 사용자가 확인하고자 하는 대상에 따라 Pod, ReplicaSet, Deployment, Namespace 등의 이벤트 정보를 제공합니다.

각 이벤트의 왼쪽 띠는 이벤트 타입을 나타냅니다. 회색은 Normal, 주황색은 Warning 상태를 의미합니다. 사용자 선택 대상에 따른 제공 쿠버네티스 이벤트는 다음과 같습니다.

선택 대상	제공 이벤트	제공 연관 이벤트
전체 (All)	프로젝트 전체 이벤트	
Namespace	Namespace 하위 전체 이벤트	
Node	해당 Node 이벤트 또는 노드 내 Pod (컨테이너) 전체 이벤트	
Deployment	① kind가 Deployment ② name이 사용자 Deployment 이름과 일치	name이 사용자 Deployment 이름으로 시작
ReplicaSet	① kind가 ReplicaSet	name이 사용자 ReplicaSet 이름으로 시작

선택 대상	제공 이벤트	제공 연관 이벤트
	② name이 사용자 ReplicaSet 이름과 일치	
Pod (컨테이너)	① kind가 Pod ② name이 사용자 Pod 이름과 일치	해당 Pod가 소속된 Deployment 이름으로 시작 또는 ReplicaSet 이름으로 시작

와탭에 저장된 쿠버네티스 이벤트는 `message`, `namespace`, `kind`, `name`, `reason`, `type` 등의 항목으로 구성되어 있습니다.

와탭 이벤트

WhaTap 이벤트는 와탭의 이벤트 설정을 통해 등록된 이벤트 조건에 따라 발생한 이벤트입니다. [컨테이너 맵](#) 상세 보기 화면의 [이벤트](#) 탭에서 사용자의 와탭 경고 설정에 따른 이벤트 목록을 확인할 수 있습니다.

The screenshot displays the WhaTap interface for monitoring containers. On the left, a 'Container' overview shows 38 containers in a 'Running' state across 4 nodes. Below this, resource usage metrics are listed: CpuByLimit (27.94%), CpuByRequest (164.87%), MemByLimit (44.92%), NetRxBytes (16.2MIB), and NetTxBytes (13.9MIB). A table of 'ActiveTX' shows 317 Normal, 116 Slow, and 68 Very Slow transactions, with a total of 501 and a TPS of 206.8. A list of 10 virtual containers is shown with their IDs.

On the right, the '이벤트' (Event) section is active. It shows a time range from 2023/08/16 14:04 to 2023/08/17 14:04. A bar chart titled '이벤트 수 추이' (Event Count Trend) shows a steady flow of events. Below the chart, a table of event logs is displayed, with columns for '타임스탬프' (Timestamp) and '로그' (Log). The logs show a series of 'DISABLED' events from the 'virtual-main' namespace, triggered by 'test' actions in the 'whatap-virtual-deployment' pods.

각 이벤트의 왼쪽 띠는 이벤트 레벨을 나타냅니다. 회색은 Info, 주황색은 Warning, 빨간색은 Critical 상태를 의미합니다. 사용자 선택 대상에 따른 제공 와탭 이벤트는 다음과 같습니다.

선택 대상	제공 이벤트
전체(All)	프로젝트 전체 이벤트
Namespace	Namespace 하위 전체 이벤트
Node	onodeName이 사용자 Node 이름과 일치
Deployment	deployment가 Deployment 이름과 일치
ReplicaSet	replicaSetName이 사용자 ReplicaSet 이름과 일치
Pod	podName이 사용자 Pod 이름과 일치
Container	containerId가 사용자 Container ID와 일치

와탭 이벤트는 `message`, `status`, `title`, `level` 등의 항목으로 구성되어 있습니다. 다음 예시 화면은 [WhaTap Containter 이벤트](#) 기록을 조회하고 있습니다.

The screenshot displays the 'Container (38)' monitoring page. On the left, a 'Container' summary card shows a grid of 38 containers, with one highlighted. Below it, resource usage for 'virtual' containers is shown: CpuByLimit (26.36%), CpuByRequest (155.38%), MemByLimit (45.1%), NetRxBytes (15.3MiB), and NetTxBytes (13MiB). The 'ActiveTX' table shows 262 Normal, 91 Slow, and 162 Very Slow transactions.

The main area shows the '이벤트' (Event) tab. A time range selector is set to '2023/08/16 14:10 ~ 2023/08/17 14:10' with a '1일' (1 day) filter. A bar chart titled '이벤트 수 추이' (Event Count Trend) shows event counts over time, with a peak of 2 events on 08/17 at 22:30:00. Below the chart is a log table with columns for timestamp, status, namespace, podName, containerId, nodeName, title, and level. One event is highlighted as 'Critical'.

- ① 상세 보기 왼쪽 요약 정보 카드에서 조회하려는 대상을 선택하세요.
- ② 요약 정보 목록에서 선택 대상의 정보를 확인하세요.
- ③ 이벤트 탭에서 WhaTap 이벤트를 선택하세요.
 상세 보기 화면에 진입하기 전 컨테이너 맵 상단 뷰 옵션을 컨테이너 뷰로 지정해 Container 이벤트를 조회할 수 있습니다.
- ④ 시간 선택자를 통해 조회하려는 시간을 선택하세요. 기본 설정은 1일 입니다.
- ⑤ 바 차트를 통해 지정한 시간대의 이벤트 수 추이를 살펴보세요.
 지정 시간대 기준으로 이벤트 수 추이를 살펴보면 이벤트 2개가 카운트된 것을 확인할 수 있습니다.
- ⑥ 이벤트 기록 목록에서 이벤트 기록 정보를 확인하세요. 목록 왼쪽 띠를 통해 이벤트별 레벨을 확인할 수 있습니다. 이벤트 메시지 및 태그 전체를 확인하려면 ▼ 아이콘을 선택하세요.
 이벤트 기록 목록에서 컨테이너 Limit 기준 CPU 사용률이 85% 이상, GC 횟수 이상 Warning 레벨 이벤트를 2개를 조회할 수 있습니다.

Call Information

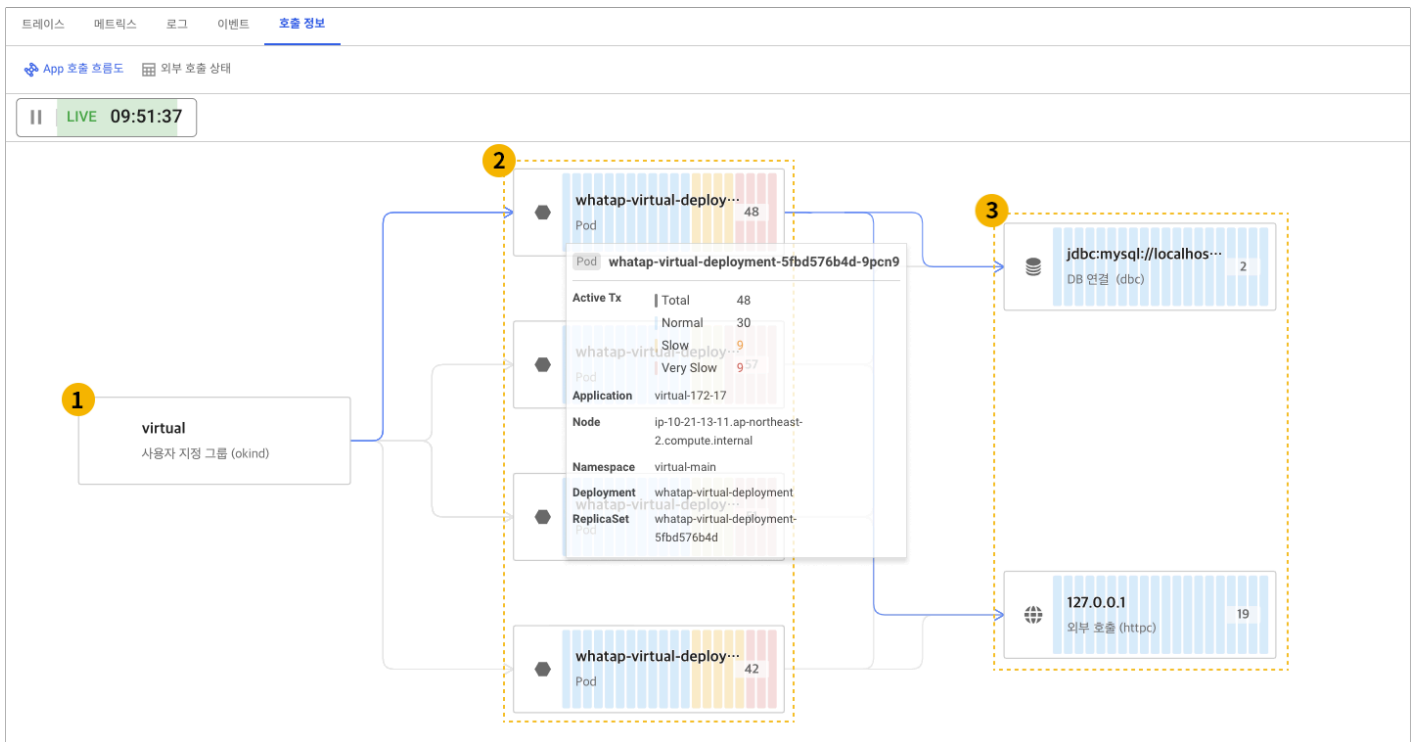
❗ 외부 호출 상태 조회 기능은 애플리케이션이 Java인 경우만 제공합니다.

홈 화면 > 프로젝트 선택 > 대시보드 > 컨테이너 맵 > 대상 선택 후 상세 보기 화면 진입 > 호출 정보 탭 선택

쿠버네티스 컨테이너 상에서 실행 중인 애플리케이션의 호출 흐름과 외부 호출 대상의 상태를 확인할 수 있습니다. 와탭 쿠버네티스 모니터링에서 제공하는 호출 정보는 다음과 같습니다.

- App 호출 흐름도
- 외부 호출 상태

App 호출 흐름도



Pod 내부에서 실행 중인 컨테이너 애플리케이션의 호출 흐름을 표현한 차트를 제공합니다. 사용자 지정 그룹(Okind), Pod, DB 연결(DBC) 및 외부 호출(HTTPC) 세 가지의 노드간 연결 관계를 확인할 수 있습니다. 각 노드는 다음과 같은 정보를 제공합니다.

1 사용자 지정 그룹(okind)

사용자가 지정한 그룹 명칭을 의미합니다. 그룹(okind)을 지정하지 않은 경우 해당 노드는 표시되지 않습니다.

2 Pod

컨테이너 애플리케이션이 속해 있는 Pod 명칭을 의미합니다. 컨테이너 애플리케이션 내 액티브 트랜잭션 총 개수와 처리 속도를 확인할 수 있습니다. 선택 시 [액티브 트랜잭션](#) 메뉴로 이동합니다.

- **Active Tx**: 컨테이너 애플리케이션 내 액티브 트랜잭션을 의미합니다.
 - **Total**: 컨테이너 애플리케이션 내 총 액티브 트랜잭션의 개수를 의미합니다.
 - **Normal**: 0~3초 내 처리되는 트랜잭션의 개수를 의미합니다.
 - **Slow**: 3~8초 내 처리되는 트랜잭션의 개수를 의미합니다.
 - **Very Slow**: 처리에 8초 이상 소요되는 트랜잭션의 개수를 의미합니다.
- **Application**: Pod 내부의 Application 명칭을 의미합니다.
- **Node**: Pod가 속한 Node 명칭을 의미합니다.
- **Namespace**: Pod가 속한 Namespace 명칭을 의미합니다.
- **ReplicaSet**: Pod가 속한 ReplicaSet 명칭을 의미합니다.
- **Deployment**: Pod가 속한 Deployment 명칭을 의미합니다.

3 DB 연결 및 외부 호출

- **DB 연결(dbc)**: Pod 내부의 컨테이너 애플리케이션이 호출한 DB Connection을 의미합니다. 선택 시 [외부 호출 상태](#) 조회 탭으로 이동합니다.
 - **Active Tx**: DB Connection 중인 컨테이너 애플리케이션 내 액티브 트랜잭션을 의미합니다.
 - **Total**: 컨테이너 애플리케이션 내 총 액티브 트랜잭션의 개수를 의미합니다.
 - **Normal**: 0~3초 내 처리되는 트랜잭션의 개수를 의미합니다.
 - **Slow**: 3~8초 내 처리되는 트랜잭션의 개수를 의미합니다.
 - **Very Slow**: 처리에 8초 이상 소요되는 트랜잭션의 개수를 의미합니다.
- **외부 호출(httpc)**: Pod 내부의 컨테이너 애플리케이션이 호출한 외부 API를 의미합니다. 선택 시 [외부 호출 상태](#) 조회 탭으로 이동합니다.
 - **Active Tx**: 외부 API 호출 중인 컨테이너 애플리케이션 내 액티브 트랜잭션을 의미합니다.

- **Total**: 컨테이너 애플리케이션 내 총 액티브 트랜잭션의 개수를 의미합니다.
- **Normal**: 0~3초 내 처리되는 트랜잭션의 개수를 의미합니다.
- **Slow**: 3~8초 내 처리되는 트랜잭션의 개수를 의미합니다.
- **Very Slow**: 처리에 8초 이상 소요되는 트랜잭션의 개수를 의미합니다.

외부 호출 상태

외부 호출 로그를 이용해 모니터링 대상이 호출하는 목적지의 연결 상태 정보를 기반으로 외부 호출 대상의 정상 여부 정보를 제공합니다. **외부 호출 상태** 조회 기능을 통해 접속 불가 대상의 가시성을 제공합니다. 이를 통해 서비스 이슈 발생 시 문제 발생 지점을 용이하게 파악할 수 있습니다. 또한 지속적인 접속 실패에 의한 스레드와 커넥션 증가 등의 자원 잠식 현상을 사전에 감지하고 이에 대응할 수 있습니다.

ⓘ 외부 호출 로그

애플리케이션 에이전트 설정을 통해 각 모니터링 대상별 외부 호출에 대한 성공 및 실패 여부를 기록한 로그입니다.

- 외부 호출 유형은 다음과 같습니다.

httpcall , db , socket

- 외부 호출 로그를 수집하려면 [에이전트 설정](#)에 다음 옵션을 추가해야 합니다.

```
logsink_rt_enabled=true
logsink_enabled=true
```

트레이스 메트릭스 로그 이벤트 **호출 정보**

App 호출 흐름도 외부 호출 상태

LIVE 10:54:21

1 에이전트 업무 Search **3** 마지막 호출 상태

2 외부 호출 로그

시간	에이전트	로그
09-27 10:39:13.706	tour-110-1...	target tour-svc.aeug.svc.cluster.local content ok
09-27 10:39:20.967	customer...	target jdbc:mariadb://cms-mariadb.titans.svc.cluster.local:3306/customer content ok
09-27 10:39:28.246	tour-110-1...	target catserver-svc.aeug.svc.cluster.local content ok
09-27 10:39:30.323	tour-110-1...	target jdbc:postgresql://mall-db-svc.titans.svc.cluster.local:5432/malldb content ok
09-27 10:39:36.158	tour-110-1...	target customer-svc.titans.svc.cluster.local content ok
09-27 10:39:50.216	tour-110-1...	target tour-svc.aeug.svc.cluster.local content ok
09-27 10:39:56.007	customer...	target jdbc:mariadb://cms-mariadb.titans.svc.cluster.local:3306/customer content ok
09-27 10:40:00.601	tour-110-1...	target jdbc:postgresql://mall-db-svc.titans.svc.cluster.local:5432/malldb content ok
09-27 10:40:01.649	tour-110-1...	target catserver-svc.aeug.svc.cluster.local content ok
09-27 10:42:11.575	customer...	target jdbc:mariadb://cms-mariadb.titans.svc.cluster.local:3306/customer content ok
09-27 10:42:11.486	tour-110-1...	target jdbc:postgresql://mall-db-svc.titans.svc.cluster.local:5432/malldb content ok
09-27 10:42:13.089	tour-110-1...	target 172.20.134.67 content connect timed out
09-27 10:42:13.089	tour-110-1...	target catserver-svc.aeug.svc.cluster.local content connect timed out exception SocketTimeoutException
09-27 10:42:14.961	tour-110-1...	target tour-svc.aeug.svc.cluster.local content ok
09-27 10:42:20.066	tour-110-1...	target 172.20.134.67 content connect timed out
09-27 10:42:20.067	tour-110-1...	target catserver-svc.aeug.svc.cluster.local content connect timed out exception SocketTimeoutException
09-27 10:42:25.995	tour-110-1...	target 172.20.134.67 content connect timed out

마지막 호출 상태

- jdbc:postgresql://mall-db-svc.titans.svc.cl...
- tour-svc.aeug.svc.cluster.local
- catserver-svc.aeug.svc.cluster.local
- jdbc:mariadb://cms-mariadb.titans.svc.clu...
- customer-svc.titans.svc.cluster.local

마지막 호출 상태

- jdbc:postgresql://mall-db-svc.titans.svc.clus...
- tour-svc.aeug.svc.cluster.local
- catserver-svc.aeug.svc.cluster.local
- jdbc:mariadb://cms-mariadb.titans.svc.clust...
- customer-svc.titans.svc.cluster.local

외부 호출 상태 탭에서 외부 호출 로그를 목록 형식으로 조회할 수 있습니다. 최근 24시간 범위의 로그를 최대 500개 불러와 표시합니다. 실시간 조회 시 데이터는 3초 간격으로 업데이트됩니다.

1 상단 바 옵션

- 에이전트 또는 업무 컬럼의 표시 여부를 선택할 수 있습니다.
- 검색어 입력창을 통해 해당 검색어가 속한 에이전트, 업무, 로그가 포함된 열을 필터링할 수 있습니다.
- ▶ **재생**: 로그가 업데이트 되었을 시 최하단으로 이동해 가장 최근 로그를 확인할 수 있습니다.
- || **일시 정지**: 자동 스크롤 이동을 일시적으로 중지합니다.
- ▲ ▼: 최상단 또는 최하단으로 이동합니다.

2 외부 호출 로그 목록

아이콘과 태그를 통해 외부 호출 로그의 주요 내용을 쉽게 확인할 수 있습니다. 에러 로그의 경우 빨간색으로 표시됩니다.

- 외부 호출 유형은 아이콘 (db), (httpcall), (socket)을 통해 안내합니다.

- `target` : 외부 호출 및 연결 대상을 의미합니다.
- `content` : 로그의 내용을 의미합니다.
- `exception` : 에러 발생 시 기록된 로그의 내용을 의미합니다.

3 마지막 호출 상태

조회된 모든 로그의 외부 호출 대상 목록을 확인합니다. 가장 최근 연결 상태에 따라 **성공** 또는 **실패**로 표시됩니다.

- 조회 모드
 - **List**: 기본 목록 뷰입니다. 전체 목록의 길이가 화면 높이를 넘어갈 경우 스크롤이 생깁니다.
 - **Map**: 목록이 화면을 넘어가게 되면 목록의 길이를 줄여 가로 방향으로 외부 호출 대상을 나열합니다.

• 필터

외부 호출 대상을 선택 시 해당 외부 호출 대상에 대한 로그만 필터링해 외부 호출 로그 목록을 조회할 수 있습니다. 재선택 시 해제됩니다.

ⓘ 데이터가 없는 경우

실시간 조회 시 최근 24시간 동안 외부 호출 로그가 수집되었는지 확인합니다. 24시간 동안 외부 호출 로그가 수집되지 않은 경우 다음과 같은 안내 메시지를 제공합니다.



현재 이 프로젝트에서 최근 24시간 내에 외부(remote) 호출 관련 로그를 찾을 수 없습니다.

Remote Access Status는 애플리케이션 에이전트의 설정을 통해, 각 모니터링 대상별 외부(remote) 호출에 대한 성공/실패 여부를 기록한 로그를 시각화 합니다.

- 본 기능은 **애플리케이션이 Java인 경우만 지원**합니다.
- 본 기능을 사용하기 위해서 **[관리] > [에이전트 설정]** 메뉴에서 아래의 에이전트 설정을 추가하세요.

- logsink_rt_enabled=true
- logsink_enabled=true

설정하러 가기 →

퍼포먼스 요약

홈 화면 > 프로젝트 선택 > [대시보드](#) > [퍼포먼스 요약](#)

최근 시간별 차트를 통하여 컨테이너/Pod 수량, 노드 수량 등 쿠버네티스 주요 지표들에 어떤 변화가 있었는지 확인할 수 있습니다. 다음 이미지는 클러스터 프로젝트 ([Kubernetes](#)) 예시입니다.

CP 퍼포먼스 요약

LIVE 16:01:40 5분

1 쿠버네티스 클러스터 통합 퍼포먼스 (와탭 노드 에이전트 설치 여부와 무관한 클러스터 전체 정보)

대상 프로젝트:

쿠버네티스 클러스터 전체 컨테이너 수	쿠버네티스 클러스터 전체 Pod 수
Running Containers: 50 Waiting Containers: 1 Stopped Containers: 0	Available Pods: 41 Unavailable Pods: 0 Desired Pods: 41

쿠버네티스 클러스터 전체 컨테이너 수 추이 | 쿠버네티스 클러스터 전체 Pod 수 추이 | 쿠버네티스 클러스터 전체 노드 수 (Nodes: 2)

2 프로젝트 퍼포먼스 (프로젝트 내 와탭 노드 에이전트 및 애플리케이션 에이전트가 설치된 대상의 성능 정보)

에이전트 설치 노드에서 실행 중인 컨테이너 수 Total: 46 Running: 45 Waiting: 1	에이전트 설치 노드에서 실행 중인 Pod 수 Total: 37 Running: 37	에이전트 설치 노드 수 Total: 2 Ready: 2
---	--	--------------------------------------

Total Container CPU Rec: 1.92cores	Total Container CPU Use: 0.38cores	Total Container Memory: 2.6GiB	Total Container Memory: 4GiB	액티브 트랜잭션 수: 17	TPS: [Graph]
------------------------------------	------------------------------------	--------------------------------	------------------------------	----------------	--------------

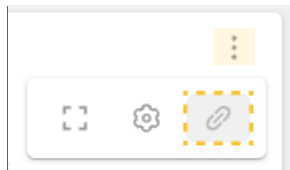
Node CPU (%) [Graph]	Node Memory (%) [Graph]	ResponseTime [Graph]	TrxCnt [Graph]
----------------------	-------------------------	----------------------	----------------

Container CPU Usage by Request (%) [Graph]	Container CPU Usage by Node (%) [Graph]	Container CPU Throttling Time [Graph]
--	---	---------------------------------------

Container Memory Working Set by Request (%) [Graph]	Container Memory Failure Count [Graph]	Container Network Rece [Graph]	Container Network Tran [Graph]
---	--	--------------------------------	--------------------------------

• 바로 가기 옵션

위젯별 오른쪽 상단에 위치한 **더보기** 아이콘을 선택하면 다음과 같이 **바로 가기** 옵션을 확인할 수 있습니다.



• 라인 차트 설정 옵션

라인 차트 위젯별 오른쪽 상단에 위치한 **더보기** 아이콘을 선택하면 **설정** 옵션을 확인할 수 있습니다. 라인 차트의 경우 설정 옵션을 통해 id가 되는 값을 변경할 수 있습니다. 해당 설정은 저장되지 않습니다.

예, **Node CPU (%)** 위젯의 경우 **Node**별 데이터 조회가 가능합니다. 메트릭스를 변경해 **ContainerName**별로 조회할 수 있습니다.

쿠버네티스 클러스터 통합 퍼포먼스

클러스터 프로젝트 (**Kubernetes**)의 경우 **1 쿠버네티스 클러스터 통합 퍼포먼스** 영역에서 와탭 노드 에이전트 설치 여부와 관계없이 클러스터 전체 정보를 표시합니다. 네임스페이스 프로젝트로 분리된 정보도 통합해 표시합니다.

쿠버네티스 네임스페이스 퍼포먼스

네임스페이스 프로젝트 (**Namespace**)의 경우 **1 쿠버네티스 네임스페이스 퍼포먼스** 영역에서 와탭 노드 에이전트 설치 여부와 관계없이 네임스페이스 정보를 표시합니다. 네임스페이스와 연관성이 있는 컨테이너, Pod, 노드의 정보를 표시합니다.

프로젝트 퍼포먼스

2 프로젝트 퍼포먼스 영역은 프로젝트 내의 와탭 노드 에이전트 및 애플리케이션 에이전트가 설치된 대상의 정보를 표시합니다. 네임스페이스 프로젝트로 분리된 정보는 네임스페이스 프로젝트에서 확인할 수 있습니다.

ⓘ 에이전트 설치가 정상적으로 완료되지 않은 경우 데이터가 표시되지 않을 수 있습니다.

노드 정보

위젯으로 제공하는 노드 정보는 다음과 같습니다.

- 최근 30분 차트
 - [Node CPU \(%\)](#), [Node Memory \(%\)](#) 등

- ① • 노드 수량 위젯 오른쪽 위에 : [더보기](#) 아이콘 > [바로 가기](#) 아이콘을 차례로 선택하면 [클러스터](#) > [노드 목록](#) 메뉴로 이동합니다.
- [Node CPU \(%\)](#) 위젯과 [Node Memory \(%\)](#) 위젯 오른쪽 위에 : [더보기](#) 아이콘 > [바로 가기](#) 아이콘을 차례로 선택하면 [분석](#) > [메트릭스 차트](#) 메뉴로 이동합니다. 해당 메뉴를 사용하면 더욱 자세한 조건을 통해 차트를 확인할 수 있습니다.

컨테이너 정보

위젯으로 제공하는 컨테이너 정보는 다음과 같습니다.

- 최근 10분 차트
 - [Container Status](#), [Container CPU \(%\)](#), [Container Memory \(%\)](#) 등

애플리케이션 정보

위젯으로 제공하는 애플리케이션 정보는 다음과 같습니다.

- 최근 10분 차트
 - [Active TX](#), [TPS](#), [Response Time](#) 등

- ① 애플리케이션 정보는 애플리케이션 에이전트 설치 대상이 존재할 때 표시합니다.

클러스터 요약

홈 화면 > 프로젝트 선택 > [대시보드](#) > [클러스터 요약](#)

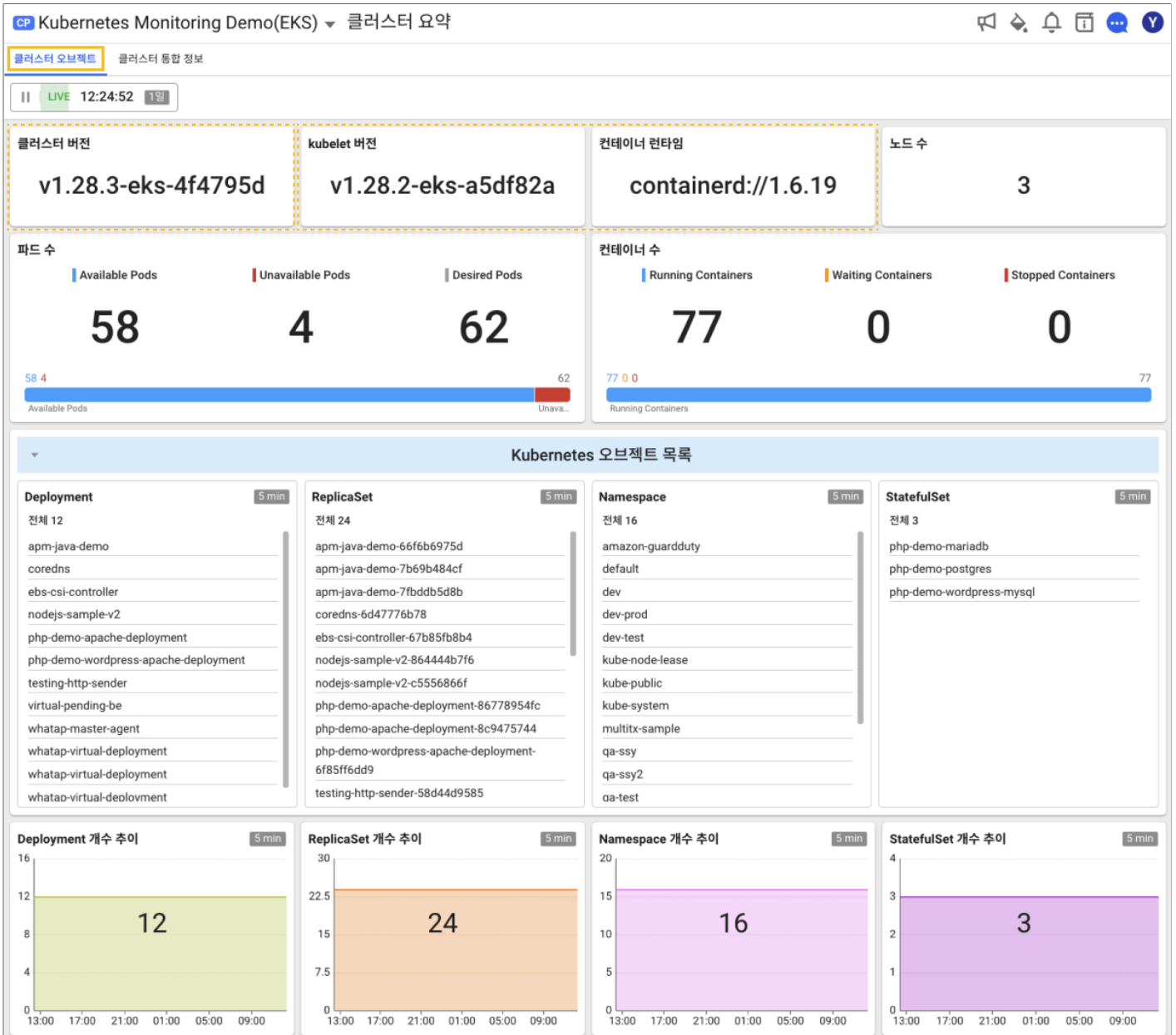
클러스터 관련 요약 정보를 확인할 수 있습니다. 네임스페이스 프로젝트로 분리된 정보를 통합해 확인할 수 있습니다.

ⓘ 클러스터 프로젝트인 경우에만 [클러스터 요약](#) 메뉴를 확인할 수 있습니다.

클러스터 오브젝트

클러스터와 관련된 버전 정보와 클러스터 내부의 노드, Pod, 컨테이너 등 쿠버네티스 오브젝트의 개수를 확인할 수 있습니다.

- ⓘ • 쿠버네티스 에이전트 1.4.1 버전 이상일 때 [클러스터 버전](#)을 확인할 수 있습니다.
- 쿠버네티스 에이전트 1.3.6 버전 이상일 때 다음 사항을 확인할 수 있습니다.
 - [kubelet 버전](#)
 - [컨테이너 런타임](#)



클러스터 버전

- 5초 주기의 데이터입니다. 실시간 데이터 조회 시 5초에 한 번씩 데이터를 호출합니다.

kubelet 버전

- 5초 주기의 데이터입니다. 실시간 데이터 조회 시 5초에 한 번씩 데이터를 호출합니다.

컨테이너 런타임

- 5초 주기의 데이터입니다. 실시간 데이터 조회 시 5초에 한 번씩 데이터를 호출합니다.

노드 수

- 5초 주기의 데이터입니다. 실시간 데이터 조회 시 5초에 한 번씩 데이터를 호출합니다.

Pod 수

- 5초 주기의 데이터입니다. 실시간 데이터 조회 시 5초에 한 번씩 데이터를 호출합니다.
- **Available Pods**: 노드에 할당된 Pod들의 개수를 의미합니다.
- **Unavailable Pods**: 노드에 할당되지 못한 Pod 개수를 의미합니다.
- **Desired Pods**: 의도한 Pod 수를 의미합니다.

| Available Pods + Unavailable Pods

컨테이너 수

- 상태별 컨테이너의 개수를 확인할 수 있습니다.
- 5초 주기의 데이터입니다. 실시간 데이터 조회 시 5초에 한 번씩 데이터를 호출합니다.

Kubernetes 오브젝트 목록

- 쿠버네티스 오브젝트([Deployment](#), [ReplicaSet](#), [Namespace](#), [StatefulSet](#)) 목록을 확인할 수 있습니다.
- 5분 주기의 데이터입니다. 실시간 데이터 조회 시 3분에 한 번씩 데이터를 호출합니다.

오브젝트 개수 추이

- 쿠버네티스 오브젝트([Deployment](#), [ReplicaSet](#), [Namespace](#), [StatefulSet](#))의 개수가 24시간 동안 어떻게 변했는지 확인할 수 있습니다.
- 5분 주기의 데이터입니다. 실시간 데이터 조회 시 3분에 한 번씩 데이터를 호출합니다.

클러스터 통합 정보

클러스터 프로젝트 정보와 네임스페이스 프로젝트로 분리된 Pod, 컨테이너, 로그, 이벤트 수, 이벤트 목록을 조회할 수 있습니다.

ⓘ 데이터를 조회하기 위해서는 해당 네임스페이스 프로젝트 **읽기** 권한이 필요합니다.

클러스터 오브젝트 [클러스터 통합 정보](#)

|| LIVE 12:29:25

Pod/컨테이너 수

프로젝트	네임스페이스	컨테이너 수		Pod 수	
		전체	State 별	전체	Phase별
K8s Namespace Project DEMO	virtual	20	Running : 20	20	Running : 20
Kubernetes Monitoring Demo(EKS) CP	amazon-guardduty	57	Running : 57	38	Running : 38
	default				
	dev				
	dev-prod				
	dev-test				
	kube-node-lease				
	kube-public				
	kube-system				
	multitx-sample				
	qa-ssy				
	qa-ssy2				
	qa-test				
	qa-test-prod				
	virtual-main				
	whatap-monitoring				

로그/이벤트 수

프로젝트	네임스페이스	로그 수		Kubernetes 이벤트 수		WhaTap 알람 수	
		최근 1시간	최근 24시간	최근 1시간	최근 24시간	최근 1시간	최근 24시간
K8s Namespace Project DEMO	virtual	14,817,409	186,103,131	0	0	0	0
Kubernetes Monitoring Demo(EKS) CP	amazon-guardduty	7,422,714	93,636,127	48	1,152	0	0
	default						
	dev						
	dev-prod						
	dev-test						
	kube-node-lease						
	kube-public						
	kube-system						
	multitx-sample						
	qa-ssy						
	qa-ssy2						
	qa-test						
	qa-test-prod						
	virtual-main						
	whatap-monitoring						

Kubernetes 이벤트 5

시간	프로젝트	Name	Type	Kind	Reason	Message
2023/11/22 12:21:49	Kubernetes Monitoring ...	whatap-virtual-deploy...	Warning	Pod	FailedScheduling	0/3 nodes are available: persistentvolumeclaim "premdata-pvc" not f
2023/11/22 12:25:49	Kubernetes Monitoring ...	virtual-pending-be-7c8d...	Warning	Pod	FailedScheduling	0/3 nodes are available: persistentvolumeclaim "premdata-pvc" not f
2023/11/22 12:26:19	Kubernetes Monitoring ...	whatap-virtual-deploy...	Warning	Pod	FailedScheduling	0/3 nodes are available: persistentvolumeclaim "premdata-pvc" not f
2023/11/22 12:26:19	Kubernetes Monitoring ...	virtual-pending-be-7c8d...	Warning	Pod	FailedScheduling	0/3 nodes are available: persistentvolumeclaim "premdata-pvc" not f
2023/11/22 12:26:49	Kubernetes Monitoring ...	whatap-virtual-deploy...	Warning	Pod	FailedScheduling	0/3 nodes are available: persistentvolumeclaim "premdata-pvc" not f

Pod/컨테이너 수

컨테이너 개수와 Pod 개수를 프로젝트 단위로 확인할 수 있습니다.

- **컨테이너 수**
 - **전체**: 클러스터에 속한 전체 컨테이너의 개수를 표시합니다.
 - **State 별**: 상태별로 진행 중인 컨테이너의 개수를 표시합니다.
- **Pod 수**
 - **전체**: 클러스터에 속한 전체 Pod의 개수를 표시합니다.
 - **Phase 별**: 상태별로 진행 중인 Pod의 개수를 표시합니다.

로그/이벤트 수

로그, Kubernetes 이벤트, WhaTap 알람의 개수를 프로젝트 단위로 확인할 수 있습니다.

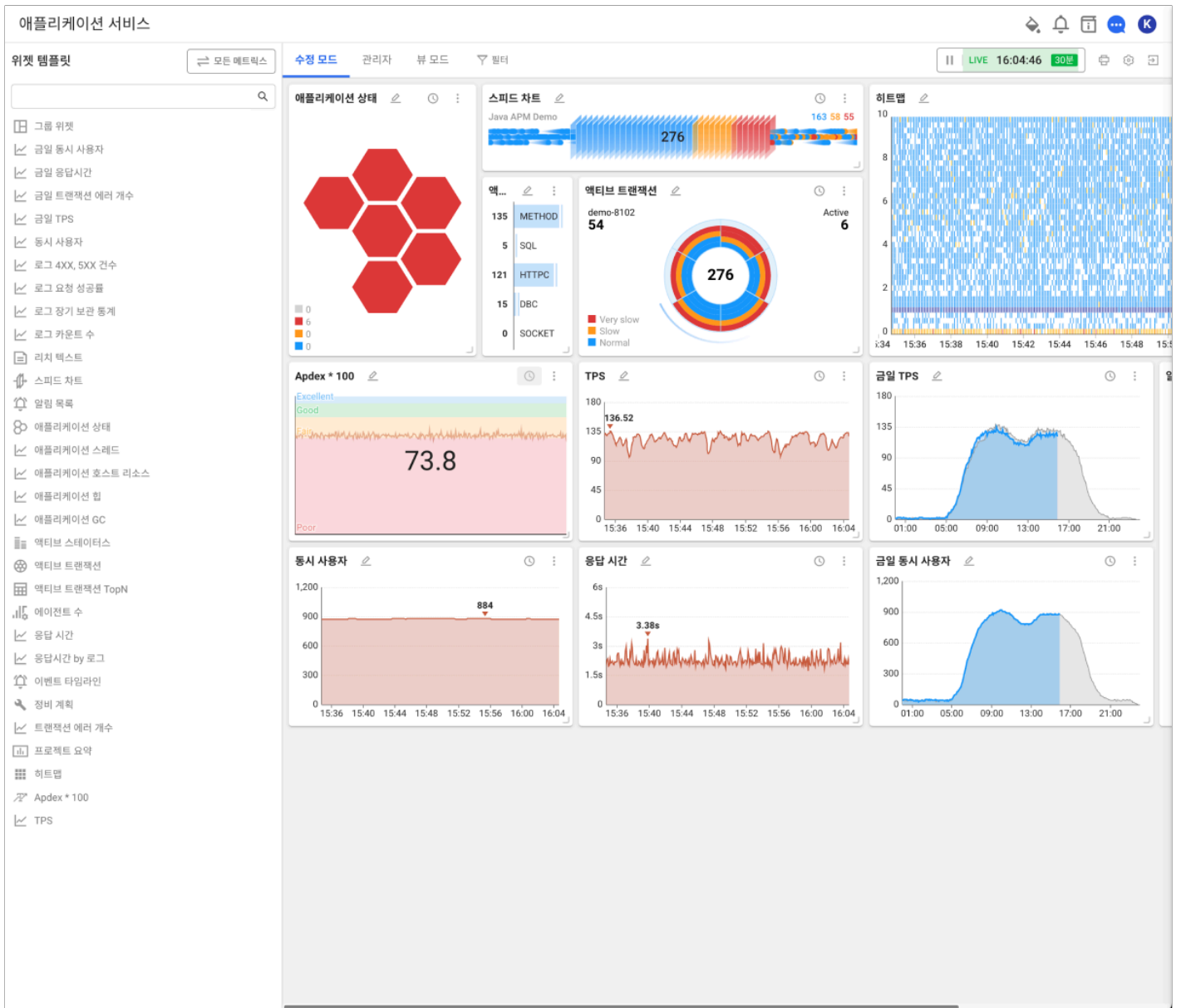
- **로그 수**
 - **최근 1시간**: 최근 1시간 동안 쌓인 로그 라인의 개수를 표시합니다.
 - **최근 24시간**: 하루 동안 쌓인 로그 라인의 개수를 표시합니다.
- **Kubernetes 이벤트 수**
 - **최근 1시간**: 최근 1시간 동안 쌓인 이벤트 라인의 개수를 표시합니다.
 - **최근 24시간**: 하루 동안 쌓인 이벤트 라인의 개수를 표시합니다.
- **WhaTap 알람 수**
 - **최근 1시간**: 최근 1시간 동안 발송된 알람 개수를 표시합니다.
 - **최근 24시간**: 하루 동안 발송된 알람 개수를 표시합니다.

Kubernetes 이벤트 5

클러스터 프로젝트와 네임스페이스 프로젝트에서 발생한 최근 5개 쿠버네티스 이벤트 목록을 확인할 수 있습니다.

Flex 보드

Flex 보드는 사용자 정의형 통합 대시보드입니다. 애플리케이션, 서버, 데이터베이스, 컨테이너 등 프로젝트의 데이터로 실시간 대시보드를 만들 수 있습니다.



- 사전 구성된 템플릿을 제공해 초기 설정을 손쉽게 진행할 수 있습니다. 이를 통해 사용자가 원하는 다양한 대시보드를 구성할 수

있습니다.

- 다양한 형태의 데이터 위젯을 추가할 수 있고, 개별 위젯의 속성을 수정해 사용자가 원하는 데이터 형식을 표시할 수 있습니다.
- 데이터 필터링을 통해 사용자가 원하는 모니터링 대상을 간추릴 수 있습니다.
- 시간 범위를 설정해 중요 시간의 데이터를 확인할 수 있습니다.
- 보조 차트를 활용해 다양한 방법으로 데이터를 확인할 수 있습니다.
- 대시보드를 즐겨 찾는 메뉴로 등록할 수 있습니다.
- 개인화된 대시보드를 타 계정에 복사해 활용할 수 있습니다.

Flex 보드는 다음의 진입 경로에 따라 활용 범위가 달라집니다.

- 홈 화면 > [Flex 보드](#)
 - 위젯 생성 시 조회 가능한 모든 프로젝트를 선택 옵션으로 제공합니다.
 - 사용자 계정에 대시보드가 저장되며 다른 사용자에게 복사하기 기능을 이용해 공유할 수 있습니다.
 - 개인 계정 대시보드로 권한에 따른 영향은 없으나 읽기 전용으로 공유된 대시보드의 경우 수정할 수 없습니다.
- 홈 화면 > 프로젝트 선택 > [대시보드](#) > [Flex 보드](#)
 - 위젯 생성 시 해당 프로젝트 정보를 자동 입력합니다.
 - 프로젝트 멤버들에게 생성한 Flex 보드가 자동 공유됩니다.
 - 프로젝트 수정 권한 또는 **프로젝트 플렉스보드 편집** 권한, **Site Admin** 권한을 가진 사용자가 수정할 수 있습니다.

✔ Flex 보드를 수정할 수 있는 사용자는 다음의 주요 기능을 이용할 수 있습니다.

- 대시보드를 json 파일 형식으로 내보내거나 가져올 수 있습니다.
- 대시보드 내의 데이터 요청 및 응답 내용을 확인할 수 있습니다.
- 위젯의 설정 옵션을 json 파일 형식으로 조회하고 수정할 수 있습니다.

ⓘ 프로젝트 내 [Flex 보드](#) 메뉴에서는 대시보드 수정 권한이 있는 사용자만이 [수정 모드](#) 및 [관리자 모드](#), [필터](#) 기능에 접근할 수 있습니다. 접근할 수 있는 멤버 권한은 다음과 같습니다.

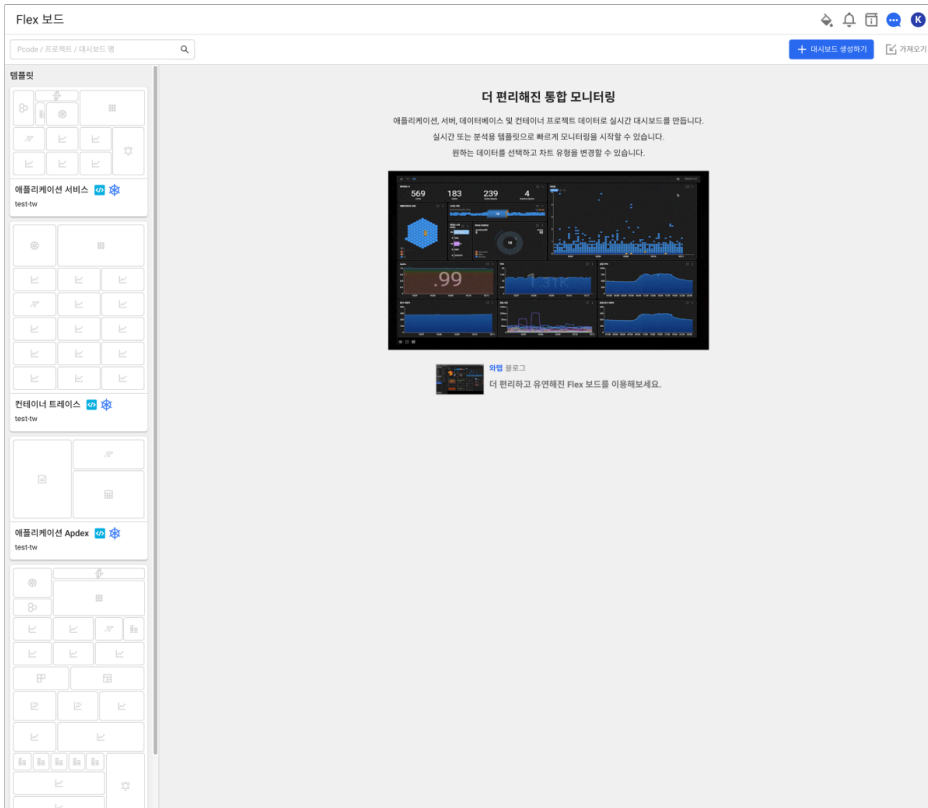
- 프로젝트 수정 권한
- **프로젝트 플렉스보드 편집** 권한
- **Site Admin** 권한
- **화면 모드** 및 **필터** 기능에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

- ⓘ • 멤버 권한에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

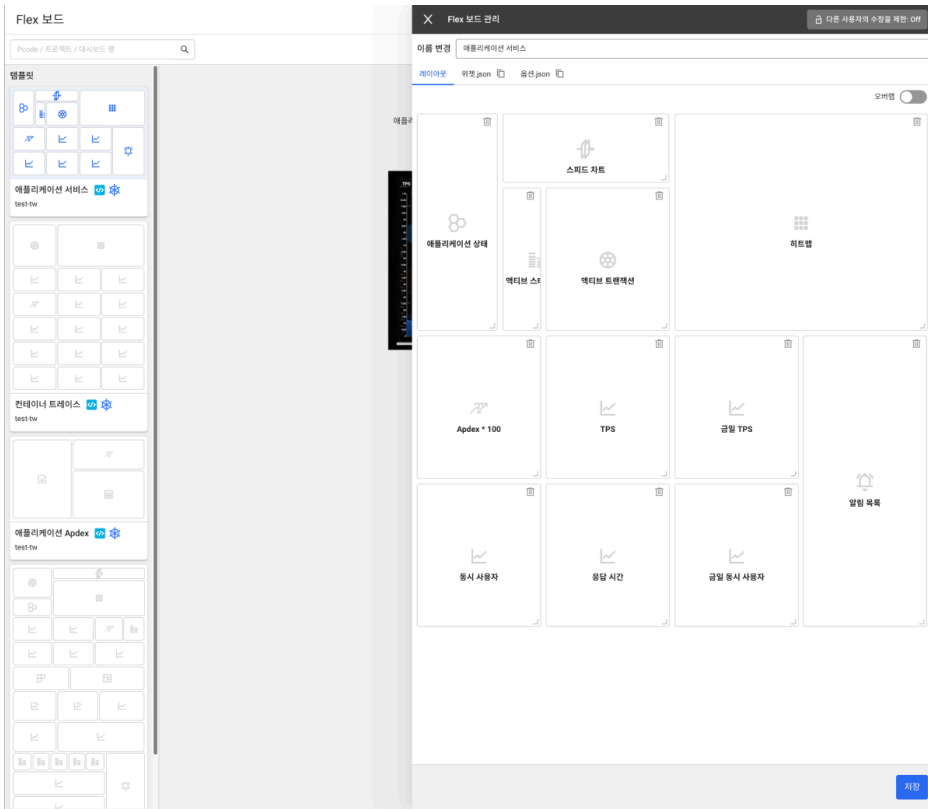
사전 템플릿 적용

제공된 사전 구성 템플릿을 이용하세요. 초기 설정을 손쉽게 진행하고 설정을 변경해 활용할 수 있습니다.

1. Flex 보드 메뉴를 선택하세요.
2. 왼쪽 템플릿 목록에서 원하는 템플릿을 선택하세요.



3. Flex 보드 관리 창이 나타나면 이름 변경에 원하는 이름을 입력하세요.



- 레이아웃 항목에 위젯을 마우스로 선택한 다음 원하는 위치로 드래그해 이동할 수 있습니다.
- 오버랩 토글 버튼을 선택해 옵션을 활성화하면 위젯을 서로 겹치게 배치할 수 있습니다.

4. 저장 버튼을 선택하세요.

선택한 템플릿 기준의 Flex 보드를 생성합니다. Flex 보드 메뉴에서 생성한 보드 목록을 확인할 수 있습니다.

Flex 보드 만들기


대시보드 생성

Flex 보드를 생성하고 위젯을 배치해 자신만의 대시보드를 만들 수 있습니다.

1. Flex 보드 메뉴에서 오른쪽 위에 **+ 대시보드 생성하기** 버튼을 선택하세요.
2. **대시보드 생성하기** 창이 나타나면 **이름** 입력 텍스트 상자에 대시 보드 이름을 입력하세요.
3. 위젯 배치 방법을 선택해 **대시보드 생성하기** 버튼을 클릭하세요.
 - **고정 레이아웃 플렉스 보드**: 픽셀 기반으로 자유롭게 위젯을 배치할 수 있습니다.
 - **반응형 플렉스 보드**: 브라우저 사이즈 기준으로 그리드 반응형 레이아웃을 제공합니다. **반응형 플렉스 보드**를 선택하면 그리드를 설정한 다음 **대시보드 생성하기** 버튼을 선택하세요.
4. **위젯 템플릿** 선택 화면에서 대시보드에 배치할 위젯을 선택하세요. 매트릭스 위젯을 추가하려면 **위젯 템플릿**의 오른쪽에 **모든 매트릭스** 버튼을 선택하세요.

- ❗ **위젯 템플릿**: 일반적인 모니터링 상황에서 중요하게 다뤄지는 지표를 간추려 사전 정의된 위젯 목록을 선택할 수 있습니다.
- **모든 매트릭스**: 사용자의 프로젝트에서 수집 중인 모든 매트릭스 데이터를 기준으로 위젯을 생성할 때 사용합니다. 매트릭스 위젯에 대한 자세한 설명은 [다음 문서](#)를 참조하세요.

5. 배치한 위젯의 위치를 이동하거나 크기를 조절하세요.
 - 배치한 위젯의 위쪽으로 마우스 커서를 이동하세요. 커서 모양이 십자 형태로 변경되면 위젯을 마우스로 클릭한 상태에서 드래그하세요. 위치를 이동할 수 있습니다.
 - 배치한 위젯의 오른쪽 아래로 마우스 커서를 이동하세요. 커서 모양이 화살표 형태로 변경되면 위젯을 마우스로 클릭한 상태로 드래그하세요. 원하는 크기로 조절할 수 있습니다.
6. 추가할 위젯을 모두 배치를 완료한 다음 **뷰 모드**를 선택해 레이아웃을 확인하세요. 레이아웃을 다시 수정하려면 **수정 모드**를 선택해 위젯의 배치를 변경하세요.


모든 과정을 완료했다면 오른쪽 위에  (**목록으로 가기**) 버튼을 선택하세요. Flex 보드 메뉴의 **보드** 목록에서 생성한 보드를 확인할 수 있습니다. 생성한 보드를 선택해 새로 생성한 Flex 보드를 확인할 수 있습니다.

- ❗ 고객의 의견을 반영해 대시보드 템플릿 또는 위젯을 추가하려고 합니다. 필요한 대시보드 템플릿 또는 위젯이 있다면

ⓘ support@whatap.io로 문의해 주세요.

Flex 보드 관리하기

Flex 보드 편집하기

Flex 보드 메뉴의 [대시보드 목록](#)에서  버튼을 선택하세요. 화면 오른쪽에 Flex 보드 관리 창이 나타납니다.

ⓘ Flex 보드 편집 기능 권한 안내

- 홈 화면 > [Flex 보드](#)
개인 계정 대시보드로 권한에 따른 영향은 없으나 읽기 전용으로 공유된 대시보드의 경우 수정할 수 없습니다.
- 홈 화면 > 프로젝트 선택 > [대시보드](#) > [Flex 보드](#)
프로젝트 수정 권한 또는 [프로젝트 플렉스보드 편집](#) 권한, **Site Admin** 권한을 가진 사용자가 수정할 수 있습니다.
- 멤버 권한에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.


The image shows the Flex Board management interface. On the left, the 'Flex 보드' (Flex Board) configuration screen is visible, featuring a search bar for 'Pcode / 프로젝트 / 대시보드 명' and a list of dashboard items. On the right, the 'Flex 보드 관리' (Flex Board Management) screen is shown, displaying a grid of dashboard widgets. The widgets include '스피드 차트' (Speed Chart), '애플리케이션 상태' (Application Status), '액티브 트랜잭션' (Active Transactions), 'Apdex * 100', 'TPS', '금일 TPS', '동시 사용자' (Concurrent Users), '응답 시간' (Response Time), '금일 동시 사용자' (Concurrent Users Today), and '알림 목록' (Alert List). A '저장' (Save) button is located at the bottom right of the management screen.

설정할 수 있는 속성은 다음과 같습니다.



- **이름 변경:** 대시보드의 이름을 수정할 수 있습니다.
- **프로젝트:** 대시보드 위젯에 데이터를 가져올 프로젝트를 선택할 수 있습니다.

❗ 이 옵션은 홈 화면 > Flex 보드 메뉴에서만 설정할 수 있습니다.

• 레이아웃

- 위젯을 선택하고 드래그하여 위치를 이동할 수 있습니다.
- 위젯 오른쪽 아래를 마우스로 선택한 다음 드래그해 위젯의 크기를 조절할 수 있습니다.
- 위젯을 삭제하려면 위젯 오른쪽 위에  버튼을 선택하세요.
- **오버랩** 토글 버튼을 활성화하면 위젯의 배치를 겹쳐진 상태로 배치할 수 있습니다.

❗ 홈 화면 > Flex 보드 메뉴로 진입한 경우, 각 위젯마다 데이터를 가져올 프로젝트를 선택할 수 있습니다. 위젯 오른쪽 위에 + 버튼을 클릭한 다음 원하는 프로젝트를 선택하세요. 프로젝트를 검색해 선택할 수 있고, 다중 선택할 수 있습니다. 선택을 완료한 다음 **적용** 버튼을 선택하세요.

- **위젯.json**: 대시보드의 위젯 설정을 json 형식으로 불러올 수 있습니다.  버튼을 선택하면 json 내용을 복사합니다.
- **옵션.json**: 대시보드에 설정된 옵션을 json 형식으로 불러올 수 있습니다.  버튼을 선택하면 json 내용을 복사합니다.

Flex 보드 삭제하기

Flex 보드 > 대시보드 목록에서 삭제하려는 항목의 오른쪽에  버튼을 선택하세요. 확인 메시지가 나타나면 **삭제** 버튼을 선택하세요.

- ❗ • 대시보드 소유자가 대시보드를 삭제하는 경우 대시보드를 공유한 모든 사용자 계정에서도 해당 대시보드를 삭제합니다.
- 대시보드를 공유 받은 사용자가 대시보드를 삭제하면, 해당 사용자의 **대시보드 목록**에서만 삭제합니다. 원본 대시보드는 그대로 유지합니다.

메트릭스 위젯

메트릭스란?

프로젝트에서 수집 중인 숫자로된 성능 지표를 **메트릭스**라고 부릅니다. 메트릭스는 '카테고리'로 분류되며, 카테고리별 태그를 이용해 데이터를 탐색할 수 있습니다.

- **카테고리(Category)**: 관련한 지표들을 묶는 단위
- **태그(Tags)**: 수집 대상을 구분할 수 있는 고유 정보가 포함된 데이터
- **필드(Fields)**: 에이전트로부터 수집한 지표

메트릭스에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

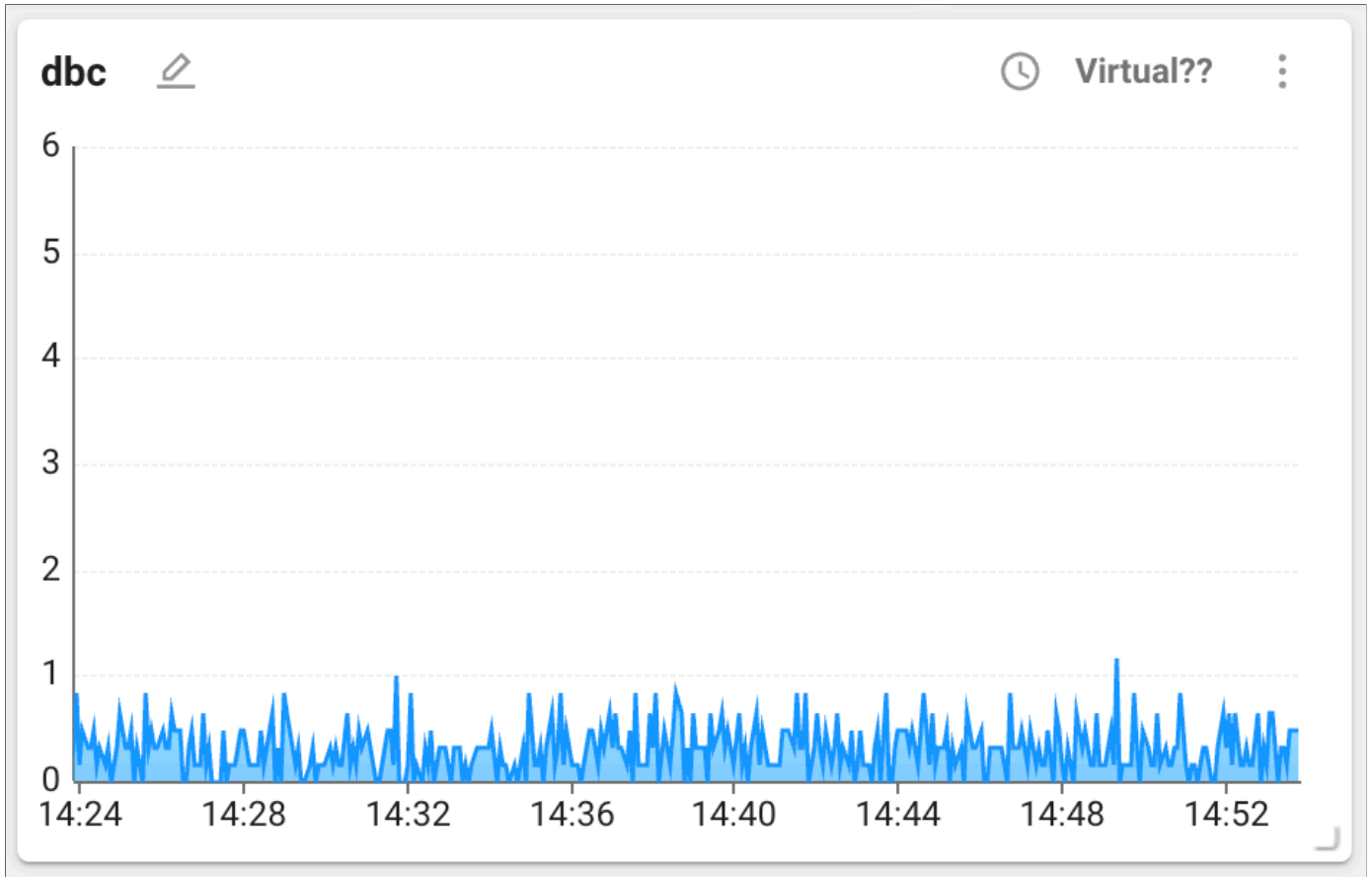
메트릭스 위젯 종류

모든 **메트릭스**에서 추가할 수 있는 위젯 유형은 카테고리에 대한 태그와 필드 정보를 나열한 테이블 형식의 위젯, 단일 필드에 대한 추이 그래프로 나뉩니다.

- **테이블 위젯**: 선택한 메트릭스 카테고리에 대한 데이터를 표 형식으로 제공합니다.

container	containerKey	host_ip	okindName	oname	onodeName	pid	Project	type	dbc	httpc	method	socket	sql	oid
prod-agent.service	1204611323	10.21.2.34	demo-okind-0	demo-8102	node-0	3788129	W_JAVA_DEMO	java	0.36	9.6	11	-	3.05	1387800924
prod-agent.service	1204611323	10.21.2.34	demo-okind-1	demo-8103	node-1	3788128	W_JAVA_DEMO	java	0.33	9.66	10.55	-	3.14	633280970
prod-agent.service	1204611323	10.21.2.34	demo-okind-1	demo-8105	node-1	3788134	W_JAVA_DEMO	java	0.3	9.19	11.37	-	2.91	-857948929
prod-agent.service	1204611323	10.21.2.34	demo-okind-1	demo-8101	node-1	3788130	W_JAVA_DEMO	java	0.28	9.93	10.82	-	3.07	-877561626
prod-agent.service	1204611323	10.21.2.34	demo-okind-0	demo-8100	node-0	3788127	W_JAVA_DEMO	java	0.34	9.71	11.38	-	2.88	-1128904592
prod-agent.service	1204611323	10.21.2.34	demo-okind-0	demo-8104	node-0	3788126	W_JAVA_DEMO	java	0.29	9.69	10.96	-	3.11	-1143239575

- **시리즈 위젯**: 선택한 메트릭스 카테고리의 선택 필드에 대한 데이터를 시리즈 차트 형식으로 제공합니다.



① 메트릭스 위젯 설정에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.


위젯 관리하기

Flex 보드에 배치한 위젯 관리를 통해 대시보드를 사용자가 원하는 대로 꾸밀 수 있습니다.

위젯 추가

이미 만들어진 Flex 보드에 위젯을 추가할 수 있습니다.

1. Flex 보드 메뉴의 **보드** 목록에서 위젯을 추가할 대시보드를 선택하세요.
2. 화면 왼쪽 위에 **수정 모드**를 선택하세요.
3. 왼쪽에 **위젯 템플릿** 목록에서 원하는 위젯을 선택하세요. 매트릭스 위젯을 추가하려면 **모든 매트릭스**를 선택한 다음 원하는 위젯을 클릭하세요.
 - 홈 화면의 Flex 보드 메뉴로 진입한 경우 위젯을 추가했다면 **+ 프로젝트 선택** 버튼을 클릭해 수집 대상 프로젝트를 선택하세요.
 - 추가할 위젯이 있다면 같은 과정을 반복하세요.


모든 과정을 완료했다면 화면 위에 **뷰 모드**를 선택해 레이아웃을 확인한 다음  (**목록으로 가기**) 버튼을 선택하세요.

위젯 속성 변경


Flex 보드 메뉴의 **보드** 목록에서 위젯 속성을 변경할 보드를 선택하세요. **수정 모드**로 진입합니다.

위젯 이름 변경

이름을 변경할 위젯의 이름 오른쪽에  버튼을 선택하세요. 위젯의 **제목**, **폰트 크기**, **색상**을 설정한 다음 **저장** 버튼을 선택하세요.

- ✔ • 저장 시 **전체 위젯에 적용** 토글 버튼의 기능을 활성화하면 폰트 크기와 색상을 다른 위젯에도 적용합니다.
 - 위젯의 제목을 잠금 설정하려면  버튼을 선택한 다음 **저장** 버튼을 클릭하세요.

시간 설정

시간을 설정하려는 위젯의 오른쪽 위에  버튼을 선택하세요. **시간 선택** 창이 나타납니다.

- 대시보드 화면에 설정한 시간을 적용하려면 **대시보드 설정을 따름**을 체크하세요. 대시보드의 시간 설정은 오른쪽 상단에 위치합니다.
- 사용자가 원하는 시간을 설정을 적용하려면 **대시보드 설정을 따름**을 체크 해제하세요. < 또는 > 버튼을 이용해 원하는 시간을 설정하세요.

시간 설정을 완료했다면 ✕ 버튼을 선택하세요.

ⓘ 위젯의 데이터 유형에 따라서 시간 설정 기능을 지원하지 않을 수 있습니다.

출력 데이터 옵션 설정

차트형 위젯의 출력 데이터 옵션을 설정해 사용자가 원하는 데이터를 선택할 수 있습니다. 위젯의 오른쪽 위에 ⋮ 버튼을 선택하세요.

- 프로젝트, 에이전트, 애플리케이션 등의 종류를 선택해 데이터를 필터링할 수 있습니다.
- 개별, 병합 데이터를 선택해 차트에 표시되는 그래프의 유형을 변경할 수 있습니다.
- ⋮ > ⚙️을 선택하세요. **차트 설정** 창을 통해 세부 옵션을 변경할 수 있습니다. 설정할 수 있는 옵션은 다음과 같습니다.
 - **차트 유형**: 집계하는 현재 수치를 차트에 표시할 수 있는 옵션입니다.
 - **y축 차트 최대값**: y축에 차트 최대값을 **자동** 또는 **고정**으로 선택할 수 있습니다.
 - **수평선 설정**: **+ 추가** 버튼을 선택해 임계치 또는 범위를 수평선으로 표시할 수 있습니다. 여러 개를 추가해 적용할 수 있습니다. 수평선을 추가한 다음에는 **적용** 버튼을 선택해야 차트에 반영됩니다.
 - **차트 최대값 표시**: 차트에 표시되는 그래프의 최대값을 표시합니다.
 - **보조 차트**: 위젯의 주요 차트에 보조 차트를 추가 설정할 수 있습니다. 보조 차트의 종류를 선택한 다음 레이아웃, 크기, 데이터 병합 여부를 선택하세요. 다른 위젯에도 공통 적용하려면 **전체 위젯에 적용** 버튼을 선택하세요. 보조 차트는 데이터를 다양한 방법으로 보고 싶을 때 유용합니다.

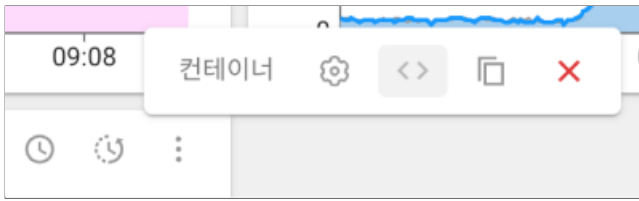
옵션 설정을 모두 완료했다면 **차트 설정** 창의 왼쪽 위에 ✕ 버튼을 클릭하거나 **차트 설정** 창 영역 밖을 클릭하세요.

ⓘ 위젯의 데이터 유형에 따라서 선택할 수 있는 옵션은 다를 수 있습니다.

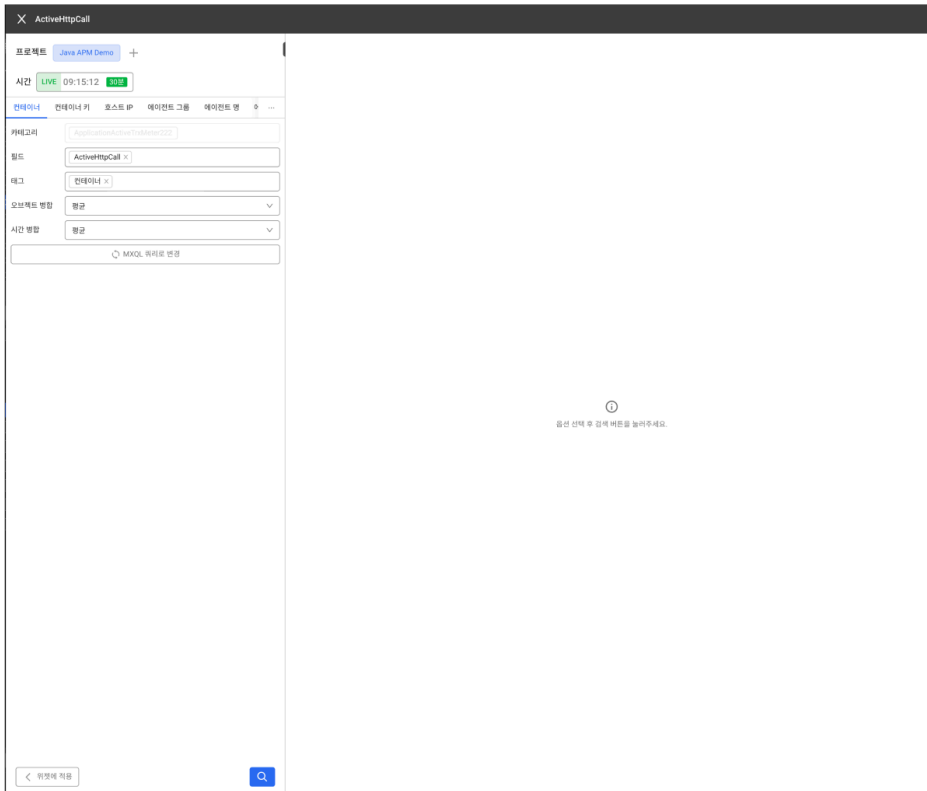
메트릭스 위젯 데이터 변경

메트릭스 위젯의 데이터 조회 조건을 변경할 수 있습니다.

1. Flex 보드 > 수정 모드에서 변경할 위젯의 오른쪽 위에 ⋮ 버튼을 선택하세요.




2. <> 버튼을 선택하세요. 데이터 설정 편집 창이 나타납니다.



3. 다음을 참조해 설정을 변경한 다음 < 위젯에 적용 > 버튼을 선택하세요.
 - 위젯 생성 시 기본 생성되는 메트릭스 옵션이 탭에 표시됩니다.
 - 메트릭스 옵션에는 **카테고리**와 **필드**, **태그**가 있습니다.
 - **카테고리** 항목은 현재 변경 기능을 제한합니다.
 - **필드**, **태그** 항목은 복수 선택할 수 있습니다. 선택한 태그 옵션으로 차트 데이터를 그룹화합니다.
 - 🔍 버튼을 선택해 원본데이터를 조회할 수 있습니다.
 - **오브젝트 병합** 및 **시간 병합** 항목에 대한 자세한 내용은 [데이터 병합 옵션](#)을 참조하세요.

데이터 병합 옵션

Flex 보드의 [모든 메트릭스](#) 목록에서 배치한 메트릭스 위젯은 데이터 병합 옵션을 제공합니다.

1. 수정 모드에서 [모든 메트릭스](#) 버튼을 선택해 [위젯 템플릿](#) 목록을 [모든 메트릭스](#) 목록으로 변경하세요.
2. 메트릭스 위젯을 레이아웃에 배치하세요.
3. 메트릭스 위젯 오른쪽 위에  버튼을 선택하세요.
4. 태그(예: 에이전트 명) 옵션을 선택하면 데이터 병합 옵션이 나타납니다. 위젯의 크기에 따라 메트릭스 옵션의 위치가 다를 수 있습니다.



데이터 병합은 [오브젝트 병합](#)과 [시간 병합](#) 방법을 제공합니다.

- **오브젝트 병합**

좌측에서 선택한 태그(예: 에이전트 명)를 기준으로 차트 데이터를 그룹화할 때 사용합니다. 이때 서로 다른 필드(예: CPU)값을 가진 데이터들 중에서 태그가 일치하는 경우 해당 데이터를 병합하는 방법입니다.

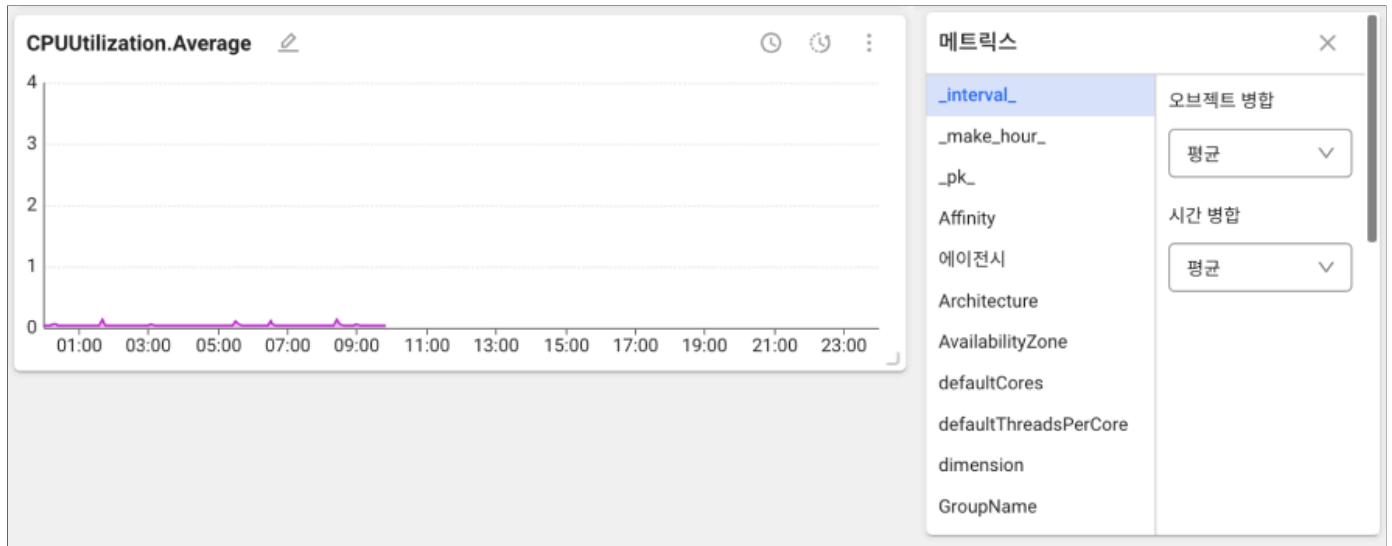
- **시간 병합**

조회 시간이 긴 경우 주로 사용합니다. 원본 데이터에서 필드 값이 같은 데이터끼리 시간 범위에 따라 일정한 시간 간격(5분 또는 1시간)으로 데이터를 병합합니다. 이 병합 방법을 시간 병합이라고 합니다.

- **AWS CloudWatch 메트릭스**

AWS 모니터링 지표의 경우 통계 방식이 필드명 뒤에 suffix(접미사) 형식으로 추가됩니다. 데이터 병합 기본값은 이 suffix를

참조하고 있습니다.



위젯 복사

1. 위젯의 오른쪽 위에 버튼을 선택하세요.
2. 버튼을 선택하세요.
3. 복사 창이 나타나면 + 위젯 추가를 선택하세요.
4. 목록에 위젯의 이름을 변경하세요.
5. 적용 버튼을 선택하세요.

수정 모드 화면에서 복사한 위젯을 확인할 수 있습니다.

위젯 삭제

1. 위젯의 오른쪽 위에 버튼을 선택하세요.
2. 버튼을 선택하세요.
3. 확인 메시지 창이 나타나면 삭제 버튼을 선택하세요. 삭제를 취소하려면 취소 버튼을 선택하세요.


Flex 보드 화면 모드

Flex 보드의 화면 모드에 따라 제공하는 기능을 설명합니다.

- ❗ • 프로젝트 내 **Flex 보드** 메뉴에서는 대시보드 수정 권한이 있는 사용자만이 **수정 모드** 및 **관리자** 모드, **필터** 기능에 접근할 수 있습니다. 접근할 수 있는 멤버 권한은 다음과 같습니다.
 - 프로젝트 수정 권한
 - 프로젝트 플렉스보드 편집 권한
 - **Site Admin** 권한
- 멤버 권한에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

화면 모드

Flex 보드는 다음과 같은 화면 모드를 제공합니다. 화면 모드에 따라 설정할 수 있는 기능이 다릅니다.

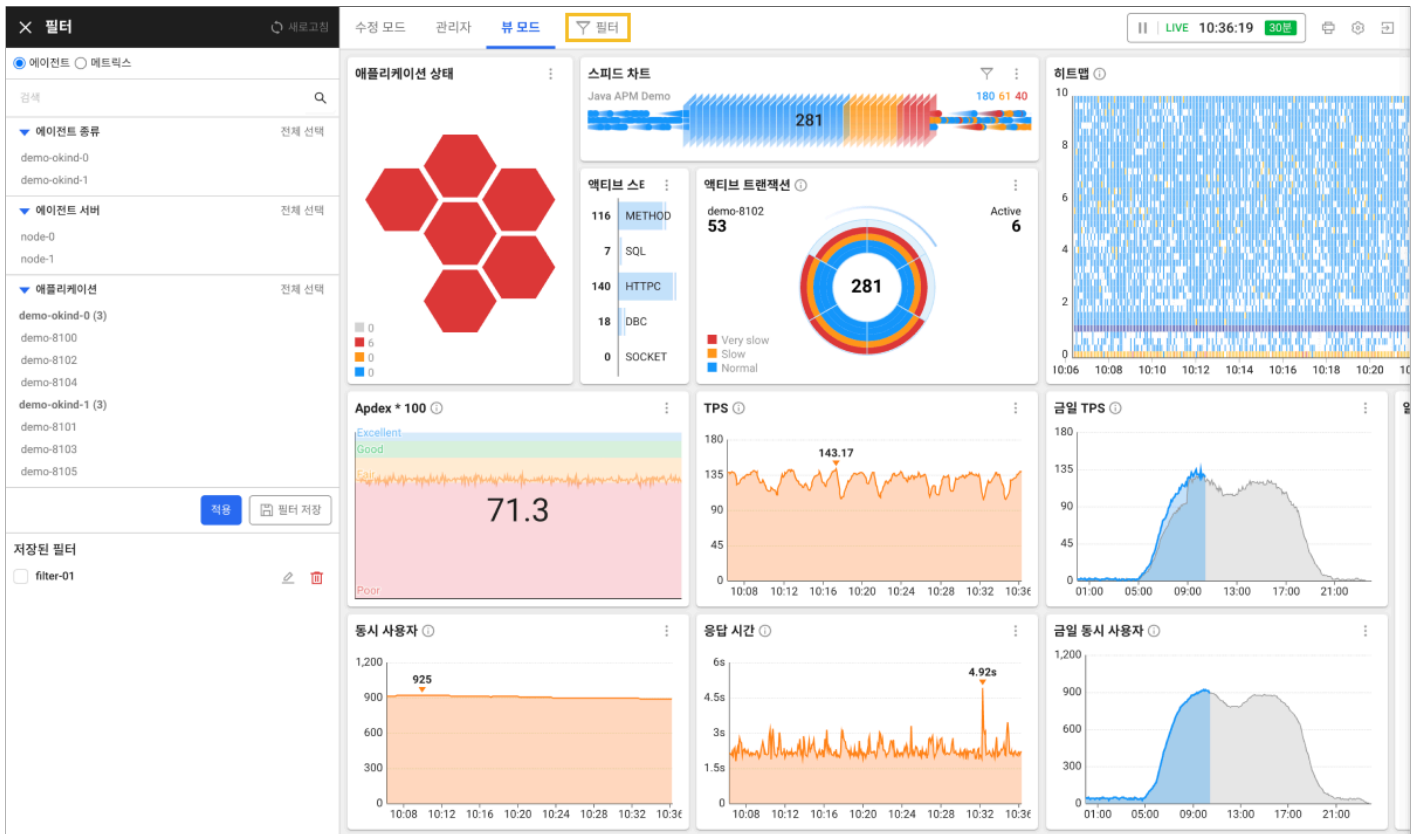
- **수정 모드**: 위젯 템플릿 목록에서 Flex 보드의 레이아웃에 위젯을 배치하거나 위젯을 수정, 삭제, 위치 조정할 수 있습니다.
- **관리자**: 위젯의 세부 속성을 json 형태로 관리할 수 있습니다. 레이아웃에 배치한 위젯의 오른쪽 위에 위치한  버튼을 선택하세요.
- **뷰 모드**: 위젯 배치, 크기 조절, 세부 옵션 등의 설정을 완료한 다음 대시보드와 같이 각 위젯의 데이터를 실시간으로 모니터링할 수 있습니다.

Flex 보드에서는 다음의 공통 기능을 사용할 수 있습니다.

- 데이터 필터링
- 실시간 데이터 조회 범위 설정
- 인쇄 모드
- Flex 보드 관리
- 데이터 병합 옵션

데이터 필터링

Flex 보드는 수백 가지의 종류의 데이터를 다룹니다. 데이터 필터링 기능을 이용해 모니터링하려는 대상을 간추려 Flex 보드를 구성하세요.



1. 화면 위에 ① 필터를 선택하세요.

2. 화면 왼쪽으로 ② 필터 창이 나타납니다.

- 에이전트 옵션을 선택했다면 에이전트 종류 또는 애플리케이션 목록 기준으로 필터링 대상을 선택할 수 있습니다. 필터링 대상을 모두 선택하려면 전체 선택을 선택하세요. 필요한 경우 검색 기능을 이용해 선택할 수 있습니다.
- 매트릭스 옵션을 선택했다면 카테고리 옵션에서 프로젝트, 에이전트 유형에 따라 필터링 대상을 선택할 수 있습니다. 카테고리 옵션은 + And를 선택해 추가할 수 있습니다. 포함, 제외 옵션을 선택하면 문자열로 필터링 대상을 자동 선택합니다.

3. 필터링 대상 선택을 완료했다면 적용 버튼을 선택하세요.

사용자가 설정한 필터링 설정을 저장해 이용하려면 필터 저장 버튼을 선택하세요. 필터 이름을 입력하고 저장을 선택하세요. ③ 저장된 필터는 필터 창 하단에 나타납니다.

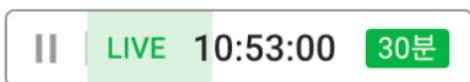
필터링 조건은 에이전트 타입과 매트릭스 타입으로 구분합니다.

타입	에이전트	메트릭스
선택 옵션	와탭 프로젝트 모니터링 대상 (애플리케이션, 서버, 데이터베이스, 컨테이너 및 쿠버네티스 POD와 NODE)	에이전트를 포함한 모든 메트릭스 태그
적용 범위	모든 위젯	메트릭스 데이터를 사용하는 모든 위젯
필터링 조건	체크박스	일치 , 포함 , 제외 방식

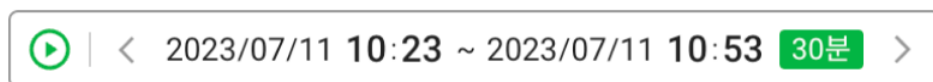
- ① • 에이전트 타입은 기본적인 모니터링 대상만을 선택 옵션으로 제공합니다.
 - 메트릭스 타입은 프로젝트에서 수집 중인 모든 메트릭스 데이터의 태그를 선택 옵션으로 제공합니다. 메트릭스 타입은 클라우드 모니터링과 같이 에이전트가 아닌 형태로 데이터를 수집한 후 필터링을 적용할 때 유용합니다.
 - 메트릭스에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

실시간 데이터 조회 범위 설정

Flex 보드 메뉴의 화면 오른쪽 위에 UI 요소를 이용해 위젯에 데이터를 출력하는 시간을 설정할 수 있습니다.




II 버튼을 선택하면 실시간 데이터 조회를 일시 정지하고 다음과 같이 조회 시간을 설정할 수 있습니다.



- 가장 오른쪽의 녹색 배경의 시간 버튼을 선택하세요. 다양한 시간 설정을 선택할 수 있는 옵션이 나타납니다. 원하는 시간 설정을 선택하세요.
- < 또는 > 버튼을 클릭하면 설정된 시간 범위 만큼 조절할 수 있습니다.
- 날짜와 시간 텍스트 영역을 클릭하면 날짜와 시간을 선택할 수 있는 옵션이 나타납니다.
- 조회 시간 설정을 모두 완료했다면 ▶ 버튼을 선택하세요. 실시간 데이터 조회를 시작합니다.


인쇄하기


Flex 보드 화면을 PDF로 저장해 인쇄할 수 있습니다. Flex 보드 화면의 오른쪽 위에  버튼을 선택하세요. 화면을 인쇄 모드로 전환합니다. 다음의 옵션을 설정한 다음 [PDF 다운로드](#)를 선택하세요.

- **필터 옵션 보기**: 화면에 필터 옵션을 켜거나 끌 수 있습니다.
- **가로/세로 방향으로 변환**: 화면을 가로 또는 세로로 변환할 수 있습니다.


인쇄 모드를 종료하려면 오른쪽 위에  **인쇄모드 종료**를 선택하세요.

Flex 보드 관리

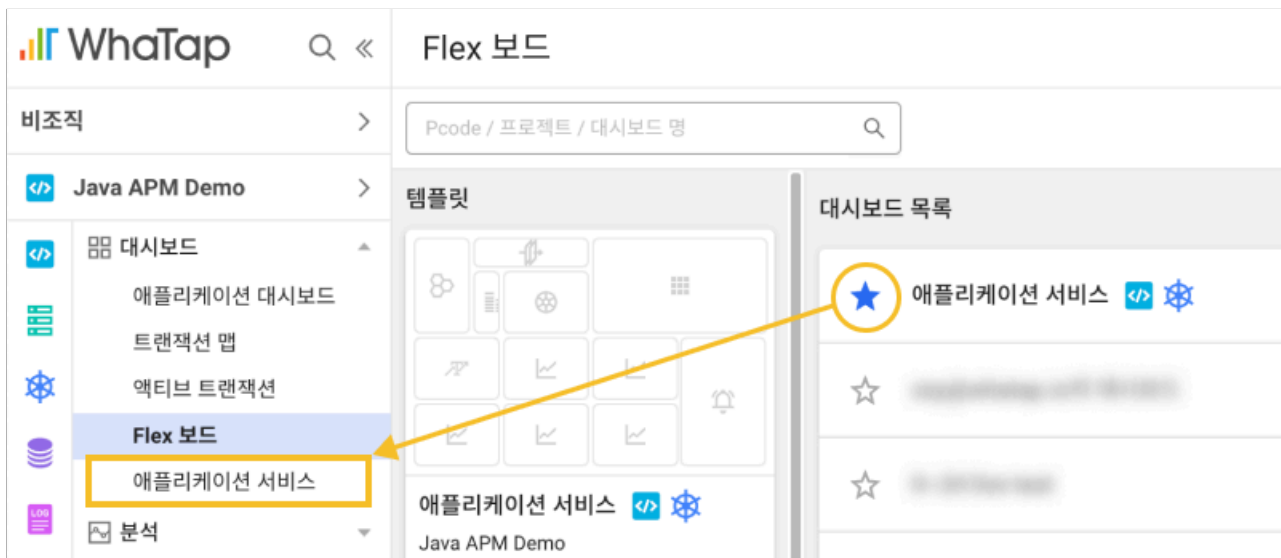
Flex 보드의 레이아웃을 변경하거나 위젯의 위치 변경, 크기 조절, 이름 변경 등의 관리 기능을 제공합니다. Flex 보드 화면의 오른쪽 위에  버튼을 선택하세요. [Flex 보드 관리](#) 창이 나타납니다.

- **이름 변경**: Flex 보드의 이름을 변경할 수 있습니다. 변경하려는 이름을 입력하세요.
- **레이아웃**: 위젯의 배치를 변경하거나 위치 및 크기를 조절할 수 있습니다. 배치한 위젯을 삭제하려면  버튼을 클릭하세요. **오버랩** 토글 버튼을 선택해 옵션을 활성화하면 위젯을 서로 겹치게 배치할 수 있습니다.
- **다른 사용자의 수정을 제한**: 다른 사용자가 해당 Flex 보드의 수정을 할 수 없도록 설정합니다. 수정 권한이 있는 멤버만 수정 또는 삭제, 위젯을 편집할 수 있습니다. 버튼을 클릭하면 **다른 사용자의 수정을 제한: on**으로 변경됩니다.
- **위젯 json**: Flex 보드에 포함된 위젯을 json 형태로 편집해 관리할 수 있습니다.
- **옵션 json**: Flex 보드의 설정을 json 형태로 편집해 관리할 수 있습니다. 프로젝트 정보, Flex 보드 필터, 시간 선택 등의 설정값을 편집할 수 있습니다.

모든 설정을 완료한 다음에는 화면 오른쪽 아래에 [저장](#) 버튼을 클릭하세요.

- ① • **위젯 json, 옵션 json** 기능은 대시보드 관리자 권한의 사용자에게만 제공합니다.
 - [Flex 보드 관리](#) 기능은 [대시보드](#) > [Flex 보드](#) 메뉴에서 [보드 목록](#)의  버튼을 선택해 진입할 수도 있습니다.

대시보드 즐겨찾기로 등록

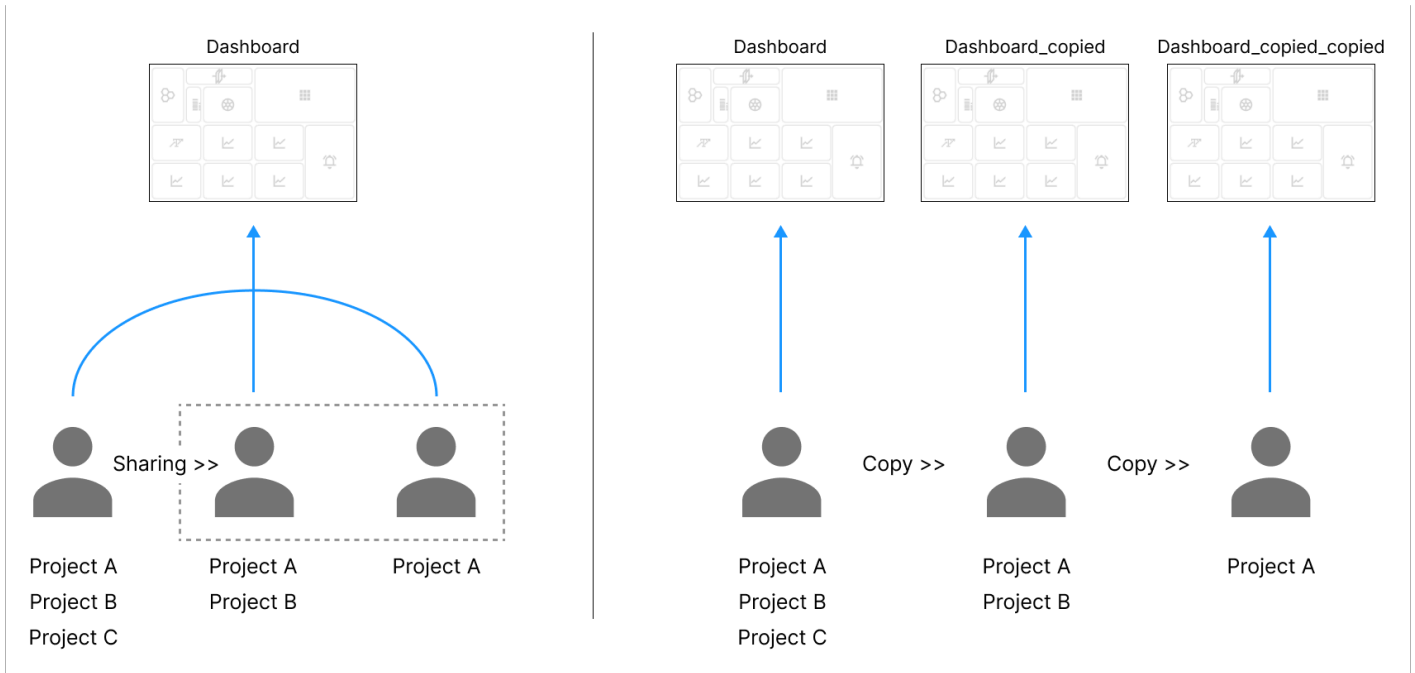


사용자가 생성한 Flex 보드는 [대시보드](#) 메뉴 하위에 즐겨찾는 메뉴로 등록할 수 있습니다. [대시보드](#) > [Flex 보드](#) 메뉴에서 [대시보드 목록](#)의 ☆ 버튼을 선택하세요. 즐겨찾기로 설정된 항목은 ★으로 변경됩니다.

Flex 보드 공유하기

통합 Flex 보드 메뉴에서 생성한 대시보드를 다른 멤버와 공유하거나 복사해 재사용할 수 있습니다.

공유와 복사의 차이



- 대시보드를 공유한 경우 공유한 멤버는 모두 같은 대시보드를 볼 수 있습니다.
- 대시보드를 다른 멤버 또는 나에게 복사한 경우 복사된 별도의 대시보드를 볼 수 있습니다.
- **읽기 전용**으로 대시보드를 공유 또는 복사 받은 멤버는 대시보드를 수정할 수 없지만 **수정 모드**로 공유 또는 복사된 대시보드는 수정할 수 있습니다.
- **읽기 전용**으로 대시보드를 공유받은 멤버는 읽기 전용으로 대시보드를 공유할 수 있으며 복사할 수 있습니다.
- 대시보드에 포함된 프로젝트 중 최소 1개 프로젝트에 초대된 멤버에게 대시보드를 공유할 수 있습니다. 공유 또는 복사 받을 멤버가 프로젝트에 소속되어 있는지 확인하세요.




❗ 대시보드 공유 및 복사 기능이 업데이트됐습니다

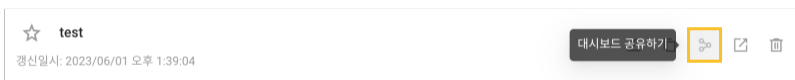
이전에는 공유 또는 복사 받을 멤버에게 수정 권한이 있어야 했지만, [Service 2.3.0 릴리스](#) 이후 권한에 상관없이 프로젝트에 소속되어 있기만 한다면 복사 또는 공유할 수 있습니다. 단, 공유 받은 대시보드의 프로젝트 중 **조회 분석** 권한이 없는 프로젝트는 데이터를 조회할 수 없습니다. 멤버 권한에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

Flex 보드 공유하기

사용자가 생성한 Flex 보드를 다른 멤버와 공유해 동일한 대시보드를 볼 수 있습니다. 이를 통해 업무를 공유할 수 있습니다. 공유한 대시보드를 수정한 경우 공유받은 모든 멤버는 동일한 대시보드를 확인할 수 있습니다.

❗ 이 기능은 **통합 Flex 보드** 메뉴에서만 이용할 수 있습니다. 프로젝트의 **Flex 보드** 메뉴에서는 이용할 수 없습니다.

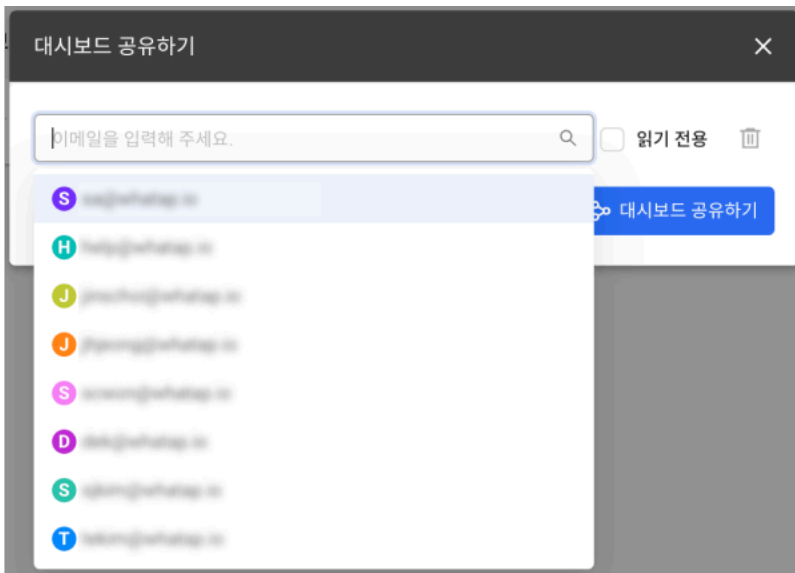
1. **통합 Flex 보드** 메뉴의 **대시보드 목록**에서 공유하려는 대시보드의  버튼을 선택하세요.



2. **대시보드 공유하기** 창이 나타나면 **+ 계정** 버튼을 선택하세요.



3. 대시보드를 공유할 멤버를 선택하세요.



- 공유 대상을 추가하려면 + **계정** 버튼을 선택하세요. 멤버를 추가로 선택할 수 있습니다.
- 대시보드에 포함된 프로젝트 중 최소 1개 프로젝트에 초대된 멤버에게 대시보드를 공유할 수 있습니다. 공유 또는 복사 받을 멤버가 프로젝트에 소속되어 있는지 확인하세요.

4. **대시보드 공유하기** 버튼을 선택하세요.



- **읽기 전용**을 선택하면 공유받은 멤버는 대시보드를 수정할 수 없습니다.
- 공유받을 멤버 모두에게 읽기 전용으로 공유하려면 **읽기 전용(전체)**를 선택하세요.

대시보드를 공유한 항목은 **대시보드 목록**에서 **공유** 태그가 표시된 것을 확인할 수 있습니다. **공유** 태그에 마우스를 오버하면 대시보드를 공유한 멤버에 대한 정보를 확인할 수 있습니다. 읽기 전용으로 공유한 멤버에게는 **읽기 전용** 태그가 표시됩니다.

대시보드를 공유받은 경우	대시보드를 공유한 경우(대시보드 소유자)
	

✔ 대시보드 공유 조건

- 대시보드 공유 기능은 홈 화면 > [통합 Flex 보드](#) 메뉴에서만 이용할 수 있습니다.
- 누구나 자신의 대시보드를 공유할 수 있으며, 공유받은 대시보드를 다시 공유하거나 복사할 수 있습니다.
- [읽기 전용](#)으로 대시보드를 공유받은 멤버는 대시보드를 수정할 수 없지만 [수정 모드](#)로 공유받은 대시보드는 수정할 수 있습니다.
- 대시보드 소유자가 대시보드를 삭제하면 대시보드를 공유받은 모든 멤버 계정에서도 해당 대시보드가 삭제됩니다.
- 대시보드를 공유받은 멤버가 대시보드를 삭제하면, 해당 멤버의 [대시보드 목록](#)에서만 삭제됩니다. 원본 대시보드는 그대로 유지합니다.
- 공유 받은 대시보드의 프로젝트 중 [조회 분석](#) 권한이 없는 프로젝트는 데이터를 조회할 수 없습니다.

Flex 보드 복사하기

사용자가 생성한 Flex 보드를 복사해 다른 멤버에게 전달할 수 있습니다.

[통합 Flex 보드](#) 메뉴의 [대시보드 목록](#)에서  버튼을 선택하세요. [나에게 복사](#) 또는 [다른 사람에게 복사](#) 옵션을 선택하세요.

- [나에게 복사](#) 옵션을 선택하면 [대시보드 목록](#)에 '_copied' 접미어가 붙은 항목이 추가됩니다.
- [다른 사람에게 복사](#) 옵션을 선택하면 + [계정](#)을 선택하세요.




다른 멤버의 이메일을 선택한 다음 [대시보드 복사하기](#) 버튼을 클릭하세요. 복사 받은 멤버의 [대시보드 목록](#) 목록에 '_copied' 접미어가 붙은 항목이 추가됩니다.

JSON 파일로 공유하기


Flex 보드 설정을 JSON 파일로 저장하고 다른 멤버에서 전달하거나 다른 멤버의 설정을 가져올 수 있습니다.

내보내기

1. [통합 Flex 보드](#) 또는 프로젝트의 [Flex 보드](#) 메뉴로 이동하세요.
2. [대시보드 목록](#)에서  버튼을 선택하세요.
3. JSON 파일이 다운로드되면 공유할 다른 멤버에게 전달하세요.

가져오기

1. [통합 Flex 보드](#) 또는 프로젝트의 [Flex 보드](#) 메뉴로 이동하세요.

2. 화면 오른쪽 위에  가져오기 버튼을 선택하세요.
3. 다운로드한 JSON 파일을 선택하세요.

ⓘ 이 기능은 프로젝트의 수정 권한을 소유한 멤버만 이용할 수 있습니다.

컨테이너 목록

홈 화면 > 프로젝트 선택 > 컨테이너 > 컨테이너 목록

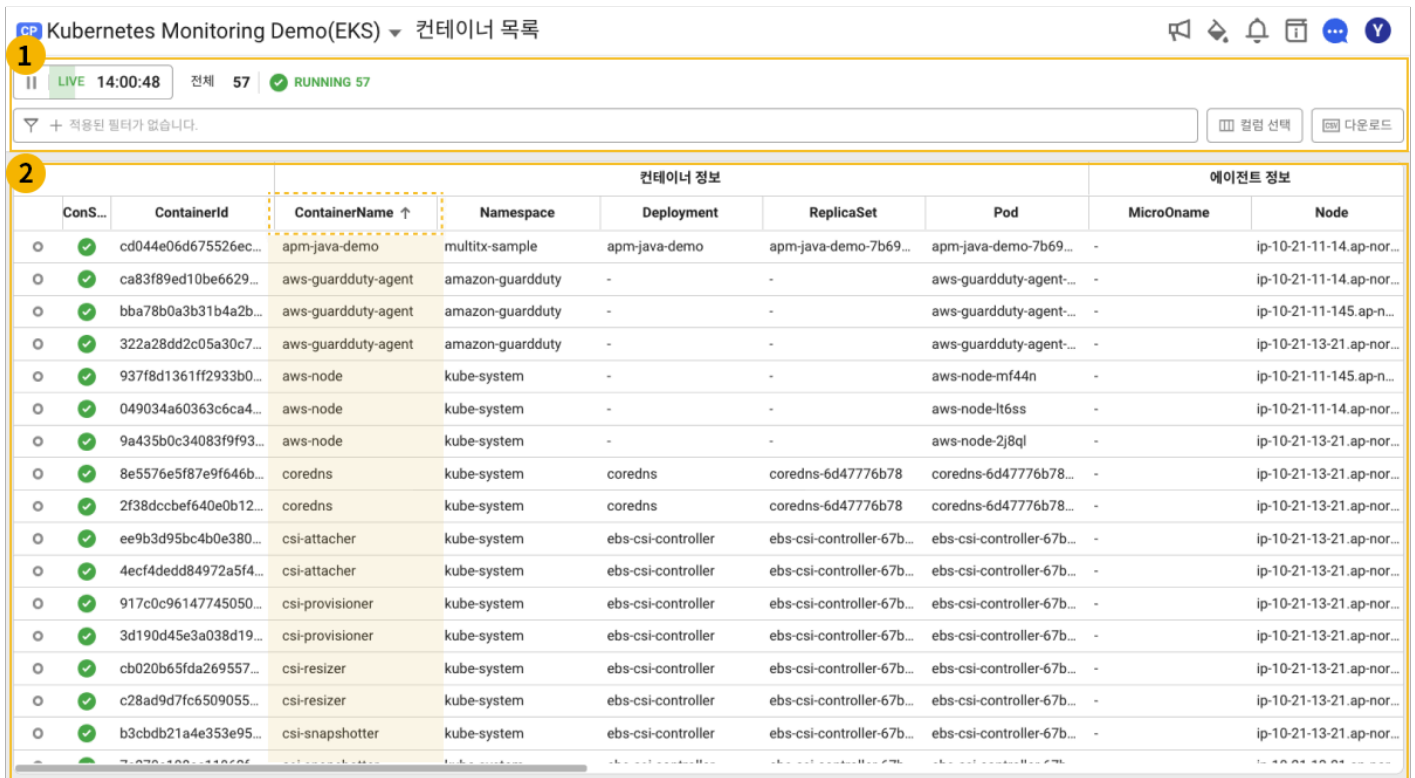
ⓘ 특정 컬럼의 데이터가 표시되지 않는 것은 쿠버네티스 노드 에이전트가 구 버전이기 때문일 수 있습니다. 이런 경우 충분한 데이터가 수집될 수 있도록 쿠버네티스 에이전트 버전을 업데이트하세요.

컨테이너 목록 메뉴를 통해 현재 프로젝트의 컨테이너의 상세 정보를 목록 형식으로 확인할 수 있습니다.

- 10초 주기로 실시간 데이터를 업데이트합니다.
- 컨테이너 상태 요약, 비교, 필터 및 CSV 파일 다운로드 기능을 제공합니다.

CSV 파일명, `kubernetes_contianer_list_{pcode}_{timestamp}.csv`

기본 화면 안내



1 영역에서 원하는 옵션을 선택해 2 영역에서 데이터를 조회할 수 있습니다.

- 1 영역 좌측 상단에서 전체 컨테이너 개수와 상태별 컨테이너 개수를 확인할 수 있습니다.
- 2 영역에서 원하는 컬럼을 선택 시 해당 컬럼 기준으로 컨테이너 목록을 정렬해 조회할 수 있습니다.
- 2 영역 Node 컬럼에서 [노드 정보 보기] 아이콘 선택 시 노드 목록 메뉴로 이동합니다.

메트릭 정보

- 원하는 컬럼 위에 마우스 커서를 위치시키면 다음과 같이 해당 컬럼이 제공하는 메트릭 정보를 확인할 수 있습니다.

MemWs	MemByRequest ↓	MemRequest	MemByLimit	MemLimit
84.23 MiB	87.74 %	MemRequest 1.87 %		192 MiB
258.79 MiB	86.26 %	300 MiB	73.94 %	350 MiB
39.27 MiB	61.35 %	64 MiB	30.68 %	128 MiB

- **컬럼 선택** 옵션 화면 내 컬럼 목록에서 컬럼별 메트릭 정보를 확인할 수 있습니다. 원하는 컬럼 위에 마우스 커서를 위치시키면 컬럼 이름 우측에 ⓘ **인포 아이콘**이 나타납니다. ⓘ **인포 아이콘** 위에 마우스 커서를 위치시키면 다음과 같이 메트릭 정보를 확인할 수 있습니다.

CpuTotUsage
 CpuByRequest
 CpuRequest
 CpuUserByNode
 CpuUserByN... ⓘ

MemWs
 MemByRequest
 MemRequest

NetRxBytes
 NetRxlops
 NetRxError
 NetRxDropped
 NetTxBytes
 NetTxlops
 NetTxError

필터

필터 옵션 활용 시 원하는 컨테이너 목록을 선택해 조회할 수 있습니다. **and** 조건을 통해 여러 개의 필터를 적용할 수 있습니다.

1. ▾ **필터** 입력 창에서 + **더하기 아이콘** 선택 시 다음과 같이 **필터 추가하기** 창이 나타납니다.



- 필터 키와 조건을 선택하세요.
 - 조건에 맞는 값을 선택하세요.
2. 필터 설정 후 적용 버튼을 선택해 해당 필터를 적용하세요.
 3. 여러 개의 필터를 추가하려면 필터 입력 창 또는 필터 추가하기 창에서 + 더하기 아이콘을 선택하세요.

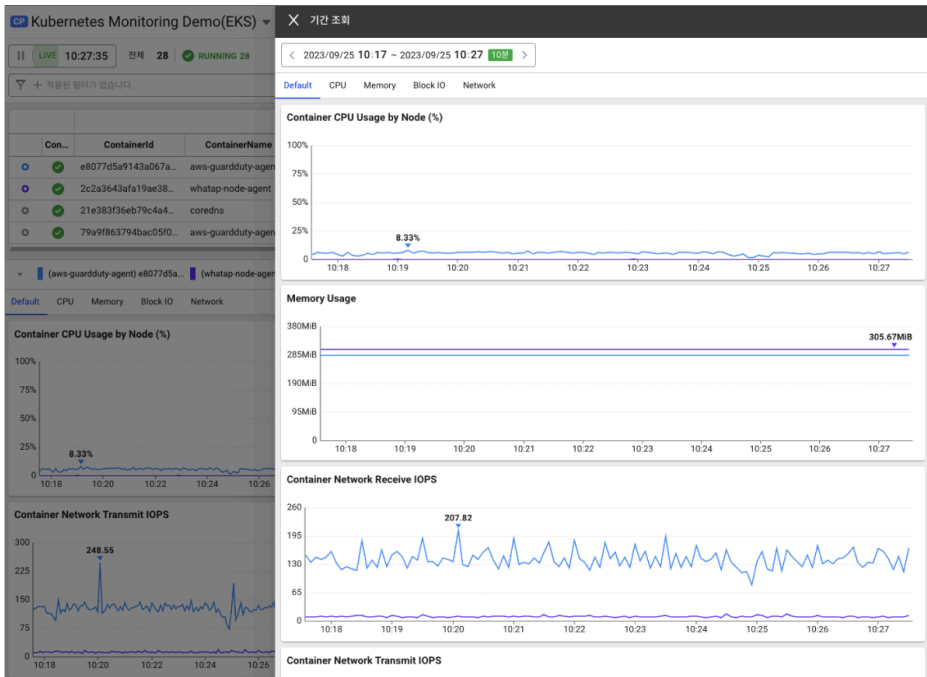
비교하기

비교하기 옵션 활용 시 여러 컨테이너의 주요 지표에 대한 비교 차트를 확인할 수 있습니다.

컨테이너 정보							에이전트 정보	
Con...	ContainerId	ContainerName	Namespace	Deployment	ReplicaSet	Pod	Agent	Node
○	✓	e8077d5a9143a067a...	aws-guardduty-agent	amazon-guardduty	-	-	aws-guardduty-agent...	ip-10-21-13-233.ap...
○	✓	2c2a3643afa19ae38...	whatap-node-agent	whatap-monitoring	-	-	whatap-node-agent-f...	ip-10-21-13-11.ap...
○	✓	21e383f36eb79c4a4...	coredns	kube-system	coredns	coredns-6d47776b78	coredns-6d47776b78...	ip-10-21-11-4.ap-n...
○	✓	79a9f863794bac05f0...	aws-guardduty-agent	amazon-guardduty	-	-	aws-guardduty-agent...	ip-10-21-11-4.ap-n...

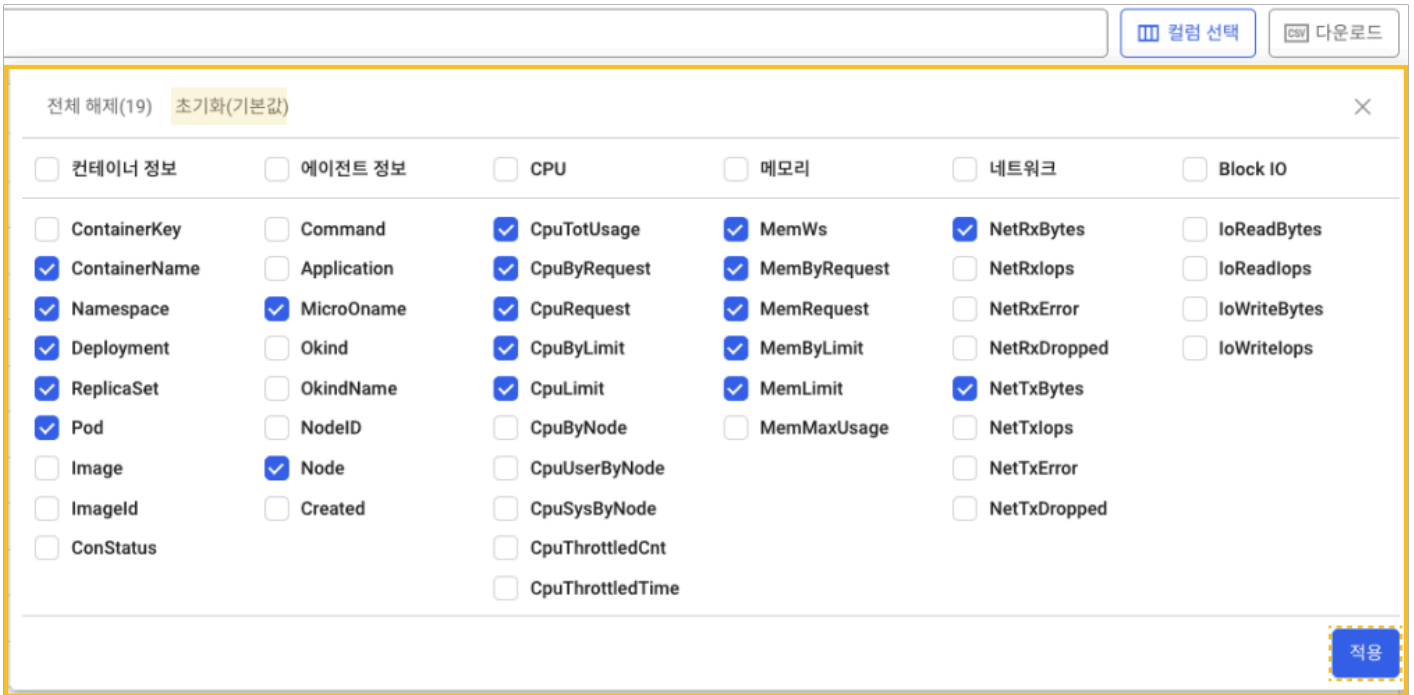
1. a 비교하기 옵션 선택 시 컨테이너 목록에 비교 컬럼이 추가됩니다.

2. 컨테이너 목록 좌측 **b** 비교 컬럼을 통해 비교를 원하는 컨테이너를 선택하세요.
 - 컨테이너는 최대 10개까지 선택할 수 있습니다.
 - 선택한 컨테이너별로 다른 색상이 부여됩니다.
3. **c** 영역에서 선택한 컨테이너의 비교 차트를 확인할 수 있습니다.
 - **b** 비교 컬럼에서 선택한 컨테이너를 **c** 영역 상단에서 확인할 수 있습니다.
 - **c** 상단을 드래그해 비교 차트 영역 높이를 조절할 수 있습니다.
4. **d** 기간 조회 버튼 선택 시 다음과 같이 조회를 원하는 기간 내 비교 차트를 확인할 수 있습니다.



컬럼 선택

컬럼 선택 옵션 활용 시 원하는 컨테이너 정보를 선택해 목록을 조회할 수 있습니다.



- 선택한 컬럼 목록은 메뉴 재진입 시 유지됩니다.
- 초기화 버튼 선택 시 사전 정의된 기본 주요 지표 컬럼이 선택됩니다. 이 경우 기존 컬럼 선택 내역은 삭제됩니다.
- 컬럼은 최대 20개를 선택할 수 있습니다.
- 컬럼 선택 후 우측 하단 적용 버튼을 클릭 해 선택 내역을 적용할 수 있습니다.

컨테이너 볼륨

홈 화면 > 프로젝트 선택 > 컨테이너 > 컨테이너 볼륨

컨테이너 볼륨 메뉴를 통해 컨테이너에 마운트된 볼륨 및 해당 용량 상태를 확인할 수 있습니다.

기본 화면 안내

1 CP Kubernetes Monitoring Demo(EKS) ▾ 컨테이너 볼륨

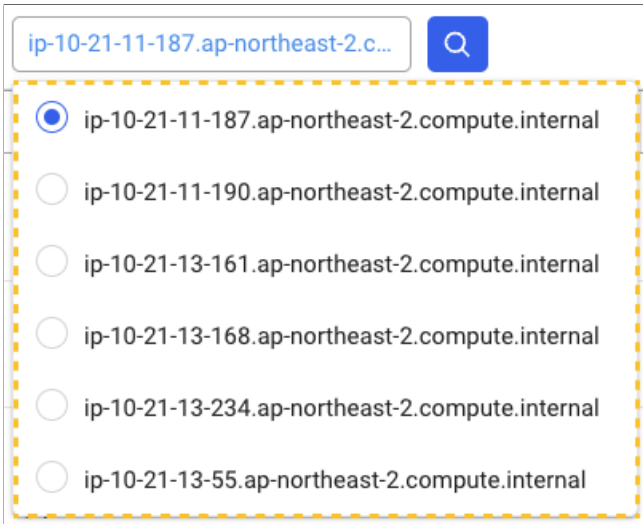
2 에이전트 노드 ID
ip-10-21-11-187.ap-northeast-2.c... 🔍

Containerid	Pod Name	Source	Driver	Mount	Total(Byte)	Used(Byte)	UsedPercent
9dc571e47d7f238f6039694570f9579b9f8d7802c9bb168ac9fac5674d1850b9	whatap-virtual-deployme-nt-68cb9f4d9b-pllp			/	21,462,233,088 Byte	6,009,425,264 Byte	28%
9dc571e47d7f238f6039694570f9579b9f8d7802c9bb168ac9fac5674d1850b9	whatap-virtual-deployme-nt-68cb9f4d9b-pllp	/var/lib/kubelet/pods/aa55a2d6-e1ea-47cb-8f06-3f81cd71dfc0/etc-hosts		/etc/hosts	21,462,233,088 Byte	6,009,425,264 Byte	28%
6cb32b51bc9bad1134e121238f9a6fb1a10d939a14362fbee0299ca4931317a	whatap-node-agent-zmcf			/	21,462,233,088 Byte	6,009,425,264 Byte	28%
6cb32b51bc9bad1134e121238f9a6fb1a10d939a14362fbee0299ca4931317a	whatap-node-agent-zmcf			/rootfs	21,462,233,088 Byte	6,009,425,264 Byte	28%
6cb32b51bc9bad1134e121238f9a6fb1a10d939a14362fbee0299ca4931317a	whatap-node-agent-zmcf			/rootfs/run/containerd/io.containerd.runtime.v2.task/k8s.io/bf88a6f98af2124a4d5065c1b91d9e6476ec469a41e947c0a717b7a4bd50e631/rootfs	21,462,233,088 Byte	6,009,425,264 Byte	28%
6cb32b51bc9bad1134e121238f9a6fb1a10d939a14362fbee0299ca4931317a	whatap-node-agent-zmcf			/rootfs/run/containerd/io.containerd.runtime.v2.task/k8s.io/efc2eeb240a022d6dd50ebb3d61ca08c6af353ac8dfd0c0010acf3f7f20130bc/rootfs	21,462,233,088 Byte	6,009,425,264 Byte	28%

1 영역에서 조회를 원하는 에이전트 노드를 선택 후 2 영역에서 데이터를 확인할 수 있습니다.

노드별 조회

입력창 선택 시 다음과 같이 노드 목록창이 나타납니다. 조회를 원하는 에이전트 노드를 선택 후 🔍 [돋보기](#) 버튼을 클릭하세요.



컨테이너 볼륨 정보

컨테이너 볼륨 메뉴에서 제공하는 정보 항목을 다음과 같이 안내합니다.

항목	설명
ContainerId	컨테이너의 고유 식별자인 컨테이너 ID입니다. 이 ID는 컨테이너 런타임에 의해 생성됩니다. 클러스터 내에서 컨테이너를 식별하고 관리하기 위해 필수적입니다.
Pod Name	컨테이너가 속한 Pod의 이름입니다. 쿠버네티스 클러스터에서는 여러 컨테이너가 하나의 Pod 내에서 실행될 수 있으며, 이 이름을 통해 해당 컨테이너가 어떤 Pod에 속해 있는지 식별할 수 있습니다.
MountType	볼륨이 마운트된 방식을 나타냅니다. 단, tmpfs 는 메모리를 사용하는 임시적인 파일 시스템이기 때문에 해당 컨테이너 볼륨 정보 목록에 표시되지 않습니다. -bind: 호스트 상 특정 디렉토리가 마운트 된 것을 의미합니다. -volume: 호스트 상 Docker가 관리하는 특정 파일 시스템(예: /var/lib/docker/volumes/)이 마운트 된 것을 의미합니다.
Source	호스트 상에서의 해당 볼륨의 경로입니다. 이 경로는 볼륨 데이터가 실제로 저장되는 위치를 의미합니다.
Driver	해당 볼륨 생성 시 사용된 볼륨 드라이버입니다. 볼륨 드라이버는 데이터의 저장 방식과 볼륨의 관리 방법을

항목	설명
	결정합니다. 현재 이 정보는 주로 도커 런타임에서 제공됩니다. containerd와 CRI-O와 같은 더 낮은 수준의 컨테이너 런타임에서는 이러한 고수준 세부 정보를 직접 제공하지 않습니다. 모니터링 대시보드는 런타임 API를 통해 이러한 볼륨 데이터를 제한적으로 제공합니다.
Mount	컨테이너 내에서 볼륨이 마운트된 경로입니다. 이 경로를 통해 컨테이너 내의 애플리케이션이 볼륨에 저장된 데이터에 접근할 수 있습니다.
Total(Byte)	볼륨의 전체 용량을 바이트 단위로 나타냅니다. 이는 볼륨에 저장할 수 있는 데이터의 최대 크기를 의미합니다.
Used(Byte)	현재 볼륨에서 사용 중인 용량을 바이트 단위로 나타냅니다. 이는 볼륨 내에 실제로 저장된 데이터의 크기를 의미합니다.
UsedPercent	볼륨 용량의 사용률을 백분율로 나타냅니다. 이 비율은 볼륨의 전체 용량 대비 현재 사용 중인 용량의 비율을 보여줍니다. 사용률이 높아질수록 볼륨 공간이 부족해지고 있음을 의미합니다. 이는 성능 저하나 데이터 손실 위험을 알리는 중요한 지표가 될 수 있습니다.

컨테이너 이미지

홈 화면 > 프로젝트 선택 > 컨테이너 > 컨테이너 이미지

컨테이너 이미지 메뉴를 통해 컨테이너에서 사용하고 있는 이미지 정보와 해당 이미지를 기반으로 실행 중인 컨테이너의 개수 및 CPU, Memory 사용 총량을 목록 형식으로 확인할 수 있습니다.

CP Kubernetes Monitoring Demo(EKS) ▾ 컨테이너 이미지

LIVE 11:04:55 전체 21 + 적용된 필터가 없습니다. b 다운로드

Registry	Name	Tag	Repo Digest	Containers	Total Container CPU	Total Container Memory
a 필터 추가하기						
필터 키		조건				
Registry						
Name						
Tag		적용				
602401143452.dkr.ecr.ap-nor...	eks/csi-provisioner	v3.5.0-eks-1-28-6	sha256:8ccbc5ae325687c39...	6	1064 m	971.32 MiB
602401143452.dkr.ecr.ap-nor...	eks/csi-resizer	v1.9.0-eks-1-28-6	sha256:fa1d851c35134381d...	6	22 m	182.71 MiB
602401143452.dkr.ecr.ap-nor...	eks/csi-snapshotter	v6.3.0-eks-1-28-6	sha256:2fbd620d25bda985b...	8	2 m	60.95 MiB
602401143452.dkr.ecr.ap-nor...	eks/eksbuild.1		sha256:16ebda509ca6f641...	6	8 m	58.10 MiB
602401143452.dkr.ecr.ap-nor...	eks/eksbuild.1		sha256:c2ccb4b5e1f29bd91...	6	0 m	103.71 MiB
602401143452.dkr.ecr.ap-nor...	eks/eksbuild.1		sha256:9d7ac873c0a88dc77...	14	324 m	1.96 GiB
602401143452.dkr.ecr.ap-nor...	eks/eksbuild.1		sha256:3dba81d0c3c8ce51a...	8	0 m	100.10 MiB
602401143452.dkr.ecr.ap-nor...	eks/eksbuild.1		sha256:ad924171bfd55464...	2	0 m	19.23 MiB
602401143452.dkr.ecr.ap-nor...	eks/eksbuild.1		sha256:867f945ad962ef458f...	2	2 m	25.41 MiB
602401143452.dkr.ecr.ap-nor...	eks/eksbuild.1		sha256:3207206f677adff845...	2	2 m	19.77 MiB
602401143452.dkr.ecr.ap-nor...	eks/eksbuild.1		sha256:de9c28e2d8eb849b1...	2	0 m	19.29 MiB
docker.io	whatap/virtual	latest	sha256:54cd72a83ee3dba65...	10	681 m	1.75 GiB
592247757306.dkr.ecr.ap-nor...	apm-php-demo-mariadb	main	sha256:712ebee8166a09d4b...	1	0 m	122.77 MiB
592247757306.dkr.ecr.ap-nor...	apm-php-demo-wordpr...	main	sha256:7509810d2fefefee8f...	1	0 m	179.56 MiB
602401143452.dkr.ecr.ap-nor...	eks/coredns	v1.10.1-eksbuild.1	sha256:29e5328ddaad2dce9...	2	0 m	31.48 MiB
592247757306.dkr.ecr.ap-nor...	apm-php-demo-postgres	main	sha256:5dbac2284cc79e651...	1	0 m	25.24 MiB
592247757306.dkr.ecr.ap-nor...	apm-java-demo	main	sha256:48661165f2bdc3182...	1	4 m	321.60 MiB
592247757306.dkr.ecr.ap-nor...	nodejs_sample.v2	main	sha256:75d76a0fe3261c7ca...	1	222 m	74.70 MiB
592247757306.dkr.ecr.ap-nor...	apm-php-demo-apache	main	sha256:4ae5ada099d3d05da...	1	8 m	90.77 MiB
592247757306.dkr.ecr.ap-nor...	php-demo-runtime	php7.3-apache-wp-5.8.3	sha256:361ffffccb30f7db0dd...	1	0 m	68.79 MiB

a 필터


필터 창에서 필터 키 선택 시 Registry, Name, Tag, Repo Digest에 대한 필터 기능을 제공합니다.

b CSV 다운로드

상단 오른쪽의 CSV 다운로드 버튼 선택 시 원본 데이터를 CSV 형식의 파일로 다운로드할 수 있습니다. 파일명은 kubernetes_container_images_ 뒤에 프로젝트 코드와 데이터 조회 시점의 타임스탬프를 덧붙여 생성합니다.

예, `kubernetes_container_images_{pcode}_{timestamp}.csv`

C 컨테이너 상세 보기

Name 컬럼 오른쪽  [컨테이너 상세 보기](#) 아이콘 선택 시 해당 이미지가 검색된 [컨테이너 목록](#) 메뉴로 이동합니다.

Out Of Memory Killed 컨테이너

홈 화면 > 프로젝트 선택 > 컨테이너 > Out Of Memory Killed 컨테이너

Out Of Memory Killed 컨테이너 메뉴를 통해 **Out Of Memory(OOM)**가 발생한 컨테이너를 빠르게 검색하고 해당 시점의 메모리 사용량의 변화 추이를 확인할 수 있습니다. 변화 추이는 급격하거나 점진적일 수 있습니다. 이를 확인해 적절한 POD 수와 메모리 임계점에 대한 객관적 판단이 가능합니다.

⚠ OOM 발생 예시

컨테이너의 메모리 사용량이 메모리 제한을 초과하거나, 컨테이너가 실행 중인 노드의 메모리가 부족할 경우 Out Of Memory가 발생합니다. 만약 컨테이너가 코드 137 로 종료되면 Out Of Memory에 의해 SIGKILL 을 수신한 것입니다.

기본 화면 안내

CP Kubernetes Monitoring Demo(New) ▾ Out Of Memory Killed 컨테이너

< 2023/08/23 00:00 ~ 2023/08/24 00:00 1일 >

OomKilledAt	name	podName	containerId	created	command	imageName	onodeName	state	memLimit	namespace	memory_workingset
2023/08/23 13:45	virtual	whatap-virtual-dej	20c75019-f2eb-4f	2023/07/28 17:4E		docker.io/whatap, demo-k8s-worker		RUNNING	268435456	virtual-main	
2023/08/23 13:51	virtual	whatap-virtual-dej	b07cb43f-7cdd-4f	2023/08/11 13:44		docker.io/whatap, demo-k8s-worker		RUNNING	268435456	virtual-main	
2023/08/23 18:34	virtual	whatap-virtual-dej	f89de3ef-abf6-4d	2023/07/28 17:4E		docker.io/whatap, demo-k8s-worker		RUNNING	268435456	virtual-main	
2023/08/23 18:35	virtual	whatap-virtual-dej	b649f550-00f6-43	2023/08/11 13:44		docker.io/whatap, demo-k8s-worker		RUNNING	268435456	virtual-main	
2023/08/23 18:45	virtual	whatap-virtual-dej	13e3a13b-d57c-4	2023/08/11 13:44		docker.io/whatap, demo-k8s-worker		RUNNING	268435456	virtual-main	

- 좌측 상단의 시간 선택자를 클릭해 조회를 원하는 시간대를 선택할 수 있습니다. 최대 2일까지 조회할 수 있습니다.
- OOM Killed 컨테이너 정보를 목록 형식으로 제공합니다. 자세한 내용은 다음 안내를 참조하세요.

OOM Killed 컨테이너 정보

Out Of Memory Killed 컨테이너 메뉴에서 제공하는 컬럼 정보를 다음과 같이 안내합니다.

컬럼	의미
OomKilledAt	Out Of Memory Killed 가 발생한 TimeStamp
name	컨테이너의 이름
podName	컨테이너가 속한 Pod의 이름
containerId	컨테이너의 ID
created	컨테이너가 생성된 TimeStamp
command	컨테이너의 실행 명령어
imageName	컨테이너의 이미지의 이름
onodeName	컨테이너가 작동 중인 노드 이름
state	컨테이너의 상태
memLimit	컨테이너의 메모리 Limit 크기
namespace	컨테이너의 Pod가 속한 namespace 의 이름
memory_workingset	컨테이너에서 비활성화 된 메모리를 제외한 실제 사용 중인 메모리

Pod 목록

홈 화면 > 프로젝트 선택 > 워크로드 > Pod 목록

Pod 목록을 조회합니다. 원하는 컬럼 선택 시 해당 컬럼 기준으로 Pod 목록을 정렬해 조회할 수 있습니다.

CP koogle-dev-project ▾ Pod 목록

|| LIVE 19:13:31 전체 34 Running 34

+ 적용된 필터가 없습니다. [컬럼 선택] [다운로드]

파드 정보									
PodP...	Pod	PodIp	Namespace	Node	ConRestartCnt	Age	CpuByLimit	CpuLimitByNode	
✓	amex-6ccf9d89f5-gb...	10.21.120.39	titans	ip-10-21-120-186.ap...	0	14d 17h 37m 35s	0%	4.50%	0%
✓	api-call-669cbcf45f-k...	10.21.120.19	sample	ip-10-21-120-186.ap...	0	23d 19h 57m 35s	0.11%	100%	0.11
✓	aws-node-cq6t6	10.21.110.129	kube-system	ip-10-21-110-129.ap...	0	25d 15h 00m 16s	0.31%	100%	0.31
✓	aws-node-xljik	10.21.120.186	kube-system	ip-10-21-120-186.ap...	0	25d 06h 24m 15s	0.20%	100%	0.20
✓	aws-node-zcf2b	10.21.120.134	kube-system	ip-10-21-120-134.ap...	0	17d 11h 29m 01s	0.15%	100%	0.15
✓	coredns-76b4dcc5cc-...	10.21.120.223	kube-system	ip-10-21-120-186.ap...	0	29d 20h 57m 35s	0.10%	100%	0.10
✓	coredns-76b4dcc5cc-...	10.21.120.203	kube-system	ip-10-21-120-186.ap...	0	29d 20h 57m 35s	0%	100%	0%
✓	customer-7bbd6f949...	10.21.120.187	titans	ip-10-21-120-134.ap...	100	17d 01h 12m 21s	0%	0%	0%
✓	datadog-koogle-8827k	10.21.120.21	default	ip-10-21-120-186.ap...	0	8d 23h 50m 55s	0.99%	400%	3.95
✓	datadog-koogle-clust...	10.21.120.231	default	ip-10-21-120-186.ap...	0	8d 19h 40m 55s	0%	100%	0%
✓	datadog-koogle-xh2gp	10.21.110.89	default	ip-10-21-110-129.ap...	0	25d 12h 13m 36s	1.10%	400%	4.39
✓	datadog-koogle-zqtgd	10.21.120.242	default	ip-10-21-120-134.ap...	0	17d 08h 42m 21s	0.71%	400%	2.84
✓	db-transaction-6f7d9...	10.21.120.72	sample	ip-10-21-120-186.ap...	0	14d 00h 40m 55s	0%	100%	0%
✓	demo-7c97d7b78c-7...	10.21.120.132	default	ip-10-21-120-186.ap...	0	21d 12h 40m 55s	0.20%	100%	0.20
✓	dmz-router-extend-13...	10.21.110.13	titans	ip-10-21-110-129.ap...	0	11d 11h 33m 36s	2.99%	3.50%	0.10

- 10초 주기로 실시간 데이터를 업데이트합니다.
- 좌측 상단에서 전체 Pod 개수와 상태별 Pod 개수를 확인할 수 있습니다.
- 검색, 컬럼 정렬, 상세 보기, CSV 파일 다운로드를 제공합니다.

CSV 파일명, `kubernetes_pod_list_{pcode}_{timestamp}.csv`

① [노드 목록](#)과 [컨테이너 목록](#) 및 [Pod 목록](#)과 [컨테이너 애플리케이션 목록](#) 등 목록형 메뉴의 **메트릭 정보**, **필터**, **컬럼 선택** 안내는 동일합니다.

메트릭 정보

- 원하는 컬럼 위에 마우스 커서를 위치시키면 다음과 같이 해당 컬럼이 제공하는 메트릭 정보를 확인할 수 있습니다.

	CPU	MemoryUsedPct	
		Field Unit pct	
ContainerRuntime	Cpu	MemoryUsedPct	IoPerce
ontainerd://1.6.19	16.88 %	18.88 %	...gent/1/
ontainerd://1.6.19	12.22 %	17.39 %	...gent/1/
ontainerd://1.6.19	16.35 %	17.65 %	...gent/1/

- 컬럼 선택** 옵션 화면 내 컬럼 목록에서 컬럼별 메트릭 정보를 확인할 수 있습니다. 원하는 컬럼 위에 마우스 커서를 위치시키면 컬럼 이름 우측에 ⓘ **인포 아이콘**이 나타납니다. ⓘ **인포 아이콘** 위에 마우스 커서를 위치시키면 다음과 같이 메트릭 정보를 확인할 수 있습니다.

<input type="checkbox"/> OS	<input type="checkbox"/> CpuLoad15	<input type="checkbox"/> MemorySwapTot...	<input type="checkbox"/> UsedInodePercent
<input checked="" type="checkbox"/> Image	<input type="checkbox"/> CpuNice	<input type="checkbox"/> MemorySwapUs...	<input type="checkbox"/> IoPercent
<input type="checkbox"/> Kernel	<input type="checkbox"/> CpuSoftirq	<input type="checkbox"/> MemoryTotal	<input checked="" type="checkbox"/> IoPercentMax
<input type="checkbox"/> KubeProxy	<input type="checkbox"/> CpuSteal	<input checked="" type="checkbox"/> MemoryUsedPct	<input type="checkbox"/> QueueLength
<input checked="" type="checkbox"/> Kubelet	<input type="checkbox"/> CpuUsr		<input type="checkbox"/> ReadBps
<input checked="" type="checkbox"/> ContainerRuntime	<input checked="" type="checkbox"/> Field Unit pct		<input type="checkbox"/> Readlops
	<input checked="" type="checkbox"/> CpuUsr ⓘ		<input type="checkbox"/> TotalSpace

필터

필터 옵션 활용 시 원하는 Pod 목록을 선택해 조회할 수 있습니다. **and** 조건을 통해 여러 개의 필터를 적용할 수 있습니다.

- ▽ **필터** 입력 창에서 + **더하기 아이콘** 선택 시 다음과 같이 **필터 추가하기** 창이 나타납니다.



- **필터 키**와 **조건**을 선택하세요.
 - 조건에 맞는 값을 선택하세요.
2. 필터 설정 후 **적용** 버튼을 선택해 해당 필터를 적용하세요.
 3. 여러 개의 필터를 추가하려면 **필터** 입력 창 또는 **필터 추가하기** 창에서 **+ 더하기 아이콘**을 선택하세요.

컬럼 선택

컬럼 선택 옵션 활용 시 원하는 Pod 정보를 선택해 목록을 조회할 수 있습니다.

전체 해제(10) 초기화(기본값)

노드 정보	CPU	메모리	디스크	네트워크
<input type="checkbox"/> CapacityCpu	<input type="checkbox"/> Cpuidle	<input type="checkbox"/> MemoryBuffers	<input type="checkbox"/> BlkSize	<input type="checkbox"/> DroppedIn
<input checked="" type="checkbox"/> Uptime	<input type="checkbox"/> CpuIOWait	<input type="checkbox"/> MemoryCached	<input type="checkbox"/> FreePercent	<input type="checkbox"/> DroppedOut
<input checked="" type="checkbox"/> Age	<input type="checkbox"/> CpuIrq	<input type="checkbox"/> MemoryFree	<input type="checkbox"/> FreeSpace	<input type="checkbox"/> ErrorIn
<input type="checkbox"/> Architecture	<input type="checkbox"/> CpuLoad1	<input type="checkbox"/> MemoryPageFau...	<input type="checkbox"/> TotalNode	<input type="checkbox"/> ErrorOut
<input type="checkbox"/> CapacityMemory	<input type="checkbox"/> CpuLoad5	<input type="checkbox"/> MemoryShared	<input type="checkbox"/> UsedNode	<input type="checkbox"/> HwAddr
<input type="checkbox"/> OS	<input type="checkbox"/> CpuLoad15	<input type="checkbox"/> MemorySwapTot...	<input type="checkbox"/> UsedNodePercent	<input type="checkbox"/> PacketIn
<input checked="" type="checkbox"/> Image	<input type="checkbox"/> CpuNice	<input type="checkbox"/> MemorySwapUs...	<input type="checkbox"/> IoPercent	<input type="checkbox"/> PacketOut
<input type="checkbox"/> Kernel	<input type="checkbox"/> CpuSoftIrq	<input type="checkbox"/> MemoryTotal	<input checked="" type="checkbox"/> IoPercentMax	<input type="checkbox"/> TrafficUsedSize
<input type="checkbox"/> KubeProxy	<input type="checkbox"/> CpuSteal	<input checked="" type="checkbox"/> MemoryUsedPct	<input type="checkbox"/> QueueLength	<input checked="" type="checkbox"/> TrafficUsedSize...
<input checked="" type="checkbox"/> Kubelet	<input type="checkbox"/> CpuSys	<input type="checkbox"/> ReadBps	<input type="checkbox"/> ReadOps	<input type="checkbox"/> TrafficIn
<input checked="" type="checkbox"/> ContainerRuntime	<input checked="" type="checkbox"/> Cpu...	<input type="checkbox"/> ReadOps	<input type="checkbox"/> TrafficOut	

적용

- 선택한 컬럼 목록은 메뉴 재진입 시 유지됩니다.
- 초기화 버튼 선택 시 사전 정의된 기본 주요 지표 컬럼이 선택됩니다. 이 경우 기존 컬럼 선택 내역은 삭제됩니다.
- 컬럼은 최대 20개를 선택할 수 있습니다.
- 컬럼 선택 후 우측 하단 적용 버튼을 클릭 해 선택 내역을 적용할 수 있습니다.

상세 보기

목록에서 조회를 원하는 Pod의 🔍 상세 보기 아이콘 선택 시 해당 Pod에 대한 상세 정보를 확인할 수 있습니다.

ⓘ 상세 보기 창의 [트레이스](#), [메트릭스](#), [로그](#), [이벤트](#) 탭의 경우 컨테이너 맵 상세 보기와 동일합니다.

리소스

대상 선택 후 상세 보기의 [리소스](#) 탭에서 선택한 리소스와 관련된 리소스 목록 및 관계도를 조회할 수 있습니다.

X amex-6ccf9d89f5-gbc24

리소스 메트릭스 로그 이벤트

▶ < 2024/01/10 19:16 >

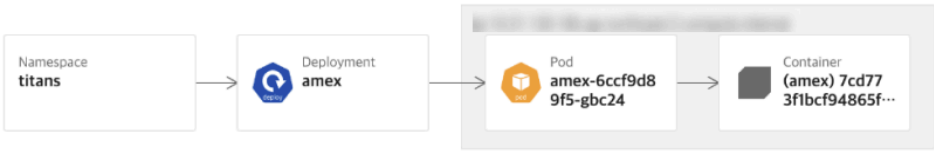
컨테이너 → 컨테이너 목록으로 이동

ConS...	ContainerId	ContainerName	Image	ConStatus	Pod	CpuByLimit	MemByLimit
✔	7cd773f1bcf948...	amex	docker.io/library/tomc...	Up 4 hours 9 minutes	amex-6ccf9d89f5-gbc24	0% <div style="width: 100%; height: 10px; background-color: #ccc;"></div>	71.94% <div style="width: 71.94%; height: 10px; background-color: #007bff;"></div>

노드 → 노드 목록으로 이동

Status	OnodeName	Age	Image	CapacityCpu	CapacityMemory	Cpu	MemoryUsedPct
✔	ip-10-21-120-186...	19d 07h 44m 16s	Amazon Linux 2	2	15.33 GiB	8.46% <div style="width: 8.46%; height: 10px; background-color: #007bff;"></div>	30.25% <div style="width: 30.25%; height: 10px; background-color: #007bff;"></div>

리소스 관계



```

    graph LR
      titans[Namespace titans] --> amex[Deployment amex]
      amex --> pod[Pod amex-6ccf9d89f5-gbc24]
      pod --> container["Container (amex) 7cd773f1bcf94865f..."]
    
```

요약 정보

선택한 리소스와 관련된 리소스 목록의 요약 정보를 제공합니다. 리소스 명 옆에 상세보기 아이콘을 선택 시 해당 리소스 목록 화면으로 이동합니다.

- **컨테이너**: 해당 컨테이너만 필터링된 상태의 [컨테이너 목록](#)으로 이동합니다.
- **Pod**: 해당 파드만 필터링된 [Pod 목록](#)으로 이동합니다.
- **노드**: 해당 노드 이름이 선택된 상태의 [노드 상세](#)로 이동합니다.

리소스 관계도

리소스간 연결 관계와 상태를 시각적으로 확인할 수 있습니다. 목록과 동일하게 Pod, 컨테이너, 애플리케이션에 마우스 오버 시 상세 지표를 확인할 수 있습니다. 리소스 선택 시 해당 리소스 목록 페이지로 이동합니다.

컨테이너 애플리케이션 목록

홈 화면 > 프로젝트 선택 > 워크로드 > 컨테이너 애플리케이션 목록

컨테이너화된 애플리케이션의 목록을 조회합니다. 원하는 컬럼 선택 시 해당 컬럼 기준으로 애플리케이션 목록을 정렬해 조회할 수 있습니다.

CP Kubernetes Monitoring Demo(EKS) ▾ 컨테이너 애플리케이션 목록

LIVE 10:52:22 + 적용된 필터가 없습니다. 컬럼 선택 CSV

Oname	애플리케이션		사용자		액티브 트랜잭션		
	OkindName	Type	ConcurrentUser	DbConnection	HttpCall	Method	
virtual1	virtual	java	994	2	21	17	
virtual12	virtual	java	996	3	26	22	
virtual15	virtual	java	999	0	22	22	
virtual2	virtual	java	986	0	29	29	
virtual3	virtual	java	999	3	25	23	
virtual4	virtual	java	981	1	19	18	
virtual5	virtual	java	996	2	21	20	
virtual6	virtual	java	991	3	26	21	
virtual7	virtual	java	996	3	24	23	
virtual8	virtual	java	994	0	18	19	

- 10초 주기로 실시간 데이터를 업데이트합니다.
- 검색, 컬럼 정렬, 상세 보기, CSV 파일 다운로드를 제공합니다.

CSV 파일명, `kubernetes_application_list_{pcode}_{timestamp}.csv`

ⓘ 노드 목록과 컨테이너 목록 및 Pod 목록과 컨테이너 애플리케이션 목록 등 목록형 메뉴의 메트릭 정보, 필터, 컬럼 선택 안내는 동일합니다.

메트릭 정보

- 원하는 컬럼 위에 마우스 커서를 위치시키면 다음과 같이 해당 컬럼이 제공하는 메트릭 정보를 확인할 수 있습니다.

	CPU	MemoryUsedPct	
ContainerRuntime	Cpu	MemoryUsedPct	IoPerce
ontainerd://1.6.19	16.88 %	18.88 %	...gent/1/
ontainerd://1.6.19	12.22 %	17.39 %	...gent/1/
ontainerd://1.6.19	16.35 %	17.65 %	...gent/1/

- **컬럼 선택** 옵션 화면 내 컬럼 목록에서 컬럼별 메트릭 정보를 확인할 수 있습니다. 원하는 컬럼 위에 마우스 커서를 위치시키면 컬럼 이름 우측에 ⓘ **인포 아이콘**이 나타납니다. ⓘ **인포 아이콘** 위에 마우스 커서를 위치시키면 다음과 같이 메트릭 정보를 확인할 수 있습니다.

<input type="checkbox"/> OS	<input type="checkbox"/> CpuLoad15	<input type="checkbox"/> MemorySwapTot...	<input type="checkbox"/> UsedInodePercent
<input checked="" type="checkbox"/> Image	<input type="checkbox"/> CpuNice	<input type="checkbox"/> MemorySwapUs...	<input type="checkbox"/> IoPercent
<input type="checkbox"/> Kernel	<input type="checkbox"/> CpuSoftlirq	<input type="checkbox"/> MemoryTotal	<input checked="" type="checkbox"/> IoPercentMax
<input type="checkbox"/> KubeProxy	<input type="checkbox"/> CpuSteal	<input checked="" type="checkbox"/> MemoryUsedPct	<input type="checkbox"/> QueueLength
<input checked="" type="checkbox"/> Kubelet	<input checked="" type="checkbox"/> CpuUsr		<input type="checkbox"/> ReadBps
<input checked="" type="checkbox"/> ContainerRuntime	<input checked="" type="checkbox"/> Field Unit pct		<input type="checkbox"/> Readlops
	<input checked="" type="checkbox"/> CpuUsr ⓘ		<input type="checkbox"/> TotalSpace

필터

필터 옵션 활용 시 원하는 애플리케이션 목록을 선택해 조회할 수 있습니다. **and** 조건을 통해 여러 개의 필터를 적용할 수 있습니다.

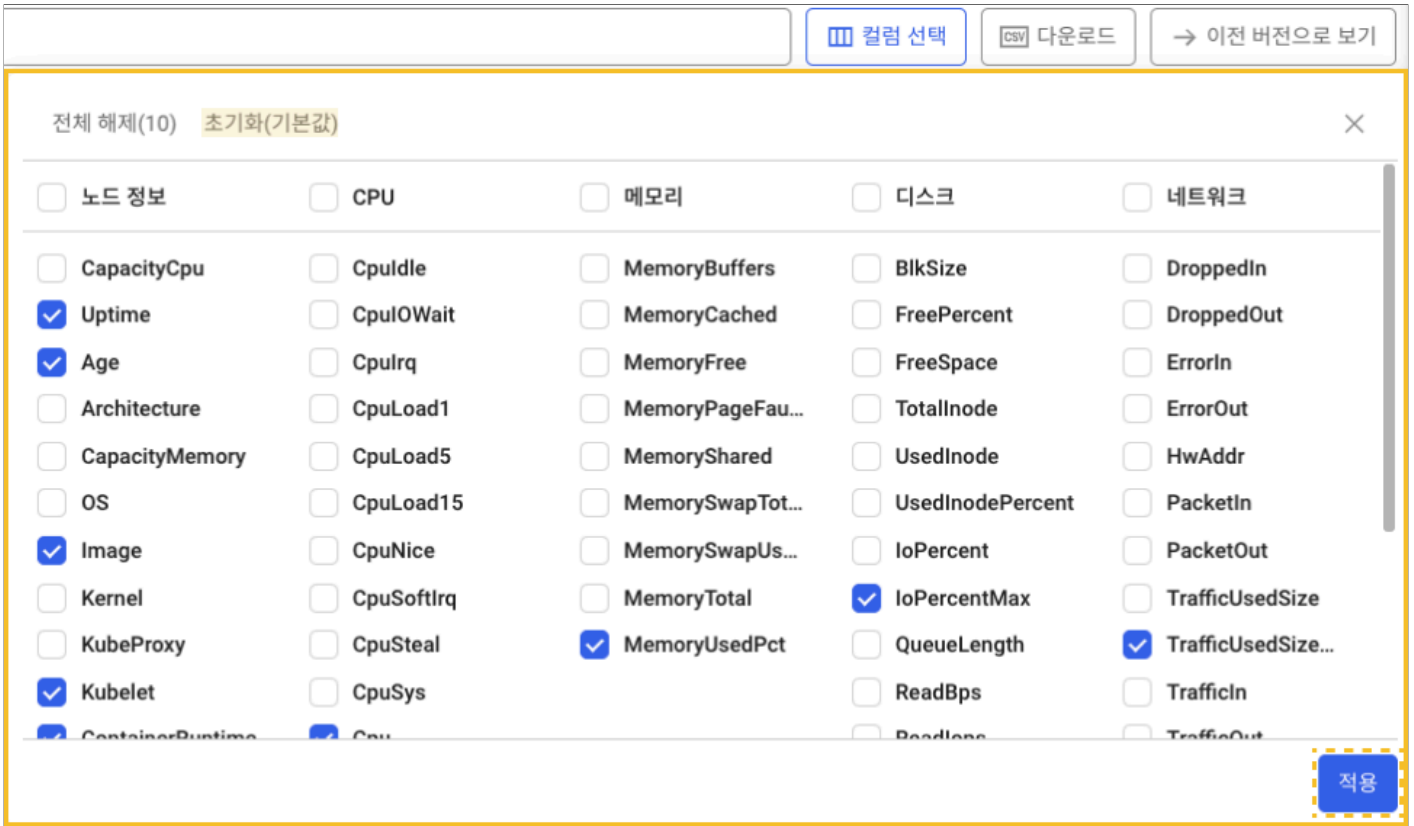
1. ▾ **필터** 입력 창에서 + **더하기 아이콘** 선택 시 다음과 같이 **필터 추가하기** 창이 나타납니다.



- **필터 키**와 **조건**을 선택하세요.
 - 조건에 맞는 값을 선택하세요.
2. 필터 설정 후 **적용** 버튼을 선택해 해당 필터를 적용하세요.
 3. 여러 개의 필터를 추가하려면 **필터** 입력 창 또는 **필터 추가하기** 창에서 **+ 더하기 아이콘**을 선택하세요.

컬럼 선택

컬럼 선택 옵션 활용 시 원하는 애플리케이션 목록 정보를 선택해 목록을 조회할 수 있습니다.



- 선택한 컬럼 목록은 메뉴 재진입 시 유지됩니다.
- 초기화 버튼 선택 시 사전 정의된 기본 주요 지표 컬럼이 선택됩니다. 이 경우 기존 컬럼 선택 내역은 삭제됩니다.
- 컬럼은 최대 20개를 선택할 수 있습니다.
- 컬럼 선택 후 우측 하단 적용 버튼을 클릭 해 선택 내역을 적용할 수 있습니다.

상세 보기

목록에서 조회를 원하는 애플리케이션의 🔍 상세 보기 아이콘 선택 시 해당 애플리케이션에 대한 상세 정보를 확인할 수 있습니다.

① 상세 보기 창의 [트레이스](#), [메트릭스](#), [로그](#), [이벤트](#), [호출 정보](#) 탭의 경우 컨테이너 맵 상세 보기와 동일합니다.

리소스

대상 선택 후 상세 보기의 [리소스](#) 탭에서 선택한 리소스와 관련된 리소스 목록 및 관계도를 조회할 수 있습니다.

✕
virtual1

[리소스](#) | [트레이스](#) | [메트릭스](#) | [로그](#) | [이벤트](#) | [호출 정보](#)

2024/03/15 09:26

컨테이너 → 컨테이너 목록으로 이동

ContainerId	ContainerName	Image	ConStatus	Pod	CpuByLimit	MemByLimit
853d2618dd518...	virtual	docker.io/whatap/virtu...	Up 9 hours 12 minutes	whatap-virtual-deploy...	25.26%	67.49%

Pod → Pod 목록으로 이동

Pod	containerIdsCount	ReplicaSet	Deployment	Namespace	CpuByLimit	MemByLimit
whatap-virtual-d...	1	whatap-virtual-deploy...	whatap-virtual-deploy...	virtual-main	25.26%	67.49%

노드 → 노드 목록으로 이동

OnodeName	Age	Image	CapacityCpu	CapacityMemory	Cpu	MemoryUsedPct
ip-10-21-13-157...	1d 09h 14m 40s	Amazon Linux 2	2	1.88GiB	9.60%	66.05%

리소스 관계

```

graph LR
    NS[Namespace virtual-main] --> D[Deployment whatap-virtual-deployment]
    D --> RS[ReplicaSet whatap-virtual-deployment]
    RS --> P[Pod whatap-virtual-deployment]
    P --> C[Container (virtual) 853d2618dd518a...]
    subgraph Node [ip-10-21-13-157.ap-northeast-2.compute.internal]
        C
    end
  
```

요약 정보

선택한 리소스와 관련된 리소스 목록의 요약 정보를 제공합니다. 리소스 명 옆에 상세보기 아이콘을 선택 시 해당 리소스 목록 화면으로 이동합니다.

- **컨테이너**: 해당 컨테이너만 필터링된 상태의 [컨테이너 목록](#)으로 이동합니다.
- **Pod**: 해당 파드만 필터링된 [Pod 목록](#)으로 이동합니다.
- **노드**: 해당 노드 이름이 선택된 상태의 [노드 상세](#)로 이동합니다.

리소스 관계도

리소스간 연결 관계와 상태를 시각적으로 확인할 수 있습니다. 목록과 동일하게 Pod, 컨테이너, 애플리케이션에 마우스 오버 시 상세 지표를 확인할 수 있습니다. 리소스 선택 시 해당 리소스 목록 페이지로 이동합니다.

Deployment 목록

홈 화면 > 프로젝트 선택 > 워크로드 > Deployment 목록

Deployment의 목록을 조회합니다. 여러 Pod로 구성된 애플리케이션의 전체적인 운영 상태를 모니터링해 시스템의 안정성과 성능을 유지하도록 돕습니다. [Deployment 목록](#) 내 원하는 컬럼 선택 시 해당 컬럼 기준으로 목록을 정렬해 조회할 수 있습니다.

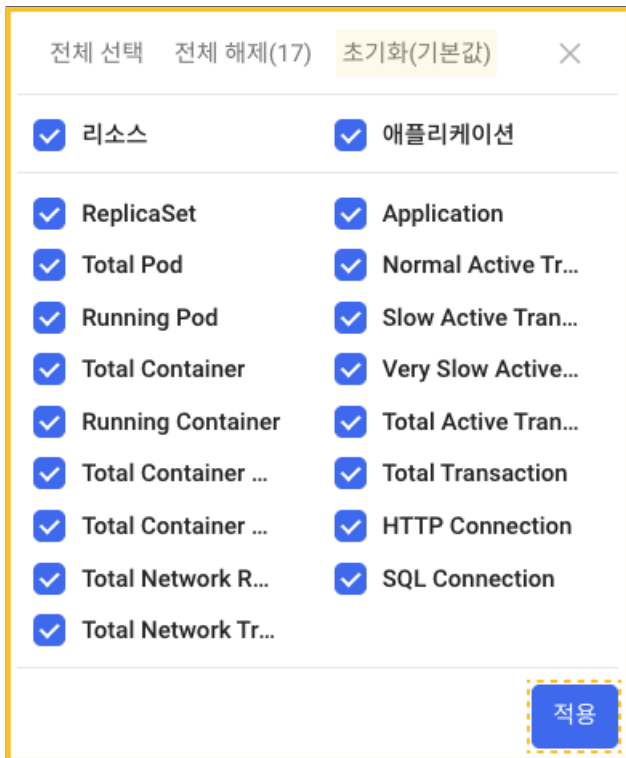
Deployment	ReplicaSet	Total Pod	Running Pod	리소스			
				Total Container	Running Container	Total Container CPU	Total C
apm-java-demo	1	1	1	1	1	4m	
coredns	1	2	2	2	2	0m	
ebs-csi-controll...	1	2	2	12	12	0m	
nodejs-sample...	1	1	1	1	1	19m	
php-demo-apa...	1	1	1	1	1	8m	
php-demo-wor...	1	1	1	1	1	0m	
testing-http-se...	1	1	1	1	1	8m	
whatap-master...	1	1	1	2	2	80m	
whatap-virtual...	1	10	10	10	10	649m	

- 10초 주기로 실시간 데이터를 업데이트합니다.
- 검색, 컬럼 정렬, 상세 보기, CSV 파일 다운로드를 제공합니다.

CSV 파일명, `kubernetes_deployment_list_{pcode}_{timestamp}.csv`

컬럼 선택

[컬럼 선택](#) 옵션 활용 시 원하는 Deployment 정보를 선택해 목록을 조회할 수 있습니다.



- 선택한 컬럼 목록은 메뉴 재진입 시 유지됩니다.
- **초기화** 버튼 선택 시 사전 정의된 기본 주요 지표 컬럼이 선택됩니다. 이 경우 기존 컬럼 선택 내역은 삭제됩니다.
- 컬럼은 최대 20개를 선택할 수 있습니다.
- 컬럼 선택 후 우측 하단 **적용** 버튼을 클릭 해 선택 내역을 적용할 수 있습니다.

상세 보기

목록에서 조회를 원하는 Deployment의 🔍 **상세 보기** 아이콘 선택 시 해당 Deployment에 대한 상세 정보를 확인할 수 있습니다.

ⓘ 상세 보기 창의 [트레이스](#), [메트릭스](#), [로그](#), [이벤트](#) 탭의 경우 컨테이너 맵 상세 보기와 동일합니다.

리소스

대상 선택 후 상세 보기의 [리소스](#) 탭에서 선택한 리소스와 관련된 리소스 목록 및 관계도를 조회할 수 있습니다.

× apm-java-demo

리소스 메트릭스 로그 이벤트

🕒 < 2024/03/15 15:44 >

컨테이너 → 컨테이너 목록으로 이동

ContainerId	ContainerName	Image	ConStatus	Pod	CpuByLimit	MemByLimit
30113e80d5c09...	apm-java-demo	592247757306.dkr.ecr...	Up 22 hours 48 minutes	apm-java-demo-7b69b...	0%	3.53%

Pod → Pod 목록으로 이동

Pod	containerIdsCount	ReplicaSet	Deployment	Namespace	CpuByLimit	MemByLimit
apm-java-demo-...	1	apm-java-demo-7b69b...	apm-java-demo	multitx-sample	0%	3.53%

노드 → 노드 목록으로 이동

OnodeName	Age	Image	CapacityCpu	CapacityMemory	Cpu	MemoryUsedPct
ip-10-21-13-168...	20d 22h 51m 28s	Amazon Linux 2	2	7.61 GiB	84%	77.67%

리소스 관계

```

    graph LR
      Namespace[multitx-sample] --> Deployment[apm-java-demo]
      Deployment --> ReplicaSet[apm-java-dem o-7b69b484cf]
      ReplicaSet --> Pod[apm-java-dem o-7b69b484cf...]
      Pod --> Container["(apm-java-de mo) 30113e80..."]
      subgraph Node [ip-10-21-13-168.ap-northeast-2.compute.internal]
        Container
      end
  
```

요약 정보

선택한 리소스와 관련된 리소스 목록의 요약 정보를 제공합니다. 리소스 명 옆에 상세보기 아이콘을 선택 시 해당 리소스 목록 화면으로 이동합니다.

- **컨테이너**: 해당 컨테이너만 필터링된 상태의 **컨테이너 목록**으로 이동합니다.
- **Pod**: 해당 파드만 필터링된 **Pod 목록**으로 이동합니다.
- **노드**: 해당 노드 이름이 선택된 상태의 **노드 상세**로 이동합니다.

리소스 관계도

리소스간 연결 관계와 상태를 시각적으로 확인할 수 있습니다. 목록과 동일하게 Pod, 컨테이너, 애플리케이션에 마우스 오버 시 상세 지표를 확인할 수 있습니다. 리소스 선택 시 해당 리소스 목록 페이지로 이동합니다.

노드 목록

홈 화면 > 프로젝트 선택 > 클러스터 > 노드 목록

ⓘ 특정 컬럼의 데이터가 표시되지 않는 것은 쿠버네티스 노드 에이전트가 구 버전이기 때문일 수 있습니다. 이런 경우 충분한 데이터가 수집될 수 있도록 쿠버네티스 노드 에이전트 버전을 1.3.7 이상으로 업데이트하세요.

현재 프로젝트의 노드의 정보를 목록 형태로 확인할 수 있습니다.

- 10초 주기로 실시간 데이터를 업데이트합니다.
- 노드 상태 요약, 비교, 필터 및 CSV 파일 다운로드 기능을 제공합니다.
 | CSV 파일명, `kubernetes_node_list_{pcode}_{timestamp}.csv`

기본 화면 안내

CP Kubernetes Monitoring Demo(EKS) ▾ 노드 목록

1 LIVE 12:32:26 전체 3 Ready 3

+ 적용된 필터가 없습니다. [컬럼 선택] [다운로드] → 이전 버전으로 보기

Status	OnodeName	Uptime ↑	노드 정보				CPU		메모리
			Age	Image	Kubelet	ContainerRuntime	Cpu	MemoryUsedPct	
○	ip-10-21-11-14...	12d 01h 04m 29s	12d 01h 03m 56s	Amazon Linux 2	v1.28.2-eks-a5df82a	containerd://1.6.19	37.03%	39.66%	
○	ip-10-21-11-14...	12d 01h 04m 32s	12d 01h 03m 57s	Amazon Linux 2	v1.28.2-eks-a5df82a	containerd://1.6.19	34.32%	43.25%	
○	ip-10-21-13-21...	17d 22h 23m 27s	17d 22h 22m 57s	Amazon Linux 2	v1.28.2-eks-a5df82a	containerd://1.6.19	79.78%	74.76%	

- 1 영역에서 원하는 옵션을 선택해 2 영역에서 데이터를 조회할 수 있습니다.
- 1 영역 좌측 상단에서 전체 노드 개수 및 상태별 노드 개수를 확인할 수 있습니다.
 - 1 영역 우측에서 → 이전 버전으로 보기 버튼 선택 시 개편 전 노드 목록 화면을 확인할 수 있습니다.
 - 2 영역에서 원하는 컬럼 선택 시 해당 컬럼 기준으로 노드 목록을 정렬해 조회할 수 있습니다.
 - 2 영역에서 OnodeName 컬럼 하위의 상세 아이콘 선택 시 해당 노드의 상세 페이지로 이동합니다.

ⓘ 노드 상세에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

메트릭 정보

- 원하는 컬럼 위에 마우스 커서를 위치시키면 다음과 같이 해당 컬럼이 제공하는 메트릭 정보를 확인할 수 있습니다.

	CPU	MemoryUsedPct	
		Field Unit pct	
ContainerRuntime	Cpu	MemoryUsedPct	IoPerce
ontainerd://1.6.19	16.88 %	18.88 %	...gent/1/
ontainerd://1.6.19	12.22 %	17.39 %	...gent/1/
ontainerd://1.6.19	16.35 %	17.65 %	...gent/1/

- 컬럼 선택 옵션 화면 내 컬럼 목록에서 컬럼별 메트릭 정보를 확인할 수 있습니다. 원하는 컬럼 위에 마우스 커서를 위치시키면 컬럼 이름 우측에 ⓘ 인포 아이콘이 나타납니다. ⓘ 인포 아이콘 위에 마우스 커서를 위치시키면 다음과 같이 메트릭 정보를 확인할 수 있습니다.

<input type="checkbox"/> OS	<input type="checkbox"/> CpuLoad15	<input type="checkbox"/> MemorySwapTot...	<input type="checkbox"/> UsedInodePercent
<input checked="" type="checkbox"/> Image	<input type="checkbox"/> CpuNice	<input type="checkbox"/> MemorySwapUs...	<input type="checkbox"/> IoPercent
<input type="checkbox"/> Kernel	<input type="checkbox"/> CpuSoftirq	<input type="checkbox"/> MemoryTotal	<input checked="" type="checkbox"/> IoPercentMax
<input type="checkbox"/> KubeProxy	<input type="checkbox"/> CpuSteal	<input checked="" type="checkbox"/> MemoryUsedPct	<input type="checkbox"/> QueueLength
<input checked="" type="checkbox"/> Kubelet	<input checked="" type="checkbox"/> CpuUsr		<input type="checkbox"/> ReadBps
<input checked="" type="checkbox"/> ContainerRuntime	<input checked="" type="checkbox"/> Field Unit pct		<input type="checkbox"/> ReadIops
	<input checked="" type="checkbox"/> CpuUsr ⓘ		<input type="checkbox"/> TotalSpace

필터

필터 옵션 활용 시 원하는 노드 목록을 선택해 조회할 수 있습니다. and 조건을 통해 여러 개의 필터를 적용할 수 있습니다.

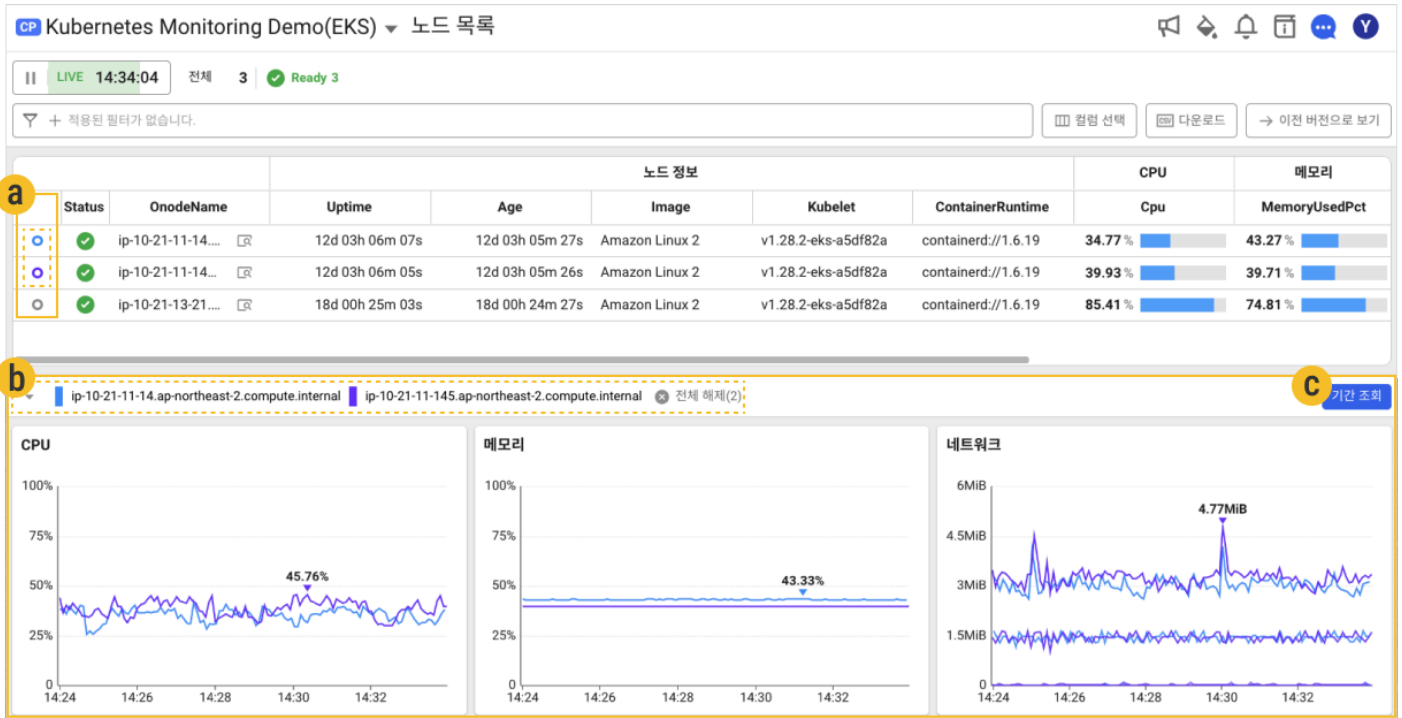
- ▽ 필터 입력 창에서 + 더하기 아이콘 선택 시 다음과 같이 필터 추가하기 창이 나타납니다.



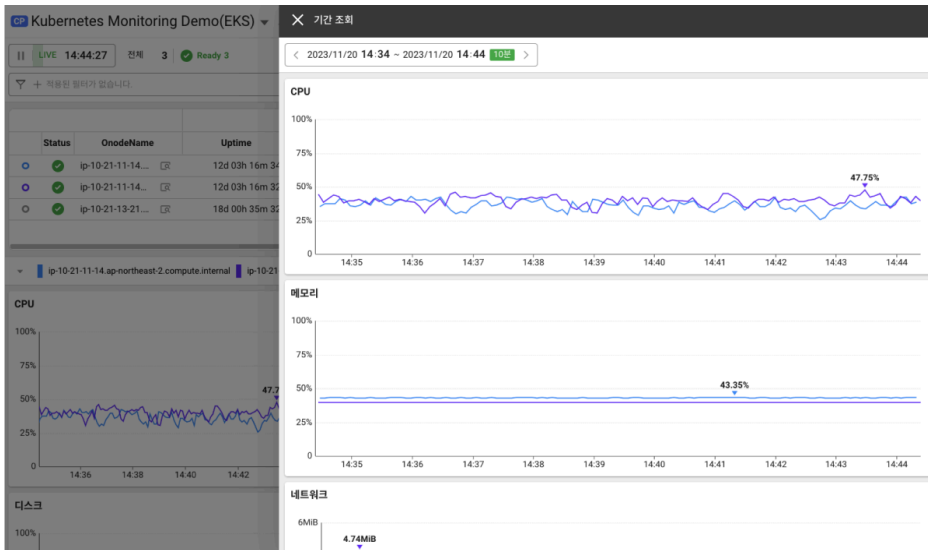
- **필터 키**와 **조건**을 선택하세요.
 - 조건에 맞는 값을 선택하세요.
2. 필터 설정 후 **적용** 버튼을 선택해 해당 필터를 적용하세요.
 3. 여러 개의 필터를 추가하려면 **필터** 입력 창 또는 **필터 추가하기** 창에서 **+ 더하기 아이콘**을 선택하세요.

비교하기

비교하기 옵션 활용 시 여러 노드의 주요 지표에 대한 비교 차트를 확인할 수 있습니다.

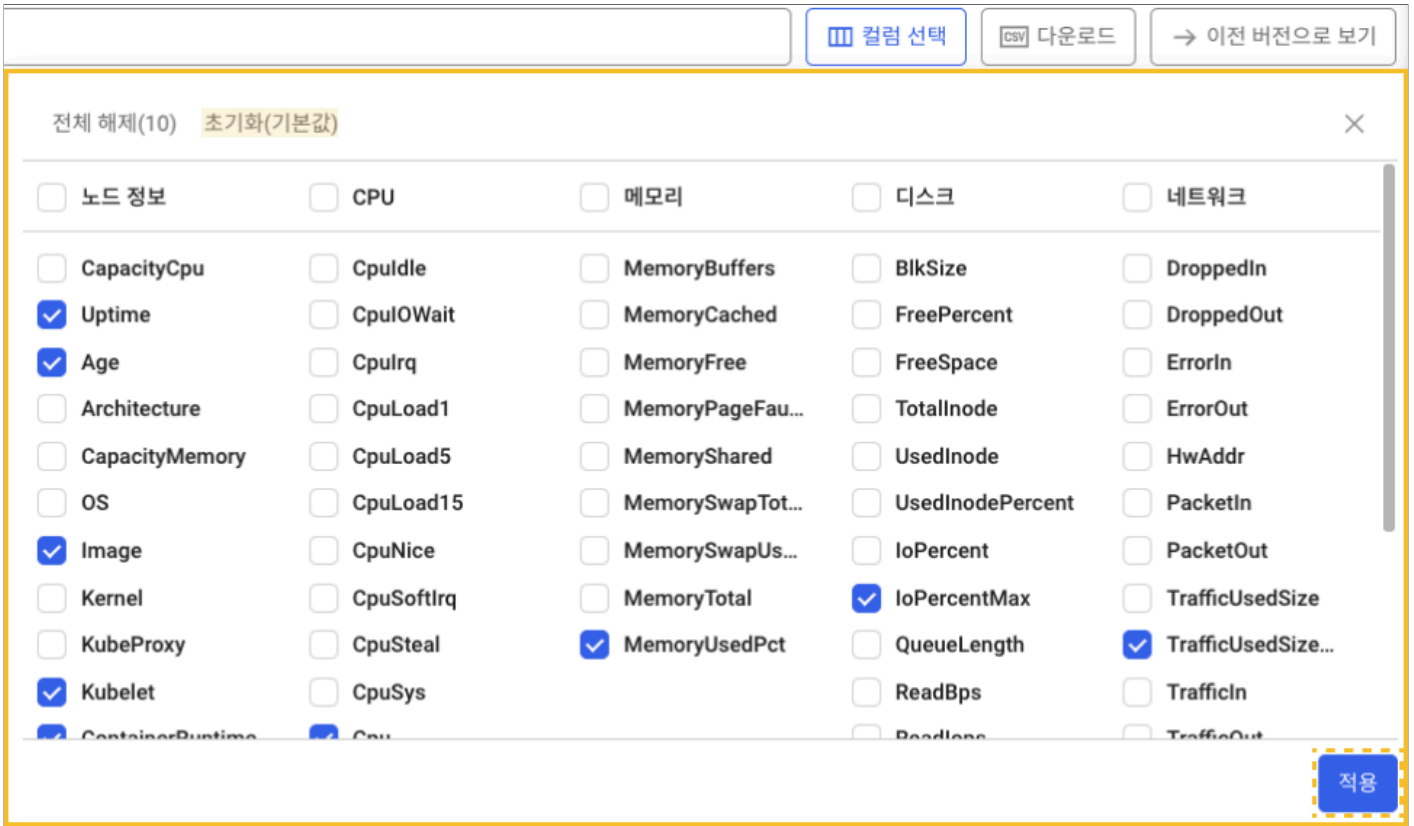


1. **a** **비교하기** 옵션 선택 시 노드 목록에 비교 컬럼이 추가됩니다.
2. 노드 목록 좌측 **b** 비교 컬럼을 통해 비교를 원하는 노드를 선택하세요.
 - 노드는 최대 10개까지 선택할 수 있습니다.
 - 선택한 노드별로 다른 색상이 부여됩니다.
3. **c** 영역에서 선택한 노드의 비교 차트를 확인할 수 있습니다.
 - **b** 비교 컬럼에서 선택한 노드를 **c** 영역 상단에서 확인할 수 있습니다.
 - **c** 상단을 드래그해 비교 차트 영역 높이를 조절할 수 있습니다.
4. **d** **기간 조회** 버튼 선택 시 다음과 같이 조회를 원하는 기간 내 비교 차트를 확인할 수 있습니다.



컬럼 선택

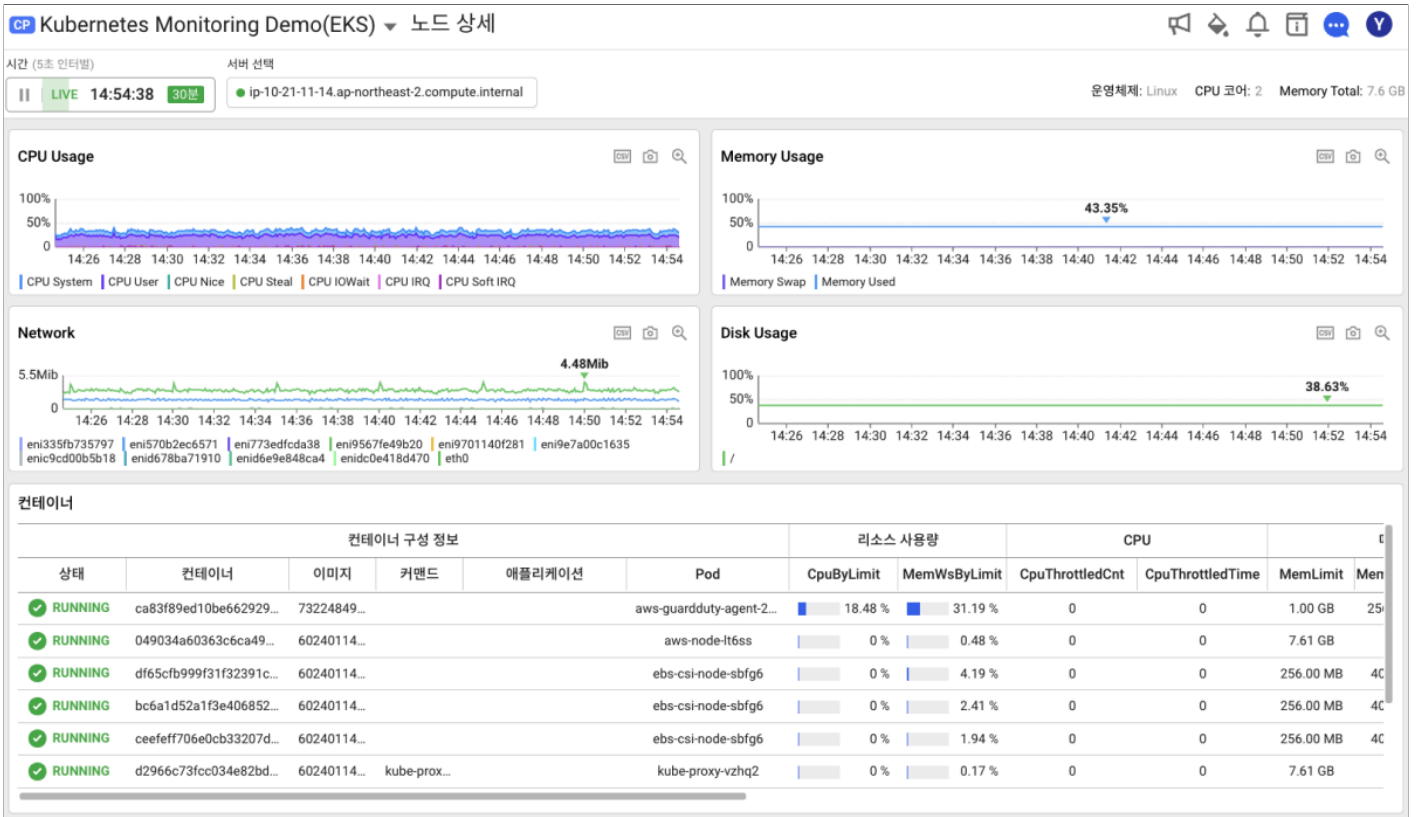
컬럼 선택 옵션 활용 시 원하는 노드 정보를 선택해 목록을 조회할 수 있습니다.



- 선택한 컬럼 목록은 메뉴 재진입 시 유지됩니다.
- 초기화 버튼 선택 시 사전 정의된 기본 주요 지표 컬럼이 선택됩니다. 이 경우 기존 컬럼 선택 내역은 삭제됩니다.
- 컬럼은 최대 20개를 선택할 수 있습니다.
- 컬럼 선택 후 우측 하단 적용 버튼을 클릭 해 선택 내역을 적용할 수 있습니다.

노드 상세

노드 상세 화면은 상단의 4개 차트와 노드 내 컨테이너 상세 정보로 구성되어 있습니다.



• 상단 차트

- **CPU Usage:** 노드의 CPU 사용량을 확인할 수 있습니다.
- **Memory Usage:** 노드의 메모리 사용량을 확인할 수 있습니다.
- **Network:** 노드의 네트워크 인터페이스의 입출력 트래픽과 오류 패킷을 확인할 수 있습니다.
- **Disk Usage:** 노드의 Disk 사용량을 확인할 수 있습니다.

• 컨테이너 정보

항목	설명
상태	컨테이너 상태
컨테이너	Container ID

항목	설명
이미지	컨테이너 이미지 정보
커맨드	컨테이너 실행 명령
애플리케이션	컨테이너에 설치된 와탭 APM 에이전트의 이름
Pod	Pod 이름
CpuByLimit	CPU Limit 기준 CPU 전체 사용률
MemoryWsByLimit	컨테이너 메모리 Limit 기준 working set 사용량
CpuThrottledCnt	컨테이너 CPU Throttled 건수
CpuThrottledTime	컨테이너 CPU Throttled 시간
MemLimit	컨테이너 메모리 Limit 크기
MemRequest	컨테이너 메모리 Request 크기
MemWs	컨테이너 메모리 working set
IoReadBytes	컨테이너 전체 블럭 디바이스의 초당 읽은 바이트 합
IoReadIops	컨테이너 전체 블럭 디바이스의 초당 읽은 건수 합
IoWriteBytes	컨테이너 전체 블럭 디바이스의 초당 쓴 바이트 합
IoWriteIops	컨테이너 전체 블럭 디바이스의 초당 쓴 건수 합
NetRxBytes	컨테이너 전체 블럭 디바이스의 초당 읽기 바이트 합
NetRxDropped	컨테이너 네트워크 수신 dropped 건수
NetRxError	컨테이너 네트워크 수신 에러 건수
NetRxIops	컨테이너 네트워크 수신 건수

항목	설명
NetTxBytes	컨테이너 네트워크 송신 데이터 크기
NetTxDropped	컨테이너 네트워크 송신 dropped 건수
NetTxError	컨테이너 네트워크 송신 에러 건수
NetTxlops	컨테이너 네트워크 송신 건수

오브젝트 매니페스트

홈 화면 > 프로젝트 선택 > 클러스터 > 오브젝트 매니페스트

ⓘ 요구 사항

와탭 마스터 에이전트 1.5.5 버전 이상이 필요합니다. 1.5.5 버전 미만일 경우 기존 [마스터 메타 정보](#) 화면이 제공됩니다.

쿠버네티스 마스터 에이전트는 클러스터에 있는 Pod, 서비스, ReplicaSet 등의 오브젝트에 대한 매니페스트를 하루에 한 번씩 자동으로 수집합니다. 해당 데이터는 날짜별로 저장되어 사용자가 특정 날짜 기준의 클러스터 상태를 쉽게 확인할 수 있습니다. 수집한 데이터는 날짜별 비교가 가능해 시간에 따른 클러스터의 변화를 파악하기 용이합니다.

또한 사용자는 쿠버네티스 마스터 에이전트의 Deployment 배포 전 컨테이너 환경변수를 설정해 수집 시간의 타임존을 자유롭게 지정할 수 있습니다. 해당 설정 방법은 다음 [컨테이너 환경변수](#) 안내를 참조하세요.

컨테이너 환경변수

• store_k8s_object_manifest_enabled

기본값

오브젝트 매니페스트 수집 기능을 활성화합니다. 기본값이 이기 때문에 별도 설정 없이 수집 기능이 활성화됩니다.

• store_k8s_object_manifest_timezone

기본값

오브젝트 매니페스트 수집을 위한 타임존을 설정합니다. 별도 설정이 없을 경우 기본값 기준 자정에 수집을 시작합니다.

- 지역 기반 타임존 또는 GMT 오프셋 형식으로 설정할 수 있습니다.
- 분 단위의 정밀한 시간 지정을 지원합니다.

예, 과 같이 설정해 특정 지역의 비표준 시간대 지정 가능

• store_k8s_object_manifest_max_fetch_limit

기본값

클러스터에서 한 번에 요청할 오브젝트 개수를 설정합니다. 별도 설정이 없을 경우 기본값 기준으로 1회 요청당 50개씩 요청합니다.

예, 전체 Pod 개수가 90개인 클러스터에서 해당 옵션 설정 시 Pod 정보를 30개씩 3회 요청

Example

```

containers:
- name: whatap-master-agent
  image: whatap/kube_mon
  resources:
    requests:
      memory: 300Mi
      cpu: 100m
    limits:
      memory: 350Mi
      cpu: 200m
  ports:
    - containerPort: 6600
  command: ["/bin/entrypoint.sh"]
  env:
    - name: WHATAP_LICENSE
      value: {PROJECT_LICENSE_VALUE}
    - name: WHATAP_HOST
      value: "13.124.11.223/13.209.172.35"
    - name: WHATAP_PORT
      value: "6600"
    - name: WHATP_MEM_LIMIT
      valueFrom:
        resourceFieldRef:
          containerName: whatap-master-agent
          resource: limits.memory
    # 추가된 환경 변수
    - name: store_k8s_object_manifest_enabled
      value: "true"
    - name: store_k8s_object_manifest_timezone
      value: "GMT+9"
    - name: store_k8s_object_manifest_max_fetch_limit
      value: "50"

```

기본 화면 안내

오브젝트 목록

오브젝트 종류 목록에서 원하는 오브젝트 선택 시 해당 오브젝트의 목록을 확인할 수 있습니다. [Name](#), [Namespace](#), [Uid](#), [CreatedTime](#) 등의 오브젝트 정보가 표시됩니다. 다음 화면은 오브젝트 종류 목록에서 [Deployment](#) 선택 시 예시 화면입니다.

CP Kubernetes Monitoring Demo(EKS) ▾ 오브젝트 매니페스트

현재 시각 날짜 마스터 에이전트

2024/01/23 Name KMaster Containers 78 Hosts 3 Images 22 Services 10 Active true IP Version 1.5.0 20240116 Memory 350MiB Start Time 2024/01/18 오후 2:26:40

	Name	Namespace	Uid	CreatedTime
Workload				
Cronjob				
DaemonSet				
Deployment	virtual-pending-be	default	74f6037a-350c-48a9-8b5c-b156354ea5ab	2023-11-20T07:07:57Z
	whatap-virtual-deployment	default	00fd002c-4a8d-4815-8fef-dc9f89d50ecc	2023-11-20T06:58:58Z
Job	coredns	kube-system	79a69bee-1b62-4fa8-85c3-c4036c7f8d43	2023-09-13T05:28:09Z
Pod	ebs-csi-controller	kube-system	9877183f-a1f9-41b5-abf1-0e8a718ed45d	2023-10-30T06:09:38Z
ReplicaSet	apm-java-demo	multitx-sample	e86ead35-d92a-4300-9bce-0390b650cdab	2023-10-27T07:07:33Z
StatefulSet	nodejs-sample-v2	multitx-sample	153484cd-95f5-4904-9912-8100e67976b3	2023-11-10T07:18:58Z
Network	php-demo-apache-deployment	multitx-sample	b64ffcd-095d-41c5-81b0-80bcd2c95bbe	2023-11-10T00:50:07Z
Ingress	php-demo-wordpress-apache-deploy...	multitx-sample	a563989e-9af8-4bdb-9308-ad490ec9de21	2023-11-01T00:12:56Z
Service	testing-http-sender	multitx-sample	4fa6b5bd-bf26-4ba9-b709-6fa0be44f2a3	2023-10-31T08:10:03Z
Config/Storage	whatap-virtual-deployment	virtual-main	09b8c928-dca0-43b6-bdb0-07209c5773ee	2023-09-14T00:21:58Z
ConfigMap	whatap-virtual-deployment	virtual	b2b003cf-0394-483a-80ed-af03a032a1db	2023-09-18T08:22:15Z
PersistentVolume	whatap-master-agent	whatap-monitoring	4e027d67-06f0-4f01-9f91-33d376e3e42d	2023-09-14T00:20:32Z
PersistentVolumeClaim				
StorageClass				
Security				
Role				
ServiceAccount				
Cluster				
ClusterRoleBinding				
ClusterRole				
Node				

오브젝트 매니페스트 상세

오브젝트 목록에서 [상세 보기](#) 버튼 선택 시 [오브젝트 매니페스트 상세](#) 창에서 선택한 오브젝트의 매니페스트를 조회할 수 있습니다.

오브젝트 매니페스트 상세

Kind Deployment Name virtual-pending-be Namespace default uid 74f6037a-350c-48a9-8b5c-b156354ea5ab CreatedTime 2023-11-20T07:07:57Z

2024/01/17 ~ 2024/01/23 7일

비교하기

현재 시각

2024/01/23 09:00

2024/01/22 09:00

2024/01/21 09:00

2024/01/20 09:00

2024/01/19 09:00

2024/01/23 09:00

```

1  {
2  "metadata": {
3  "generation": "1",
4  "uid": "74f6037a-350c-48a9-8b5c-b156354ea5ab",
5  "managedFields": [
6  {
7  "apiVersion": "apps/v1",
8  "manager": "kubectl-client-side-apply",
9  "fieldsV1": {
10 "f:metadata": {
11 "f:annotations": {
12 "f:kubectl.kubernetes.io/last-applied-configuration": {},
13 "": {}
14 }
15 },
16 "f:spec": {
17 "f:template": {
18 "f:metadata": {
19 "f:labels": {
20 "f:app": {},
21 "": {}
22 }
23 "f:name": {}
24 },
25 "f:spec": {
26 "f:securityContext": {},
27 "f:restartPolicy": {},

```

오브젝트 매니페스트 저장 시간 목록

- 해당 오브젝트의 매니페스트가 저장된 시간 목록을 좌측에서 확인할 수 있습니다. 오브젝트 매니페스트는 매일 정해진 시간에 자동으로 저장됩니다.
- 오브젝트 매니페스트 저장 시간은 마스터 에이전트의 컨테이너 환경변수 `store_k8s_object_manifest_timezone` 값을 수정해 변경할 수 있습니다.
- 오브젝트 매니페스트는 매일 정해진 저장 시간에 저장될 뿐 아니라 경고 이벤트가 발생한 경우에도 추가로 저장됩니다.

ⓘ 오브젝트 매니페스트가 저장되는 경고 이벤트는 Warning 타입의 쿠버네티스 이벤트 오브젝트를 의미합니다.

오브젝트 매니페스트 비교

비교하기 토크 활성화 시 다음과 같이 저장된 두 개의 오브젝트 매니페스트를 비교할 수 있습니다. 변경 사항이 있을 경우 하이라이팅됩니다.

Kind	Node	Name	gke-jinha-test-cluster-default-pool-a5e915f5-kb5t	uid	e762d220-2af3-4386-b222-d89dc07196a5	Creation Timestamp	2023-12-07T03:04:43Z																																																																																																																																																
<div style="display: flex; justify-content: space-between; align-items: center;"> <div style="border: 1px solid #ccc; padding: 2px;"> 2024/01/11 ~ 2024/01/17 7일 </div> <div style="display: flex; align-items: center;"> <input checked="" type="checkbox"/> 비교하기 </div> </div>																																																																																																																																																							
<div style="display: flex;"> <div style="width: 20%; border-right: 1px solid #ccc; padding-right: 5px;"> <p>현재 시각</p> <p>2024/01/17 09:02</p> <p>2024/01/16 00:01</p> <p>2024/01/15 00:01</p> <p>2024/01/14 00:01</p> <p>2024/01/13 00:01</p> <p>2024/01/12 15:04</p> <p>2024/01/11 14:18</p> <p>2024/01/11 13:58</p> <p>2024/01/11 13:38</p> <p>2024/01/11 13:18</p> <p>2024/01/11 12:58</p> <p>2024/01/11 12:38</p> <p>2024/01/11 12:18</p> <p>2024/01/11 11:58</p> <p>2024/01/11 11:38</p> <p>2024/01/11 11:18</p> <p>2024/01/11 10:58</p> <p>2024/01/11 10:38</p> <p>2024/01/11 10:18</p> <p>2024/01/11 09:58</p> <p>2024/01/11 09:38</p> <p>2024/01/11 09:18</p> <p>2024/01/11 08:58</p> <p>2024/01/11 08:38</p> <p>2024/01/11 08:18</p> <p>2024/01/11 07:58</p> <p>2024/01/11 07:38</p> <p>2024/01/11 07:18</p> <p style="background-color: #e0e0ff;">2024/01/11 07:05 Warning</p> <p>2024/01/11 06:58</p> <p>2024/01/11 06:38</p> <p>2024/01/11 06:21 Warning</p> <p>2024/01/11 06:18</p> <p>2024/01/11 06:04 Warning</p> <p>2024/01/11 05:58</p> </div> <div style="width: 80%; padding-left: 5px;"> <table border="0" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 5%; text-align: right;">489</td> <td style="width: 45%;">"machineID": "b866564eaac1c400d3a6489ebf9ec462",</td> <td style="width: 5%; text-align: right;">492</td> <td style="width: 45%;">"machineID": "b866564eaac1c400d3a6489ebf9ec462",</td> </tr> <tr> <td style="text-align: right;">490</td> <td>"bootID": "ce093e0e-751a-46e2-b54b-01bdf713e1fa",</td> <td style="text-align: right;">493</td> <td>"bootID": "ce093e0e-751a-46e2-b54b-01bdf713e1fa",</td> </tr> <tr> <td style="text-align: right;">491</td> <td>"kernelVersion": "5.15.109+",</td> <td style="text-align: right;">494</td> <td>"kernelVersion": "5.15.109+",</td> </tr> <tr> <td style="text-align: right;">492</td> <td>"containerRuntimeVersion": "containerd://1.7.0",</td> <td style="text-align: right;">495</td> <td>"containerRuntimeVersion": "containerd://1.7.0",</td> </tr> <tr> <td style="text-align: right;">493</td> <td>"kubeletVersion": "v1.27.3-gke.100",</td> <td style="text-align: right;">496</td> <td>"kubeletVersion": "v1.27.3-gke.100",</td> </tr> <tr> <td style="text-align: right;">494</td> <td>"systemUUID": "b866564e-aac1-c400-d3a6-489ebf9ec462",</td> <td style="text-align: right;">497</td> <td>"systemUUID": "b866564e-aac1-c400-d3a6-489ebf9ec462",</td> </tr> <tr> <td style="text-align: right;">495</td> <td>"kubeProxyVersion": "v1.27.3-gke.100",</td> <td style="text-align: right;">498</td> <td>"kubeProxyVersion": "v1.27.3-gke.100",</td> </tr> <tr> <td style="text-align: right;">496</td> <td>"operatingSystem": "linux",</td> <td style="text-align: right;">499</td> <td>"operatingSystem": "linux",</td> </tr> <tr> <td style="text-align: right;">497</td> <td>"osImage": "Container-Optimized OS from Google",</td> <td style="text-align: right;">500</td> <td>"osImage": "Container-Optimized OS from Google",</td> </tr> <tr> <td style="text-align: right;">498</td> <td>"architecture": "amd64"</td> <td style="text-align: right;">501</td> <td>"architecture": "amd64"</td> </tr> <tr> <td style="text-align: right;">499</td> <td>),</td> <td style="text-align: right;">502</td> <td>),</td> </tr> <tr> <td style="text-align: right;">500</td> <td>"conditions": [</td> <td style="text-align: right;">503</td> <td>"conditions": [</td> </tr> <tr> <td style="text-align: right;">501</td> <td>{</td> <td style="text-align: right;">504</td> <td>{</td> </tr> <tr> <td style="text-align: right;">502</td> <td>- "reason": "NoCorruptDockerOverlay2",</td> <td style="text-align: right;">505</td> <td>+ "reason": "NoFrequentKubeletRestart",</td> </tr> <tr> <td style="text-align: right;">503</td> <td>- "lastHeartbeatTime": "2024-01-16T23:59:37Z",</td> <td style="text-align: right;">506</td> <td>+ "lastHeartbeatTime": "2024-01-10T22:03:17Z",</td> </tr> <tr> <td style="text-align: right;">504</td> <td>- "type": "CorruptDockerOverlay2",</td> <td style="text-align: right;">507</td> <td>+ "type": "FrequentKubeletRestart",</td> </tr> <tr> <td style="text-align: right;">505</td> <td>"lastTransitionTime": "2023-12-07T03:04:44Z",</td> <td style="text-align: right;">508</td> <td>"lastTransitionTime": "2023-12-07T03:04:44Z",</td> </tr> <tr> <td style="text-align: right;">506</td> <td>- "message": "docker overlay2 is functioning properly",</td> <td style="text-align: right;">509</td> <td>+ "message": "kubelet is functioning properly",</td> </tr> <tr> <td style="text-align: right;">507</td> <td>"status": "False"</td> <td style="text-align: right;">510</td> <td>"status": "False"</td> </tr> <tr> <td style="text-align: right;">508</td> <td>},</td> <td style="text-align: right;">511</td> <td>},</td> </tr> <tr> <td style="text-align: right;">509</td> <td>{</td> <td style="text-align: right;">512</td> <td>{</td> </tr> <tr> <td style="text-align: right;">510</td> <td>- "reason": "NoFrequentUnregisterNetDevice",</td> <td style="text-align: right;">513</td> <td>+ "reason": "NoFrequentDockerRestart",</td> </tr> <tr> <td style="text-align: right;">511</td> <td>- "lastHeartbeatTime": "2024-01-16T23:59:37Z",</td> <td style="text-align: right;">514</td> <td>+ "lastHeartbeatTime": "2024-01-10T22:03:17Z",</td> </tr> <tr> <td style="text-align: right;">512</td> <td>- "type": "FrequentUnregisterNetDevice",</td> <td style="text-align: right;">515</td> <td>+ "type": "FrequentDockerRestart",</td> </tr> <tr> <td style="text-align: right;">513</td> <td>- "lastTransitionTime": "2023-12-07T03:04:45Z",</td> <td style="text-align: right;">516</td> <td>+ "lastTransitionTime": "2023-12-07T03:04:44Z",</td> </tr> <tr> <td style="text-align: right;">514</td> <td>- "message": "node is functioning properly",</td> <td style="text-align: right;">517</td> <td>+ "message": "docker is functioning properly",</td> </tr> <tr> <td style="text-align: right;">515</td> <td>"status": "False"</td> <td style="text-align: right;">518</td> <td>"status": "False"</td> </tr> <tr> <td style="text-align: right;">516</td> <td>},</td> <td style="text-align: right;">519</td> <td>},</td> </tr> <tr> <td style="text-align: right;">517</td> <td>{</td> <td style="text-align: right;">520</td> <td>{</td> </tr> <tr> <td style="text-align: right;">518</td> <td>- "reason": "NoFrequentKubeletRestart",</td> <td style="text-align: right;">521</td> <td>+ "reason": "NoFrequentContainerdRestart",</td> </tr> <tr> <td style="text-align: right;">519</td> <td>- "lastHeartbeatTime": "2024-01-16T23:59:37Z",</td> <td style="text-align: right;">522</td> <td>+ "lastHeartbeatTime": "2024-01-10T22:03:17Z",</td> </tr> <tr> <td style="text-align: right;">520</td> <td>- "type": "FrequentKubeletRestart",</td> <td style="text-align: right;">523</td> <td>+ "type": "FrequentContainerdRestart",</td> </tr> <tr> <td style="text-align: right;">521</td> <td>"lastTransitionTime": "2023-12-07T03:04:44Z",</td> <td style="text-align: right;">524</td> <td>"lastTransitionTime": "2023-12-07T03:04:44Z",</td> </tr> <tr> <td style="text-align: right;">522</td> <td>- "message": "kubelet is functioning properly",</td> <td style="text-align: right;">525</td> <td>+ "message": "containerd is functioning properly",</td> </tr> <tr> <td style="text-align: right;">523</td> <td>"status": "False"</td> <td style="text-align: right;">526</td> <td>"status": "False"</td> </tr> <tr> <td style="text-align: right;">524</td> <td>},</td> <td style="text-align: right;">527</td> <td>},</td> </tr> </table> </div> </div>								489	"machineID": "b866564eaac1c400d3a6489ebf9ec462",	492	"machineID": "b866564eaac1c400d3a6489ebf9ec462",	490	"bootID": "ce093e0e-751a-46e2-b54b-01bdf713e1fa",	493	"bootID": "ce093e0e-751a-46e2-b54b-01bdf713e1fa",	491	"kernelVersion": "5.15.109+",	494	"kernelVersion": "5.15.109+",	492	"containerRuntimeVersion": "containerd://1.7.0",	495	"containerRuntimeVersion": "containerd://1.7.0",	493	"kubeletVersion": "v1.27.3-gke.100",	496	"kubeletVersion": "v1.27.3-gke.100",	494	"systemUUID": "b866564e-aac1-c400-d3a6-489ebf9ec462",	497	"systemUUID": "b866564e-aac1-c400-d3a6-489ebf9ec462",	495	"kubeProxyVersion": "v1.27.3-gke.100",	498	"kubeProxyVersion": "v1.27.3-gke.100",	496	"operatingSystem": "linux",	499	"operatingSystem": "linux",	497	"osImage": "Container-Optimized OS from Google",	500	"osImage": "Container-Optimized OS from Google",	498	"architecture": "amd64"	501	"architecture": "amd64"	499),	502),	500	"conditions": [503	"conditions": [501	{	504	{	502	- "reason": "NoCorruptDockerOverlay2",	505	+ "reason": "NoFrequentKubeletRestart",	503	- "lastHeartbeatTime": "2024-01-16T23:59:37Z",	506	+ "lastHeartbeatTime": "2024-01-10T22:03:17Z",	504	- "type": "CorruptDockerOverlay2",	507	+ "type": "FrequentKubeletRestart",	505	"lastTransitionTime": "2023-12-07T03:04:44Z",	508	"lastTransitionTime": "2023-12-07T03:04:44Z",	506	- "message": "docker overlay2 is functioning properly",	509	+ "message": "kubelet is functioning properly",	507	"status": "False"	510	"status": "False"	508	},	511	},	509	{	512	{	510	- "reason": "NoFrequentUnregisterNetDevice",	513	+ "reason": "NoFrequentDockerRestart",	511	- "lastHeartbeatTime": "2024-01-16T23:59:37Z",	514	+ "lastHeartbeatTime": "2024-01-10T22:03:17Z",	512	- "type": "FrequentUnregisterNetDevice",	515	+ "type": "FrequentDockerRestart",	513	- "lastTransitionTime": "2023-12-07T03:04:45Z",	516	+ "lastTransitionTime": "2023-12-07T03:04:44Z",	514	- "message": "node is functioning properly",	517	+ "message": "docker is functioning properly",	515	"status": "False"	518	"status": "False"	516	},	519	},	517	{	520	{	518	- "reason": "NoFrequentKubeletRestart",	521	+ "reason": "NoFrequentContainerdRestart",	519	- "lastHeartbeatTime": "2024-01-16T23:59:37Z",	522	+ "lastHeartbeatTime": "2024-01-10T22:03:17Z",	520	- "type": "FrequentKubeletRestart",	523	+ "type": "FrequentContainerdRestart",	521	"lastTransitionTime": "2023-12-07T03:04:44Z",	524	"lastTransitionTime": "2023-12-07T03:04:44Z",	522	- "message": "kubelet is functioning properly",	525	+ "message": "containerd is functioning properly",	523	"status": "False"	526	"status": "False"	524	},	527	},
489	"machineID": "b866564eaac1c400d3a6489ebf9ec462",	492	"machineID": "b866564eaac1c400d3a6489ebf9ec462",																																																																																																																																																				
490	"bootID": "ce093e0e-751a-46e2-b54b-01bdf713e1fa",	493	"bootID": "ce093e0e-751a-46e2-b54b-01bdf713e1fa",																																																																																																																																																				
491	"kernelVersion": "5.15.109+",	494	"kernelVersion": "5.15.109+",																																																																																																																																																				
492	"containerRuntimeVersion": "containerd://1.7.0",	495	"containerRuntimeVersion": "containerd://1.7.0",																																																																																																																																																				
493	"kubeletVersion": "v1.27.3-gke.100",	496	"kubeletVersion": "v1.27.3-gke.100",																																																																																																																																																				
494	"systemUUID": "b866564e-aac1-c400-d3a6-489ebf9ec462",	497	"systemUUID": "b866564e-aac1-c400-d3a6-489ebf9ec462",																																																																																																																																																				
495	"kubeProxyVersion": "v1.27.3-gke.100",	498	"kubeProxyVersion": "v1.27.3-gke.100",																																																																																																																																																				
496	"operatingSystem": "linux",	499	"operatingSystem": "linux",																																																																																																																																																				
497	"osImage": "Container-Optimized OS from Google",	500	"osImage": "Container-Optimized OS from Google",																																																																																																																																																				
498	"architecture": "amd64"	501	"architecture": "amd64"																																																																																																																																																				
499),	502),																																																																																																																																																				
500	"conditions": [503	"conditions": [
501	{	504	{																																																																																																																																																				
502	- "reason": "NoCorruptDockerOverlay2",	505	+ "reason": "NoFrequentKubeletRestart",																																																																																																																																																				
503	- "lastHeartbeatTime": "2024-01-16T23:59:37Z",	506	+ "lastHeartbeatTime": "2024-01-10T22:03:17Z",																																																																																																																																																				
504	- "type": "CorruptDockerOverlay2",	507	+ "type": "FrequentKubeletRestart",																																																																																																																																																				
505	"lastTransitionTime": "2023-12-07T03:04:44Z",	508	"lastTransitionTime": "2023-12-07T03:04:44Z",																																																																																																																																																				
506	- "message": "docker overlay2 is functioning properly",	509	+ "message": "kubelet is functioning properly",																																																																																																																																																				
507	"status": "False"	510	"status": "False"																																																																																																																																																				
508	},	511	},																																																																																																																																																				
509	{	512	{																																																																																																																																																				
510	- "reason": "NoFrequentUnregisterNetDevice",	513	+ "reason": "NoFrequentDockerRestart",																																																																																																																																																				
511	- "lastHeartbeatTime": "2024-01-16T23:59:37Z",	514	+ "lastHeartbeatTime": "2024-01-10T22:03:17Z",																																																																																																																																																				
512	- "type": "FrequentUnregisterNetDevice",	515	+ "type": "FrequentDockerRestart",																																																																																																																																																				
513	- "lastTransitionTime": "2023-12-07T03:04:45Z",	516	+ "lastTransitionTime": "2023-12-07T03:04:44Z",																																																																																																																																																				
514	- "message": "node is functioning properly",	517	+ "message": "docker is functioning properly",																																																																																																																																																				
515	"status": "False"	518	"status": "False"																																																																																																																																																				
516	},	519	},																																																																																																																																																				
517	{	520	{																																																																																																																																																				
518	- "reason": "NoFrequentKubeletRestart",	521	+ "reason": "NoFrequentContainerdRestart",																																																																																																																																																				
519	- "lastHeartbeatTime": "2024-01-16T23:59:37Z",	522	+ "lastHeartbeatTime": "2024-01-10T22:03:17Z",																																																																																																																																																				
520	- "type": "FrequentKubeletRestart",	523	+ "type": "FrequentContainerdRestart",																																																																																																																																																				
521	"lastTransitionTime": "2023-12-07T03:04:44Z",	524	"lastTransitionTime": "2023-12-07T03:04:44Z",																																																																																																																																																				
522	- "message": "kubelet is functioning properly",	525	+ "message": "containerd is functioning properly",																																																																																																																																																				
523	"status": "False"	526	"status": "False"																																																																																																																																																				
524	},	527	},																																																																																																																																																				

① 컨테이너 맵에서 오브젝트 매니페스트 조회

컨테이너 맵에서 [오브젝트 매니페스트](#)를 확인할 수 있습니다.

The screenshot shows the Kubernetes Monitoring Demo(EKS) interface. The main view is a 'Pod' map with 39 pods in the 'Running' state. A specific pod, 'always-fail-pod', is selected and its details are shown in a side panel. The side panel includes tabs for '대상 정보' (Target Info), '메트릭스' (Metrics), '로그' (Logs), and '이벤트' (Events). The '대상 정보' tab is active, showing the following details:

- Node: ip-10-21-11-232.ap-northeast-2.compute.internal
- Namespace: lifecycle-test
- Pod: always-fail-pod
- ContainerId: 61262ddc895d2ec64022259d22e4ed9e43300...

Below the map, there is a table with pod metrics:

Pod	ActiveTX									
	CpuTotUsage	CpuByNode	MemUsage	NetRxBytes	NetTxBytes	Normal	Slow	Very Slow	Total	TPS
	1.69cores	84.27%	5.2GiB	16.9MiB	23.9MiB	227	83	53	363	180.75

- 원하는 블록을 선택하세요.
- 요약보기 창에서 **대상 정보** 탭을 선택해 Pod 또는 컨테이너 정보를 확인하세요.
- Pod, Deployment, ReplicaSet, Node 태그의 오른쪽 **더보기** 아이콘을 클릭해 [오브젝트 매니페스트](#)를 선택하세요.

kube-apiserver 대시보드

홈 화면 > 클러스터 프로젝트 선택 > 클러스터 > kube-apiserver 대시보드

ⓘ 요구 사항

- 쿠버네티스 클러스터 프로젝트(CP) 읽기 권한이 필요합니다.
- 와탭 쿠버네티스 에이전트 1.5.6 버전 이상이 필요합니다.

kube-apiserver는 **컨트롤 플레인(Control Plane)** 내/외부 요청을 처리하는 컴포넌트입니다. [kube-apiserver 대시보드](#) 메뉴를 통해 kube-apiserver의 상태와 성능 메트릭을 수집 후 kube-apiserver의 응답 시간, 처리량, 요청 실패 등을 추적해 클러스터의 가용성과 성능을 쉽게 모니터링할 수 있습니다.

기본 화면 안내



[kube-apiserver 대시보드](#) 메뉴는 최근 1시간 동안의 kube-apiserver 주요 지표를 1분 주기로 반영합니다. 다음과 같은 지표를 확인할 수 있습니다. 지표 상세는 하단 [메트릭 안내](#)를 참조하세요.

- 클러스터 부하 및 사용량 모니터링을 위해 메소드, 리소스, 응답 코드별 요청 횟수 현황 및 추이를 제공합니다.
- 클러스터 부하 상태를 파악할 수 있도록 현재 처리 중인 API 요청 수를 제공합니다.
- 생성된 OS 스레드와 현재 존재하는 고루틴의 수를 제공합니다.

kube-apiserver 모니터링 활성화

와탭 쿠버네티스 에이전트 설치 시 다운로드한 [yaml](#) 파일을 다음 안내에 따라 수정해 kube-apiserver 모니터링을 활성화하세요.

1. `whatap-master-agent` deployment의 spec 하위에 kube-apiserver 모니터링을 위한 에이전트 컨테이너 spec을 추가하세요.

```
containers:
- name: whatap-control-plane-helper
  image: whatap/kube_mon
  imagePullPolicy: Always
  ports:
  - containerPort: 9496
  command: ["/data/agent/master/whatap_control_plane_helper"]
  args: ["-collect_control_plane_monitoring_enabled=true"]
```

다음 예시를 참조하세요.

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: whatap-master-agent
  namespace: whatap-monitoring
spec:
  replicas: 1
  selector:
    matchLabels:
      name: whatap-master-agent
  template:
    metadata:
      labels:
        name: whatap-master-agent
    spec:
      containers:
      - name: whatap-control-plane-helper
        image: whatap/kube_mon
        imagePullPolicy: Always
        ports:
        - containerPort: 9496
        command: ["/data/agent/master/whatap_control_plane_helper"]
        args: ["-collect_control_plane_monitoring_enabled=true"]
      - name: whatap-master-agent
        image: whatap/kube_mon
        imagePullPolicy: Always
        resources:
          requests:
```

```

    memory: 300Mi
    cpu: 100m
  limits:
    memory: 350Mi
    cpu: 200m
  ports:
  - containerPort: 6600
  command: ["/bin/entrypoint.sh"]
  env:
  - name: WHATAP_LICENSE
    value: "xxxxxxxx-12345xxxxx-x00e000000xxx8"
  - name: WHATAP_HOST
    value: "13.124.11.223/13.209.172.35"
  - name: WHATAP_PORT
    value: "6600"
  - name: WHATP_MEM_LIMIT
    valueFrom:
      resourceFieldRef:
        containerName: whatap-master-agent
        resource: limits.memory
  volumeMounts:
  - name: start-script-volume
    mountPath: /bin/entrypoint.sh
    readOnly: true
    subPath: entrypoint.sh
  - mountPath: /whatap_conf
    name: whatap-config-volume
  volumes:
  - name: start-script-volume
    configMap:
      defaultMode: 0700
      name: master-start-script
  - name: whatap-config-volume
    emptyDir: {}
  serviceAccount: whatap

```

2. `whatap-master-agent` deployment의 `whatap-master-agent` 컨테이너 spec에서 env 설정에 다음 내용을 추가하세요.

```

env:
  - name: collect_control_plane_monitoring_enabled
    value: "true"

```

다음 예시를 참조하세요.

```

apiVersion: apps/v1
kind: Deployment
metadata:
  name: whatap-master-agent
  namespace: whatap-monitoring
spec:
  replicas: 1

```



```

selector:
  matchLabels:
    name: whatap-master-agent
template:
  metadata:
    labels:
      name: whatap-master-agent
  spec:
    containers:
      - name: whatap-control-plane-helper
        image: whatap/kube_mon
        imagePullPolicy: Always
        ports:
          - containerPort: 9496
        command: ["/data/agent/master/whatap_control_plane_helper"]
        args: ["-collect_control_plane_monitoring_enabled=true"]
      - name: whatap-master-agent
        image: whatap/kube_mon
        imagePullPolicy: Always
        resources:
          requests:
            memory: 300Mi
            cpu: 100m
          limits:
            memory: 350Mi
            cpu: 200m
        ports:
          - containerPort: 6600
        command: ["/bin/entrypoint.sh"]
        env:
          - name: collect_control_plane_monitoring_enabled
            value: "true"
          - name: WHATAP_LICENSE
            value: "xxxxxxxx-12345xxxxx-x00e000000xxx8"
          - name: WHATAP_HOST
            value: "13.124.11.223/13.209.172.35"
          - name: WHATAP_PORT
            value: "6600"
          - name: WHATP_MEM_LIMIT
            valueFrom:
              resourceFieldRef:
                containerName: whatap-master-agent
                resource: limits.memory
        volumeMounts:
          - name: start-script-volume
            mountPath: /bin/entrypoint.sh
            readOnly: true
            subPath: entrypoint.sh
          - mountPath: /whatap_conf
            name: whatap-config-volume
    volumes:
      - name: start-script-volume

```

```

configMap:
  defaultMode: 0700
  name: master-start-script
- name: whatap-config-volume
  emptyDir: {}
serviceAccount: whatap

```

3. 컨트롤 플레인 모니터링 성능 지표를 수집하기 위해 `whatap` ClusterRole에 다음의 권한을 추가하세요.

```

rules:
- nonResourceURLs: ["/metrics"]
  verbs: ["*"]

```

다음 예시를 참조하세요.

```

apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  annotations:
    rbac.authorization.kubernetes.io/autoupdate: "true"
  name: whatap
  namespace: whatap-monitoring
rules:
- apiGroups: ["*"]
  resources: ["namespaces", "pods", "pods/log", "configmaps", "services",
"endpoints", "daemonsets", "ingresses", "deployments", "nodes", "persistentvolumes", "persistentvolumeclaims", "events", "replicasets", "roles",
"rolebindings", "clusterroles", "clusterrolebindings", "jobs", "cronjobs", "statefulsets", "serviceaccounts", "configmaps", "storageclasses",
"horizontalpodautoscalers" ]
  verbs: ["get", "list", "watch"]
- apiGroups: [""]
  resources: ["pods/exec"]
  verbs: ["create"]
- apiGroups: [""]
  resources: ["configmaps"]
  verbs: ["*"]
- nonResourceURLs: ["/metrics"]
  verbs: ["*"]
---

```

ⓘ `kubectl apply` 또는 `kubectl edit` 명령을 통해 kube-apiserver 모니터링을 활성화할 수도 있습니다.

kube-apiserver 메트릭

분류	메트릭	설명
API GET 요청	분당 GET 요청 현황	kube-apiserver 의 GET 요청에 대한 분당 요청 수
	초당 GET 요청 수 추이	kube-apiserver 의 GET 요청에 대한 초당 요청 수 추이
	Resource & Subresource 별 분당 GET 요청 수 추이	kube-apiserver 에 들어온 GET 요청의 Resource 와 Subresource 별 분당 요청 수 추이
API POST 요청	분당 POST 요청 현황	kube-apiserver 의 POST 요청에 대한 분당 요청 수
	초당 POST 요청 수 추이	kube-apiserver 의 POST 요청에 대한 초당 요청 수 추이
	Resource & Subresource 별 POST 분당 요청 수 추이	kube-apiserver 에 들어온 POST 요청에 대해 Resource 와 Subresource 별로 그룹화 한 분당 요청 수 추이
API 응답 코드별 요청	[2xx] 분당 응답 요청 현황	kube-apiserver 요청 중 200 이상 300 미만의 응답 코드가 내려진 요청 수에 대한 분당 현황
	[2xx] 분당 응답 코드별 요청 수 추이	kube-apiserver 요청 중 200 이상 300 미만의 응답 코드가 내려진 요청 수에 대한 응답 코드별 분당 현황
	[2xx] 분당 Resource & Subresource 별 요청 수 추이	kube-apiserver 요청 중 200 이상 300 미만의 응답 코드가 내려진 요청에 대해 Resource 와 Subresource 별로 그룹화 한 분당 요청 수 추이
	[3xx] 분당 응답 요청 현황	kube-apiserver 요청 중 300 이상 400 미만의 응답 코드가 내려진 요청 수에 대한 분당 현황
	[3xx] 분당 응답 코드별 요청 수 추이	kube-apiserver 요청 중 300 이상 400 미만의 응답 코드가 내려진 요청 수에 대한 응답 코드별 분당 현황
	[3xx] 분당 Resource & Subresource 별 요청 수 추이	kube-apiserver 요청 중 300 이상 400 미만의 응답 코드가 내려진 요청에 대해 Resource 와 Subresource 별로 그룹화 한 분당 요청 수 추이
	[400~] 분당 응답 요청 현황	kube-apiserver 요청 중 400 이상의 응답 코드가 내려진 요청 수에 대한 분당 현황
	[400~] 분당 응답 코드별 요청 수 추이	kube-apiserver 요청 중 400 이상의 응답 코드가 내려진 요청 수에 대한 응답 코드별 분당 현황
	[400~] 분당 Resource & Subresource 별 요청 수 추이	kube-apiserver 요청 중 400 이상의 응답 코드가 내려진 요청에 대해 Resource 와 Subresource 별로 그룹화 한 분당 요청 수 추이

분류	메트릭	설명
Go Metrics	고루틴 수	현재 존재하는 고루틴 수
	스레드 수	현재 존재하는 OS 스레드 수
현재 실행 중인 요청 수 (1초)	최근 1초 내 실행 중인 요청 수	데이터 수집 시 kube-apiserver 에서 1초 내에 진행되고 있던 요청 수

kube-apiserver 메트릭스 조회

홈 화면 > 클러스터 프로젝트 선택 > 클러스터 > kube-apiserver 메트릭스 조회

ⓘ 요구 사항

- 쿠버네티스 클러스터 프로젝트(**CP**) 읽기 권한이 필요합니다.
- 와탭 쿠버네티스 에이전트 1.5.6 버전 이상이 필요합니다.

kube-apiserver는 **컨트롤 플레인**(Control Plane) 내/외부 요청을 처리하는 컴포넌트입니다. **kube-apiserver 메트릭스 조회** 메뉴를 통해 kube-apiserver에서 수집되는 지표의 원본 데이터 정확성(Data Consistency)을 확인할 수 있습니다.

기본 화면 안내


time	component	dryRun	group	resource	scope	subresource	verb	vers
03/25 17:44:00						/healthz	GET	
03/25 17:44:00						/livez	GET	
03/25 17:44:00						/readyz	GET	
03/25 17:44:00						openapi/v3	GET	
03/25 17:44:00						openapi/v3/	GET	
03/25 17:44:00	apiserver					api	GET	
03/25 17:44:00	apiserver					apis	GET	
03/25 17:44:00	apiserver			configmaps	cluster		LIST	v1
03/25 17:44:00	apiserver			configmaps	namespace		LIST	v1
03/25 17:44:00	apiserver			configmaps	namespace		WATCH	v1
03/25 17:44:00	apiserver			configmaps	resource		GET	v1
03/25 17:44:00	apiserver			configmaps	resource		PUT	v1

kube-apiserver 메트릭스 조회 메뉴는 kube-apiserver 대상으로 수집한 모니터링 원본 데이터를 **카테고리**별로 조회하는 기능을 제공합니다. 최근 5분 동안 수집한 데이터가 1분 간격으로 반영됩니다.

- 상단 왼쪽 **시간 선택자**를 통해 과거 특정 시점의 데이터를 조회할 수 있습니다. 과거 데이터 조회 시 원하는 옵션 선택 후 **🔍 검색** 아이콘을 클릭하세요.
- 조회할 **최대 개수**를 설정할 수 있습니다.
- 메트릭스를 필터링해 원본 데이터를 필요에 따라 가공할 수 있습니다.
- 상단 오른쪽 **다운로드** 아이콘을 선택해 원하는 데이터를 **CSV** 형식의 파일로 다운로드할 수 있습니다.

필터

필터 옵션 활용 시 원하는 을 선택해 조회할 수 있습니다. `and` 조건을 통해 여러 개의 필터를 적용할 수 있습니다.

1.  필터 입력 창에서 + 더하기 아이콘 선택 시 다음과 같이 필터 추가하기 창이 나타납니다.



- 필터 키와 조건을 선택하세요.
 - 조건에 맞는 값을 선택하세요.
2. 필터 설정 후 적용 버튼을 선택해 해당 필터를 적용하세요.
 3. 여러 개의 필터를 추가하려면 필터 입력 창 또는 필터 추가하기 창에서 + 더하기 아이콘을 선택하세요.

kube-apiserver 모니터링 활성화

와탭 쿠버네티스 에이전트 설치 시 다운로드한 `yaml` 파일을 다음 안내에 따라 수정해 kube-apiserver 모니터링을 활성화하세요.

1. `whatap-master-agent` deployment의 spec 하위에 kube-apiserver 모니터링을 위한 에이전트 컨테이너 spec을 추가하세요.

```
containers:
  - name: whatap-control-plane-helper
    image: whatap/kube_mon
    imagePullPolicy: Always
    ports:
      - containerPort: 9496
    command: ["/data/agent/master/whatap_control_plane_helper"]
    args: ["-collect_control_plane_monitoring_enabled=true"]
```

다음 예시를 참조하세요.

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: whatap-master-agent
  namespace: whatap-monitoring
spec:
  replicas: 1
```

```

selector:
  matchLabels:
    name: whatap-master-agent
template:
  metadata:
    labels:
      name: whatap-master-agent
  spec:
    containers:
      - name: whatap-control-plane-helper
        image: whatap/kube_mon
        imagePullPolicy: Always
        ports:
          - containerPort: 9496
        command: ["/data/agent/master/whatap_control_plane_helper"]
        args: ["-collect_control_plane_monitoring_enabled=true"]
      - name: whatap-master-agent
        image: whatap/kube_mon
        imagePullPolicy: Always
        resources:
          requests:
            memory: 300Mi
            cpu: 100m
          limits:
            memory: 350Mi
            cpu: 200m
        ports:
          - containerPort: 6600
        command: ["/bin/entrypoint.sh"]
        env:
          - name: WHATAP_LICENSE
            value: "xxxxxxxx-12345xxxxx-x00e000000xxx8"
          - name: WHATAP_HOST
            value: "13.124.11.223/13.209.172.35"
          - name: WHATAP_PORT
            value: "6600"
          - name: WHATP_MEM_LIMIT
            valueFrom:
              resourceFieldRef:
                containerName: whatap-master-agent
                resource: limits.memory
        volumeMounts:
          - name: start-script-volume
            mountPath: /bin/entrypoint.sh
            readOnly: true
            subPath: entrypoint.sh
          - mountPath: /whatap_conf
            name: whatap-config-volume
        volumes:
          - name: start-script-volume
            configMap:
              defaultMode: 0700

```

```

name: master-start-script
- name: whatap-config-volume
emptyDir: {}
serviceAccount: whatap

```

2. `whatap-master-agent` deployment의 `whatap-master-agent` 컨테이너 spec에서 env 설정에 다음 내용을 추가하세요.

```

env:
- name: collect_control_plane_monitoring_enabled
value: "true"

```

다음 예시를 참조하세요.

```

apiVersion: apps/v1
kind: Deployment
metadata:
name: whatap-master-agent
namespace: whatap-monitoring
spec:
replicas: 1
selector:
matchLabels:
name: whatap-master-agent
template:
metadata:
labels:
name: whatap-master-agent
spec:
containers:
- name: whatap-control-plane-helper
image: whatap/kube_mon
imagePullPolicy: Always
ports:
- containerPort: 9496
command: ["/data/agent/master/whatap_control_plane_helper"]
args: ["-collect_control_plane_monitoring_enabled=true"]
- name: whatap-master-agent
image: whatap/kube_mon
imagePullPolicy: Always
resources:
requests:
memory: 300Mi
cpu: 100m
limits:
memory: 350Mi
cpu: 200m
ports:
- containerPort: 6600
command: ["/bin/entrypoint.sh"]
env:
- name: collect_control_plane_monitoring_enabled

```



```

value: "true"
- name: WHATAP_LICENSE
value: "xxxxxxx-12345xxxxx-x00e000000xxx8"
- name: WHATAP_HOST
value: "13.124.11.223/13.209.172.35"
- name: WHATAP_PORT
value: "6600"
- name: WHATP_MEM_LIMIT
valueFrom:
  resourceFieldRef:
    containerName: whatap-master-agent
    resource: limits.memory
volumeMounts:
- name: start-script-volume
  mountPath: /bin/entrypoint.sh
  readOnly: true
  subPath: entrypoint.sh
- mountPath: /whatap_conf
  name: whatap-config-volume
volumes:
- name: start-script-volume
  configMap:
    defaultMode: 0700
    name: master-start-script
- name: whatap-config-volume
  emptyDir: {}
serviceAccount: whatap

```

3. 컨트롤 플레인 모니터링 성능 지표를 수집하기 위해 `whatap` ClusterRole에 다음의 권한을 추가하세요.

```

rules:
- nonResourceURLs: ["/metrics"]
  verbs: ["*"]

```

다음 예시를 참조하세요.

```

apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  annotations:
    rbac.authorization.kubernetes.io/autoupdate: "true"
  name: whatap
  namespace: whatap-monitoring
rules:
- apiGroups: ["*"]
  resources: ["namespaces", "pods", "pods/log", "configmaps", "services",
"endpoints", "daemonsets", "ingresses", "deployments", "nodes", "persistentvolumes", "persistentvolumeclaims", "events", "replicasets", "roles",
"rolebindings", "clusterroles", "clusterrolebindings", "jobs", "cronjobs", "statefulsets", "serviceaccounts", "configmaps", "storageclasses",
"horizontalpodautoscalers" ]
  verbs: ["get", "list", "watch"]
- apiGroups: [""]

```

```
resources: ["pods/exec"]
verbs: ["create"]
- apiGroups: [""]
resources: ["configmaps"]
verbs: ["*"]
- nonResourceURLs: ["/metrics"]
verbs: ["*"]
---
```

ⓘ `kubectl apply` 또는 `kubectl edit` 명령을 통해 kube-apiserver 모니터링을 활성화할 수도 있습니다.

kube-apiserver 메트릭

분류	메트릭	설명
API GET 요청	분당 GET 요청 현황	kube-apiserver 의 GET 요청에 대한 분당 요청 수
	초당 GET 요청 수 추이	kube-apiserver 의 GET 요청에 대한 초당 요청 수 추이
	Resource & Subresource 별 분당 GET 요청 수 추이	kube-apiserver 에 들어온 GET 요청의 Resource 와 Subresource 별 분당 요청 수 추이
API POST 요청	분당 POST 요청 현황	kube-apiserver 의 POST 요청에 대한 분당 요청 수
	초당 POST 요청 수 추이	kube-apiserver 의 POST 요청에 대한 초당 요청 수 추이
	Resource & Subresource 별 POST 분당 요청 수 추이	kube-apiserver 에 들어온 POST 요청에 대해 Resource 와 Subresource 별로 그룹화 한 분당 요청 수 추이
API 응답 코드별 요청	[2xx] 분당 응답 요청 현황	kube-apiserver 요청 중 200 이상 300 미만의 응답 코드가 내려진 요청 수에 대한 분당 현황
	[2xx] 분당 응답 코드별 요청 수 추이	kube-apiserver 요청 중 200 이상 300 미만의 응답 코드가 내려진 요청 수에 대한 응답 코드별 분당 현황
	[2xx] 분당 Resource & Subresource 별 요청 수 추이	kube-apiserver 요청 중 200 이상 300 미만의 응답 코드가 내려진 요청에 대해 Resource 와 Subresource 별로 그룹화 한 분당 요청 수 추이
	[3xx] 분당 응답 요청 현황	kube-apiserver 요청 중 300 이상 400 미만의 응답 코드가 내려진 요청 수에 대한 분당 현황

분류	메트릭	설명
	[3xx] 분당 응답 코드별 요청 수 추이	kube-apiserver 요청 중 300 이상 400 미만의 응답 코드가 내려진 요청 수에 대한 응답 코드별 분당 현황
	[3xx] 분당 Resource & Subresource 별 요청 수 추이	kube-apiserver 요청 중 300 이상 400 미만의 응답 코드가 내려진 요청에 대해 Resource 와 Subresource 별로 그룹화 한 분당 요청 수 추이
	[400~] 분당 응답 요청 현황	kube-apiserver 요청 중 400 이상의 응답 코드가 내려진 요청 수에 대한 분당 현황
	[400~] 분당 응답 코드별 요청 수 추이	kube-apiserver 요청 중 400 이상의 응답 코드가 내려진 요청 수에 대한 응답 코드별 분당 현황
	[400~] 분당 Resource & Subresource 별 요청 수 추이	kube-apiserver 요청 중 400 이상의 응답 코드가 내려진 요청에 대해 Resource 와 Subresource 별로 그룹화 한 분당 요청 수 추이
Go Metrics	고루틴 수	현재 존재하는 고루틴 수
	스레드 수	현재 존재하는 OS 스레드 수
현재 실행 중인 요청 수 (1초)	최근 1초 내 실행 중인 요청 수	데이터 수집 시 kube-apiserver 에서 1초 내에 진행되고 있던 요청 수

네임스페이스 현황

홈 화면 > 프로젝트 선택 > 클러스터 > 네임스페이스 현황

네임스페이스 현황 메뉴를 통해 와탭 쿠버네티스 클러스터 프로젝트 및 하위 네임스페이스 프로젝트 현황을 확인할 수 있습니다. 네임스페이스 프로젝트에서는 접근할 수 없는 메뉴입니다.

The screenshot shows the 'Namespace Status' page in the AWS CloudWatch console. The page title is 'Kubernetes Monitoring Demo(EKS) - 네임스페이스 현황'. It features a grid of namespace cards. The first two cards are expanded to show detailed metrics:

Kubernetes Monitoring Demo(EKS)			
노드	3	컨테이너	57
노드 CPU 코어	6	Pod	39
노드 메모리	24.5G	애플리케이션	10
Namespace	6	Deployment	9
ReplicaSet	9		

K8s Namespace Project DEMO			
노드	3	컨테이너	20
노드 CPU 코어	6	Pod	20
노드 메모리	24.5G	애플리케이션	20
Namespace	1	Deployment	1
ReplicaSet	1		

The remaining namespace cards shown are: 'amazon-guardduty', 'default', 'dev', 'dev-prod', 'dev-test', and 'kube-node-lease'.

네임스페이스 현황 메뉴는 프로젝트 목록 화면과 유사하지만 다음과 같은 차이가 있습니다.

• 프로젝트 목록

네임스페이스 프로젝트 생성 후 해당 프로젝트를 다른 그룹으로 이동할 수 있지만 네임스페이스를 한 눈에 파악하기 어렵습니다.

• 네임스페이스 현황

- 클러스터 프로젝트와 동일한 그룹에 속하지 않는 네임스페이스도 확인할 수 있습니다.
- 한 눈에 관련 프로젝트를 모두 확인할 수 있습니다.

클러스터 프로젝트의 경우 Kubernetes 플랫폼을 나타내는 **파란색 Kubernetes** 태그가 표시됩니다. 네임스페이스 프로젝트에는 **Namespace** 태그가 표시됩니다. 위젯 왼쪽 위에 타이틀은 프로젝트 이름이며, 네임스페이스 이름은 가장 아래에 **Namespace** 태그 옆에 표시됩니다.

Pod 시작 분석

홈 화면 > 프로젝트 선택 > 분석 > Pod 시작 분석

❗ 와탭 노드 에이전트 버전 **1.4.2 이상**에서 지원합니다. 해당 버전 미만인 경우 [기존 메뉴](#)(구 Pod 초기화 성능) 화면을 제공합니다.

원활한 클러스터 운영을 위해 Pod가 안정적으로 동작하기까지의 성능을 분석하고 Pod에 할당된 리소스 크기를 최적화할 필요가 있습니다. **Pod 시작 분석** 메뉴는 Pod가 시작한 후 약 5분 동안의 변화를 추적합니다. **Pending** 상태에서 **Running** 상태로, 그리고 안정적인 **Stable Running** 상태로 이어지는 3단계 구간의 자원 사용량을 분석해 리소스 할당량을 최적화할 수 있습니다.

이를 통해 운영자는 자원 효율성 향상 및 비용 절감, 확장성 향상, 스케일 용이성 등의 이점을 얻을 수 있습니다.

기본 화면 안내

CP Kubernetes Monitoring Demo(EKS) ▾ Pod 시작 분석
🔊 🔔 🗑️ 🗨️ 👤

조회 시각 13:34:33
30일 이내 생성된 Pod만 표시
Name

Pod 시작 성능을 분석하여 컨테이너 자원을 효율적으로 관리하세요.

와탭이 제안하는 Pod 시작 성능 분석 데이터를 통해 **적절한 리소스 요청 및 제한을 설정**하세요. 시작 분석을 원하는 Pod를 선택하면 해당 Pod가 시작된 후 약 5분 동안의 시작 성능을 분석합니다. **Pending > Running > Stable Running**으로 나뉘어진 구간별 자원 사용량을 파악하여 효율적으로 자원을 사용하고, 안정적으로 클러스터를 운영할 수 있습니다.

*Pod 시작 성능 기능은 최근 30일 데이터를 이용하기 때문에 생성일로부터 30일이 지난 Pod의 시작 성능은 분석할 수 없습니다.

^ 접어두기

Name	QoS Class	Restart Count	CPU Limits	Memory Limits	CPU Requests	Memory Requests	Start Time
apm-java-demo-7b69b...	BestEffort	0	2000m	7.61GiB	-	-	2024/02/23 16:55:39
aws-guardduty-agent...	Burstable	0	1000m	1024MiB	200m	256MiB	2024/03/13 08:24:12
aws-guardduty-agent...	Burstable	0	1000m	1024MiB	200m	256MiB	2024/03/14 00:10:47
aws-guardduty-agent-t...	Burstable	0	1000m	1024MiB	200m	256MiB	2024/03/14 00:13:35
aws-guardduty-agent-t...	Burstable	0	1000m	1024MiB	200m	256MiB	2024/03/14 00:13:49
aws-guardduty-agent...	Burstable	0	1000m	1024MiB	200m	256MiB	2024/03/14 00:11:21
aws-node-brktq	Burstable	0	2000m	1.88GiB	25m	-	2024/03/14 00:10:47
aws-node-g7skq	Burstable	0	2000m	1.88GiB	25m	-	2024/03/14 00:13:35
aws-node-jfr4k	Burstable	0	2000m	7.61GiB	25m	-	2024/02/23 16:52:33

상단 옵션 영역

상단 ① 영역에서 다음과 같은 옵션을 활용할 수 있습니다.

- 새로 고침 아이콘을 선택해 ② 영역의 Pod 목록을 다시 불러올 수 있습니다.
- 30일 이내 생성된 Pod만 표시 토글을 활성화해 ② 영역의 Pod 목록에서 30일 이내의 Pod만을 조회할 수 있습니다.

ⓘ Pod 시작 분석 메뉴는 Pod 생성일로부터 최근 30일 이내의 범위 안에서 5분간의 Pod 시작 데이터를 분석합니다. 즉 생성 후 30일이 지난 Pod의 경우 시작 성능을 분석할 수 없습니다.

- Pod 검색 입력창에 문자를 입력해 원하는 Pod를 검색할 수 있습니다.

Pod 목록 영역

② 영역에서 Pod 목록을 확인할 수 있습니다. Pod 목록이 펼쳐진 테이블의 헤더 컬럼은 다음의 정보를 포함하고 있습니다.

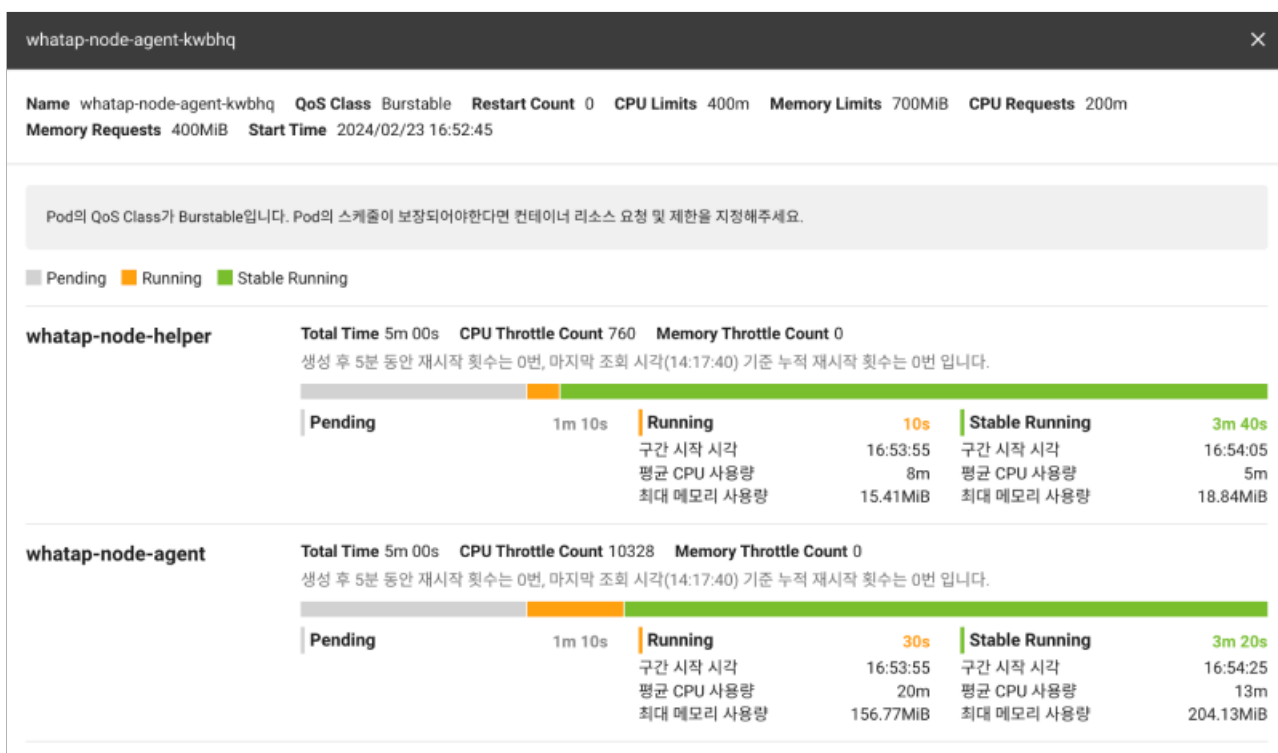
- **QoS Class:** Pod의 Quality of Service를 의미합니다. QoS 클래스는 리소스 부족 시 Pod의 우선 순위를 결정합니다. 우선 순위가 높은 Pod가 우선 순위가 낮은 Pod보다 먼저 종료됩니다.
 - **Guaranteed:** 노드에서 사용 가능한 리소스를 보장받고 다른 Pod보다 우선적으로 할당합니다. 가장 마지막에 종료됩니다.
 - Pod의 모든 컨테이너가 CPU 및 메모리 request를 가지고 있는 경우입니다.
 - Pod의 모든 컨테이너가 CPU 및 메모리 limit을 가지고 있으며 해당 값이 request와 동일한 경우입니다.
 - **Burstable:** Pod는 노드 리소스가 부족한 경우에는 일부 리소스를 할당합니다.
 - Guaranteed 클래스의 조건을 충족하지 못하는 경우입니다.
 - Pod의 컨테이너 중 하나라도 CPU 및 메모리에 대한 request 또는 limit이 설정된 경우입니다.
 - **BestEffort:** 리소스 request나 limit이 없기에 다른 Pod보다 우선 순위가 낮습니다. 가장 먼저 종료됩니다.
 - Pod의 모든 컨테이너가 CPU 및 메모리에 대한 request와 limit이 없는 경우입니다.
- **Restart Count:** Pod 내 컨테이너의 총 재시작 횟수입니다.
- **CPU/Memory Limits:** Pod 내 컨테이너에 설정된 CPU/Memory Limits 총합(`resources.limits`)입니다.
- **CPU/Memory Requests:** Pod 내 컨테이너에 설정된 CPU/Memory Requests 총합(`resources.requests`)입니다.

- **Start Time**: Pod 생성 시각(`metadata.startTime`)입니다.

- ① • QoS Class에 대한 자세한 내용은 [다음 링크](#)를 참조하세요.
 - Pod 및 컨테이너 리소스 관리에 대한 자세한 내용은 [다음 링크](#)를 참조하세요.

상세 보기 안내

- 2 영역 Pod 목록에서 분석을 원하는 항목의 가장 왼쪽에 [상세 보기](#) 버튼을 선택하세요. 선택한 Pod의 컨테이너별 시작 성능을 구간별로 분석한 내용을 다음과 같이 확인할 수 있습니다.



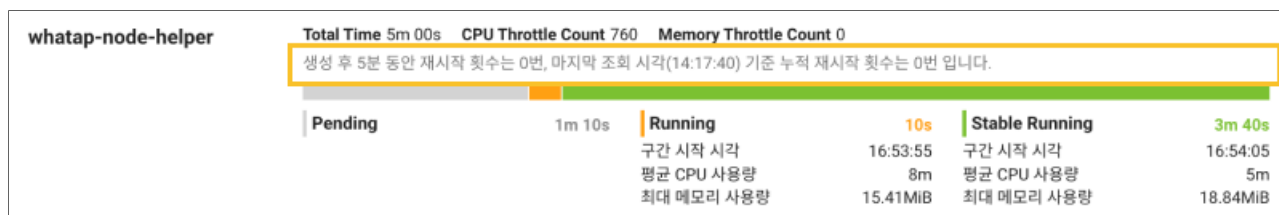
- **Pending**: Pod를 생성한 후 **Running** 상태로 진입하는 구간입니다.
- **Running**: Pod가 **Pending** 상태를 지나 안정화되기 전까지의 구간입니다.
- **Stable Running**: Pod가 **Running** 상태를 지나 안정화된 구간입니다.

ⓘ 안정화된 상태(Stable Running)

와탭에서는 Pod 생성 직후 약 5분 동안 컨테이너의 CPU Limit 대비 사용량의 평균을 계산합니다. 이 평균값보다 CPU 사용량이 적어지는 시점을 안정화 시작 상태라고 정의합니다.

상세 분석하기

- 5분 동안 한 컨테이너가 여러 번 재시작했다면 가장 처음 시작한 컨테이너의 정보만 표기합니다. 컨테이너별 재시작 횟수를 확인하세요. 재시작 횟수가 많다면 해당 컨테이너가 정상 실행되고 있는지 점검하세요.



- Pending** 구간이 길어진다면 Pod가 **Running** 상태로 진입하기까지 오랜 시간이 걸렸다는 의미입니다. Pod의 **Pending** 상태를 유발할 수 있는 요인이 있는지 점검해 보세요.

ⓘ Pod의 **pending** 상태에 대한 자세한 내용은 [다음 링크](#)를 참조하세요.

- Running** 구간이 길다면 Pod 내 컨테이너가 처음 실행될 때 리소스 사용량이 많은 구간이 있다는 의미입니다. 컨테이너 실행 과정에 문제가 없는지 살펴보세요.
- 전반적인 초기 자원 사용량이 많다면 컨테이너 리소스 요청 및 제한을 늘리는 것을 고려해 보세요.

Pod 시작 분석

홈 화면 > 프로젝트 선택 > 분석 > Pod 시작 분석

ⓘ 와탭 노드 에이전트 버전 1.4.2 미만

- 신규 [Pod 시작 분석](#) 메뉴 대신 기존 메뉴(구 **Pod 초기화 성능**) 화면을 제공합니다.
- 클러스터 프로젝트(**CP**)인 경우만 접근할 수 있습니다.

Springboot를 포함한 JVM 기반 애플리케이션은 초기화 시점에 많은 자원을 사용해 Liveness 체크에 실패하는 경우가 있습니다. 만약 일정 횟수 이상 Liveness 체크에 실패하는 경우 해당 Pod가 삭제되고 무한 재시작 현상이 발생하여 배포 실패로 이어질 수 있습니다.

와탭 쿠버네티스의 [Pod 시작 분석](#) 기능을 활용하면 Pod 시작 시 피크 사용량과 초기화 시간을 분석하여 최적의 Initial Delay(Liveness 체크 면제 시간) 설정에 도움을 받을 수 있습니다.

기본 화면 안내

CP Kubernetes Monitoring (demo@K8S) Pod 초기화 성능

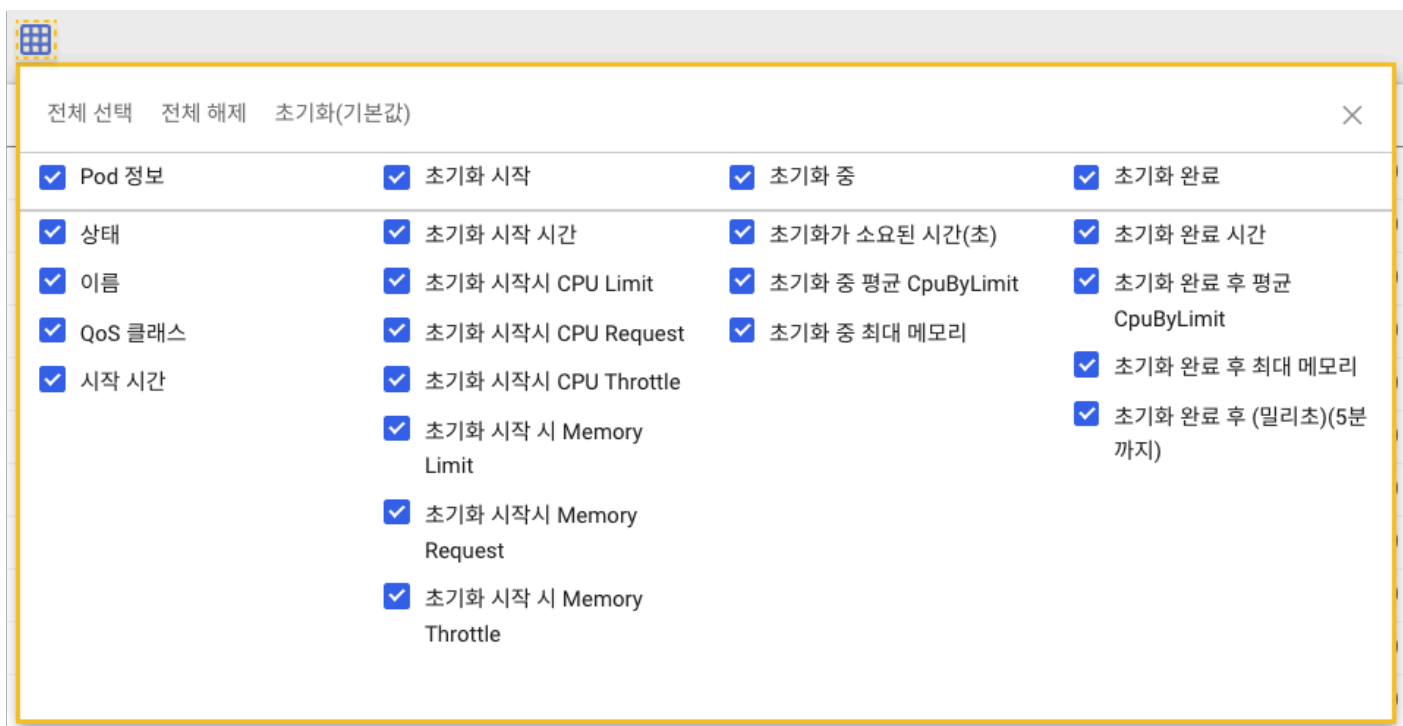
Running 43

Pod Name, Qos Class

상태	이름	QoS 클래스	시작 시간	초기화 시작 시간	초기화 시작시 CPU Limit	초기화 시작시 CPU Request	초기화 시작시 CPU Throttle	초기화 시작 시 Memory Limit	초기화 시작
✓	whatap-node-agent-lvlhx	Burstable	2023-11-08T02:28:16Z	2023-11-08 11:29:30	400	200	31	700 M	
✓	whatap-node-agent-hqcg2	Burstable	2023-11-02T05:09:16Z	2023-11-02 14:09:55	400	200	69	700 M	
✓	whatap-node-agent-4tbg5	Burstable	2023-11-08T02:28:16Z	2023-11-08 11:29:30	400	200	40	700 M	
✓	whatap-master-agent-7c76...	Burstable	2023-11-08T02:27:39Z	2023-11-08 11:28:35	200	100	501	350 M	
✓	whatap-virtual-deployment...	Burstable	2023-11-08T02:26:13Z		300	50	0	256 M	
✓	whatap-virtual-deployment...	Burstable	2023-11-08T02:26:13Z		300	50	0	256 M	
✓	whatap-virtual-deployment...	Burstable	2023-11-08T02:26:13Z		300	50	0	256 M	
✓	whatap-virtual-deployment...	Burstable	2023-11-08T02:26:19Z		300	50	0	256 M	
✓	whatap-virtual-deployment...	Burstable	2023-11-08T02:26:18Z		300	50	0	256 M	
✓	whatap-virtual-deployment...	Burstable	2023-11-08T02:26:24Z	2023-11-08 11:26:45	300	50	23	256 M	
✓	whatap-virtual-deployment...	Burstable	2023-11-08T02:26:31Z	2023-11-08 11:26:45	300	50	12	256 M	
✓	whatap-virtual-deployment...	Burstable	2023-11-08T02:26:24Z	2023-11-08 11:26:45	300	50	9	256 M	
✓	whatap-virtual-deployment...	Burstable	2023-11-08T02:26:31Z	2023-11-08 11:26:45	300	50	31	256 M	
✓	whatap-virtual-deployment...	Burstable	2023-11-08T02:26:30Z	2023-11-08 11:26:45	300	50	14	256 M	
✓	whatap-virtual-deployment...	Burstable	2023-11-08T02:26:31Z	2023-11-08 11:26:45	300	50	31	256 M	
✓	whatap-virtual-deployment...	Burstable	2023-11-08T02:26:24Z	2023-11-08 11:26:45	300	50	12	256 M	
✓	whatap-virtual-deployment...	Burstable	2023-11-08T02:26:24Z	2023-11-08 11:26:45	300	50	11	256 M	
✓	whatap-virtual-deployment...	Burstable	2023-11-08T02:26:31Z	2023-11-08 11:26:45	300	50	17	256 M	
✓	whatap-virtual-deployment...	Burstable	2023-11-08T02:26:24Z	2023-11-08 11:26:45	300	50	20	256 M	
✓	testing-http-sender-58d44d...	BestEffort	2023-11-02T05:21:29Z	2023-11-02 14:24:40	0	0	0	0	

- ① 영역 좌측 상단에서 상태별 Pod 개수를 확인할 수 있습니다.
- ① 영역 우측 상단에서 새로 고침 아이콘을 선택해 새로 고침할 수 있습니다.
- ① 영역 좌측에서 컬럼 선택 아이콘을 선택해 컬럼을 선택할 수 있습니다.
- ① 영역 우측 입력창에서 Pod 이름 또는 QoS 클래스를 검색할 수 있습니다.
- ② 영역에서 원하는 컬럼을 선택 시 해당 컬럼 기준으로 목록을 정렬해 조회할 수 있습니다.

컬럼 안내



Pod 시작 분석 목록은 다음과 같이 Pod 정보와 Pod 초기화 타임라인에 따라 컬럼을 분류할 수 있습니다.

Pod 정보

Pod의 이름과 상태, QoS 클래스, 시작 시간에 대한 정보를 확인할 수 있습니다.

- **상태:** Pod의 라이프 사이클을 의미합니다.
 - **II Pending:** 쿠버네티스 클러스터 내부에서 Pod 시작이 승인되었지만 컨테이너가 생성되지 않은 상태를 의미합니다.
 - **🟢 Running:** Pod가 노드에 바인딩되고 모든 컨테이너가 생성되고 하나 이상의 컨테이너가 아직 실행 중이거나 시작 또는 재시작 중에 있는 상태를 의미합니다.
 - **🟡 Succeeded:** Pod의 모든 컨테이너가 성공적으로 종료되었으며 재시작되지 않은 상태를 의미합니다.
 - **🔴 Failed:** Pod의 컨테이너 중 하나 이상의 컨테이너가 실패해 종료된 상태를 의미합니다.
 - **🟠 Unknown:** 어떤 이유로 Pod와 통신할 수 없는 상태를 의미합니다.

- **QoS 클래스:** Pod의 Quality of Service를 의미합니다. QoS 클래스는 리소스 부족 시 Pod의 우선 순위를 결정합니다. 우선 순위가 높은 Pod가 우선 순위가 낮은 Pod보다 먼저 종료됩니다.
 - **Guaranteed:** 노드에서 사용 가능한 리소스를 보장받고 다른 Pod보다 우선적으로 할당합니다. 가장 마지막에 종료됩니다.
 - Pod의 모든 컨테이너가 CPU 및 메모리 request를 가지고 있는 경우입니다.
 - Pod의 모든 컨테이너가 CPU 및 메모리 limit을 가지고 있으며 해당 값이 request와 동일한 경우입니다.
 - **Burstable:** Pod는 노드 리소스가 부족한 경우에는 일부 리소스를 할당합니다.
 - Guaranteed 클래스의 조건을 충족하지 못하는 경우입니다.
 - Pod의 컨테이너 중 하나라도 CPU 및 메모리에 대한 request 또는 limit이 설정된 경우입니다.
 - **BestEffort:** 리소스 request나 limit이 없기에 다른 Pod보다 우선 순위가 낮습니다. 가장 먼저 종료됩니다.
 - Pod의 모든 컨테이너가 CPU 및 메모리에 대한 request와 limit이 없는 경우입니다.
- **시작 시간:** Pod가 시작된 시간 또는 생성된 시간을 의미합니다.

초기화 시작

초기화 시작은 최초 메트릭스 데이터가 수집된 시점을 의미합니다.

- **초기화 시작 시간:** 초기화가 시작된 시점에 대한 데이터로 첫 데이터 수집 시간을 의미합니다.
- **초기화 시작 시 CPU Limit:** 초기화 시작 시 CPU limit을 의미합니다.
- **초기화 시작 시 CPU Request:** 초기화 시작 시 CPU request를 의미합니다.
- **초기화 시작 시 CPU Throttle Count:** 초기화 시작 시 컨테이너에서 CPU 제한을 초과하는 사용 시도 발생 시 증가하는 값을 의미합니다.
- **초기화 시작 시 Memory Limit:** 초기화 시작 시 설정한 메모리 limit을 의미합니다.
- **초기화 시작 시 Memory Request:** 초기화 시작 시 요청한 메모리 request를 의미합니다.
- **초기화 시작 시 Memory Throttle Count:** 초기화 시작 시 컨테이너에서 메모리 제한을 초과하는 사용 시도 발생 시 증가하는 값을 의미합니다.

초기화 중

초기화 중은 초기화 시작 후 5분간 데이터의 CpuByLimit 평균 값보다 낮은 CpuByLimit 값의 데이터가 발생한 이전 시점을 의미합니다.

- **초기화가 소요된 시간(초):** Pod 초기화에 소요된 시간을 의미합니다.

초기화 시작 시 5분간의 데이터 중 CpuByLimit의 평균 값보다 낮은 CpuByLimit 값이 수집된 시간 - 첫 데이터 수집 시간

- **초기화 중 평균 CpuByLimit**: 초기화 중 사용한 CpuByLimit의 평균 값을 의미합니다.
- **초기화 중 최대 메모리**: 초기화 중 해당 컨테이너에서 사용한 메모리 최대 사용량을 의미합니다.

ⓘ CpuByLimit(`cpu_per_quota`)은 CPU Limit 기준 CPU 전체 사용률을 의미합니다.

초기화 완료

초기화 완료는 초기화 시작 후 5분간 데이터의 CpuByLimit 평균 값보다 낮은 CpuByLimit 값의 데이터가 발생한 시점을 의미합니다.

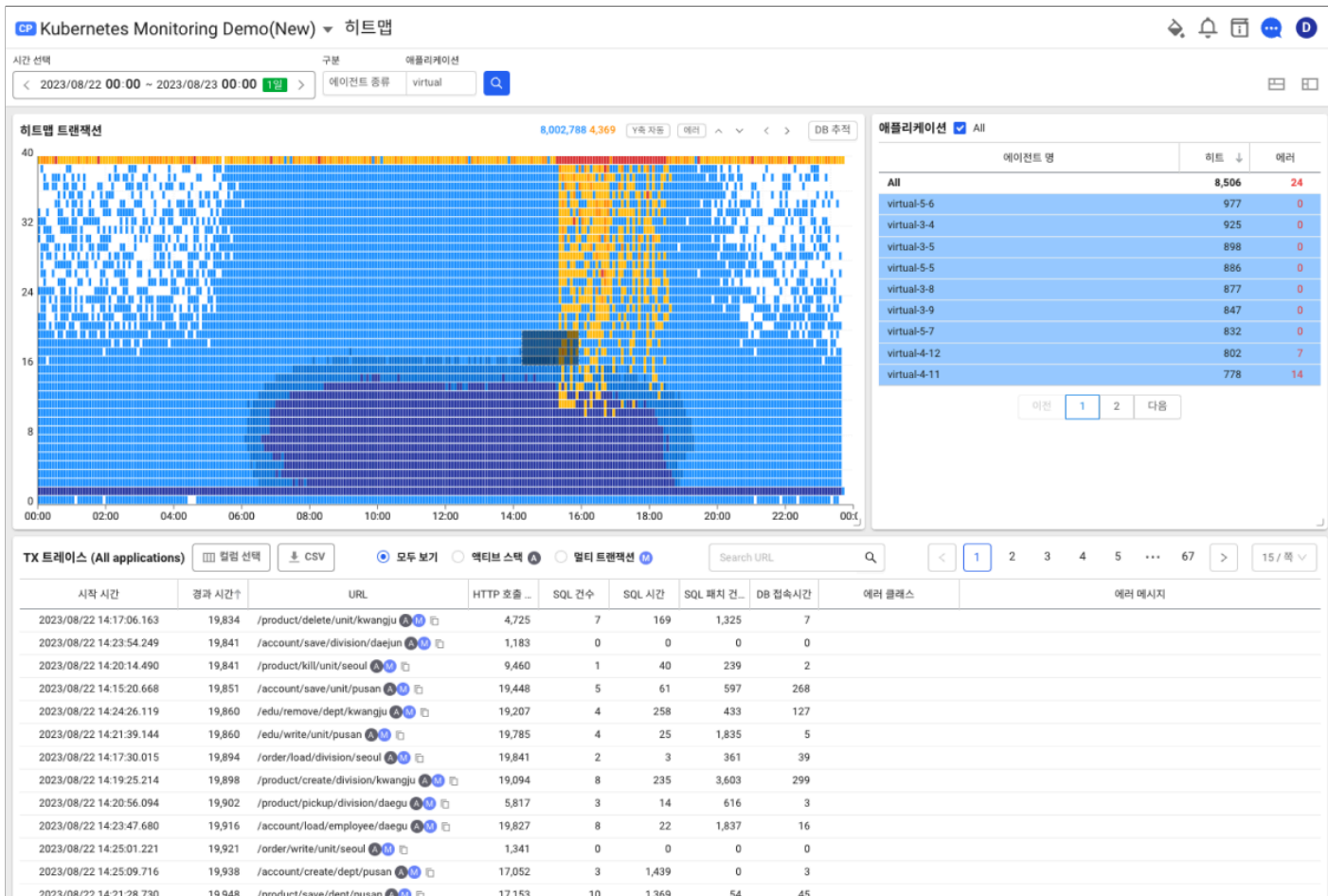
- **초기화 완료 시간**: 초기화 완료 시점을 의미합니다.
- **초기화 완료 후 평균 CpuByLimit**: 초기화 완료 후 사용한 CpuByLimit의 평균 값을 의미합니다.
- **초기화 완료 후 최대 메모리**: 초기화 완료 후 해당 컨테이너에서 사용한 메모리 최대 사용량을 의미합니다.
- **초기화 완료 후(초)(5분까지)**: 초기화 완료 후 5분 동안의 밀리초 단위 시간을 의미합니다.

애플리케이션 히트맵

홈 화면 > 프로젝트 선택 > 분석 > 히트맵

컨테이너 내부 애플리케이션의 히트맵 정보를 확인할 수 있습니다. 히트맵은 트랜잭션 수행 건수 맵으로 응답 시간 분포도를 표시합니다. 검색 조건은 시간 선택, 구분, 애플리케이션으로 구분 조건은 다음과 같습니다.

분류	설명
에이전트	컨테이너 내부에 설치된 애플리케이션 에이전트 이름
에이전트 종류	에이전트가 그룹화된 에이전트 종류 (OKIND)
에이전트 서버	에이전트가 실행 중인 쿠버네티스 노드 이름



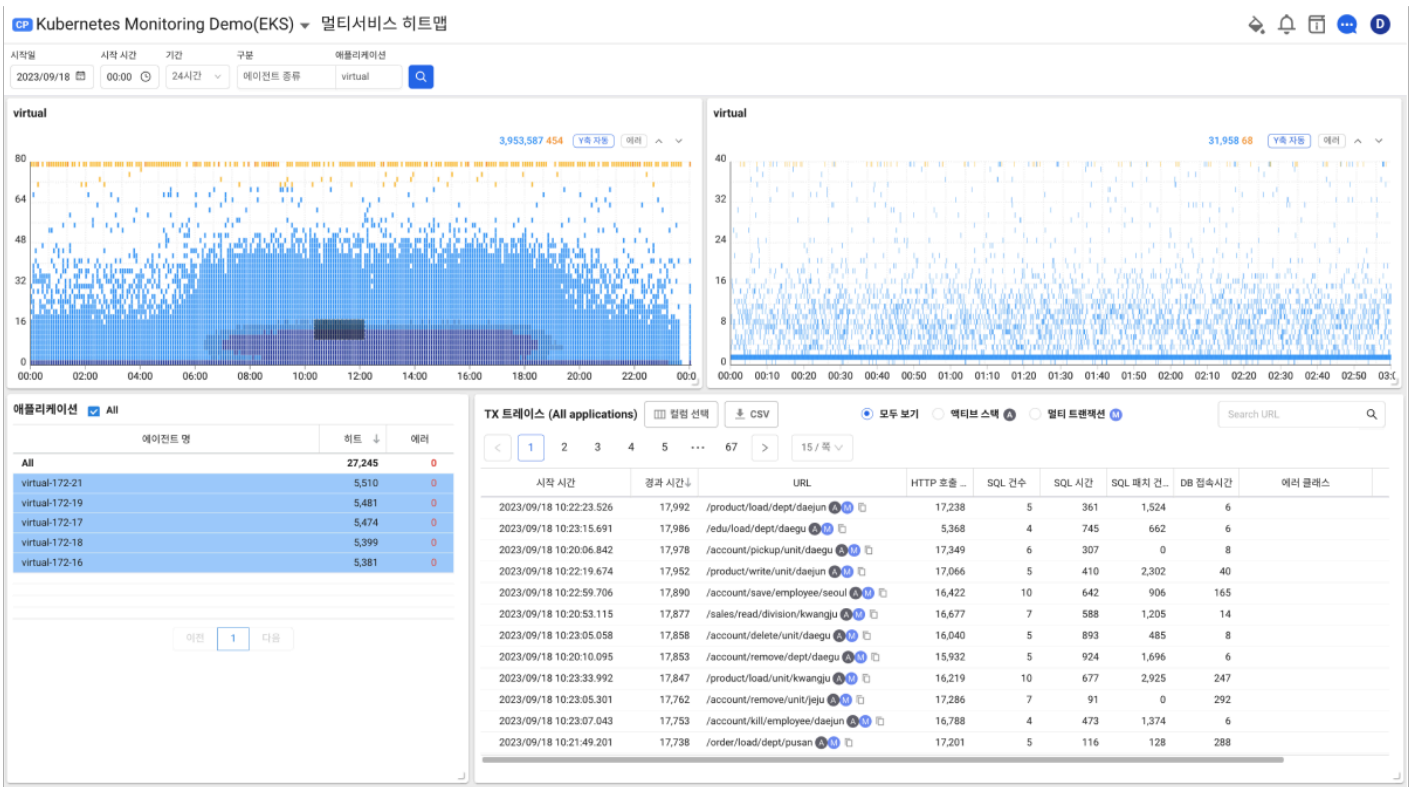
히트맵 특정 구간을 드래그하면 트레이스 정보를 확인할 수 있습니다. 트레이스 데이터는 URL 기반의 검색이 가능하며 액티브 스택이 존재하는 트레이스, 멀티 트랜잭션으로 수행하는 트레이스 정보를 필터링하여 확인할 수도 있습니다.

① 히트맵에 대한 자세한 내용은 다음 문서를 참조하세요.

- [Java](#)
- [Node.js](#)
- [Python](#)

멀티서비스 히트맵

홈 화면 > 프로젝트 선택 > 분석 > 멀티서비스 히트맵



여러 서비스를 대상으로 히트맵 차트를 확인하고 서비스별 상태를 비교 분석할 수 있습니다.

검색 조건은 다음과 같습니다.

항목	설명
시작일	검색날짜
시작시간	3일, 7일, 한 달 검색 시에는 사용할 수 없습니다

항목	설명
기간	<ul style="list-style-type: none"> • 5분/10분/30분 • 1시간/3시간/24시간 • 3일/7일/한 달
구분	에이전트 종류
애플리케이션	애플리케이션

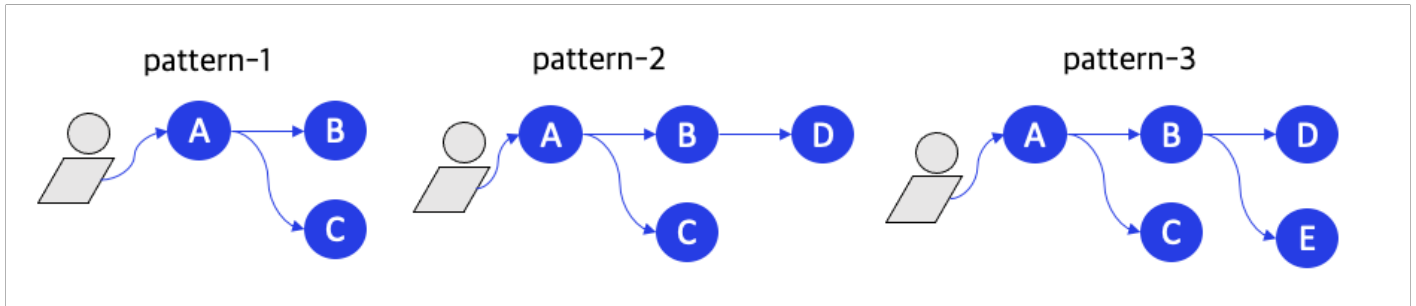
❗ 히트맵에 대한 자세한 내용은 다음 문서를 참조하세요.

- [Java](#)
- [Node.js](#)
- [Python](#)

MSA 분석

홈 화면 > 프로젝트 선택 > 분석 > MSA 분석

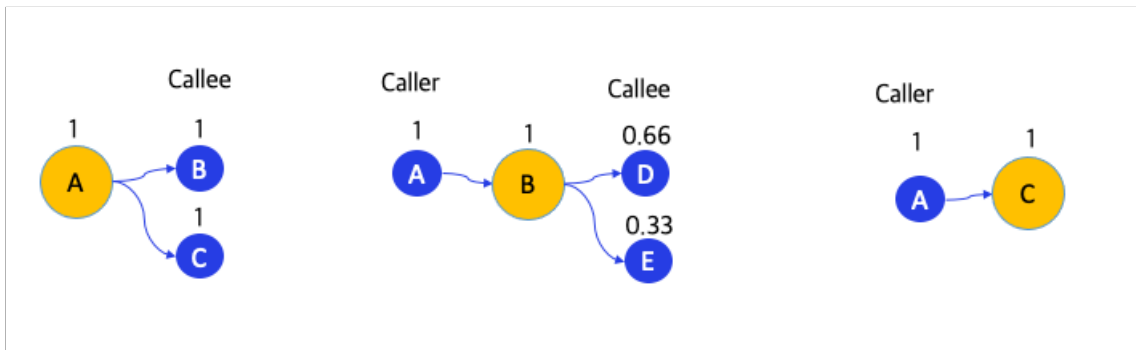
MSA 분석 기능은 와탭의 **멀티 트랜잭션 추적** 기능을 통해서 수집된 트랜잭션들 사이의 호출 비중이 어떠한지를 분석합니다. 이를 기반으로 마이크로 서비스간 의존성을 인스턴스 중심이 아닌 URL 레벨에서 확인할 수 있습니다.



예를 들어 A,B,C,D,E 라는 5개 마이크로 서비스가 있는데, 사용자가 A를 호출하면 로직에 따라서 다음과 같은 3가지의 호출 패턴이 발생한다고 가정하겠습니다.

1. A→B,C 호출
2. A→B,C 호출 + 다시 B→D 호출
3. A→B,C 호출 + 다시 B→D,E 호출

이 세가지 패턴이 한번씩 수행되었다면 MSA 분석에서는 기준 URL에 따라 연관도 분석 데이터를 확인할 수 있습니다.



❗ 멀티 트랜잭션 추적과 관련한 자세한 내용은 [다음 문서](#)를 참조하세요.

MSA 트랜잭션 통계

다음은 Caller 또는 Callee를 가지고 있는 모든 트랜잭션 통계입니다.

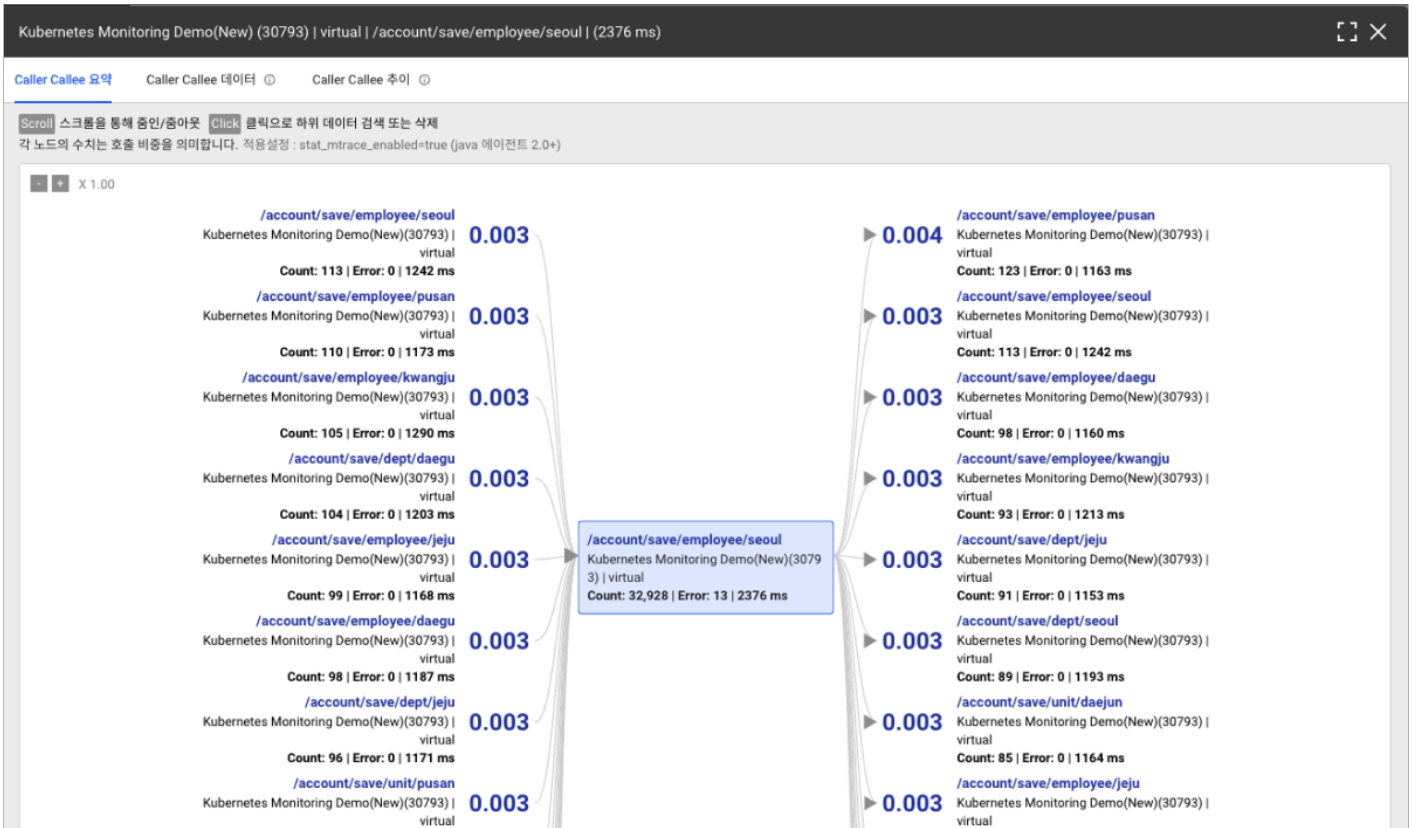
CP Kubernetes Monitoring Demo(EKS) ▾ MSA 분석									
시간 선택		필터		정렬 순서					
< 2023/10/10 00:00 ~ 2023/10/11 00:00 1일 >		에이전트 그룹 명 ▾		건수 ▾					
	상세	프로젝트	프로젝트 코드	에이전트 그룹 명	스펙	트랜잭션	건수	에러	평균 응답 시간
1		netes Monitoring Demc	33194	virtual		/account/save/employee/seoul	4,867	1	2,539
2		netes Monitoring Demc	33194	virtual		/account/save/employee/pusan	4,327	2	2,539
3		netes Monitoring Demc	33194	virtual		/account/save/employee/daegu	4,033	0	2,441
4		netes Monitoring Demc	33194	virtual		/account/save/employee/kwangju	3,849	2	2,388
5		netes Monitoring Demc	33194	virtual		/account/save/employee/daejun	3,663	0	2,422
6		netes Monitoring Demc	33194	virtual		/account/save/employee/jeju	3,640	1	2,511
7		netes Monitoring Demc	33194	virtual		/account/save/dept/seoul	3,462	1	2,453
8		netes Monitoring Demc	33194	virtual		/account/save/dept/pusan	3,434	1	2,731
9		netes Monitoring Demc	33194	virtual		/account/save/dept/daegu	3,430	1	2,531
10		netes Monitoring Demc	33194	virtual		/account/save/dept/kwangju	3,303	0	2,499
11		netes Monitoring Demc	33194	virtual		/account/save/dept/jeju	3,229	1	2,708
12		netes Monitoring Demc	33194	virtual		/account/save/division/kwangju	3,109	0	2,306
13		netes Monitoring Demc	33194	virtual		/account/save/division/seoul	3,107	1	2,577
14		netes Monitoring Demc	33194	virtual		/account/save/division/daegu	3,104	2	2,685
15		netes Monitoring Demc	33194	virtual		/account/save/division/pusan	3,094	1	2,548
16		netes Monitoring Demc	33194	virtual		/account/save/dept/daejun	3,075	0	2,325
17		netes Monitoring Demc	33194	virtual		/account/save/division/jeju	2,984	0	2,415
18		netes Monitoring Demc	33194	virtual		/account/save/division/daejun	2,969	1	2,409
19		netes Monitoring Demc	33194	virtual		/account/save/unit/seoul	2,954	1	3,164
20		netes Monitoring Demc	33194	virtual		/account/save/unit/pusan	2,947	0	2,349

위 통계 데이터 중 조회를 원하는 URL의 상세 아이콘을 선택하면 상세 보기 창이 나타납니다. 상세 보기 창에서 해당 트랜잭션에 대한 Caller Callee 요약, Caller Callee 데이터, Caller Callee 추이 등의 상세 데이터 조회가 가능합니다.

아이콘을 선택하면 상세 보기 창을 전체 화면으로 볼 수 있습니다.

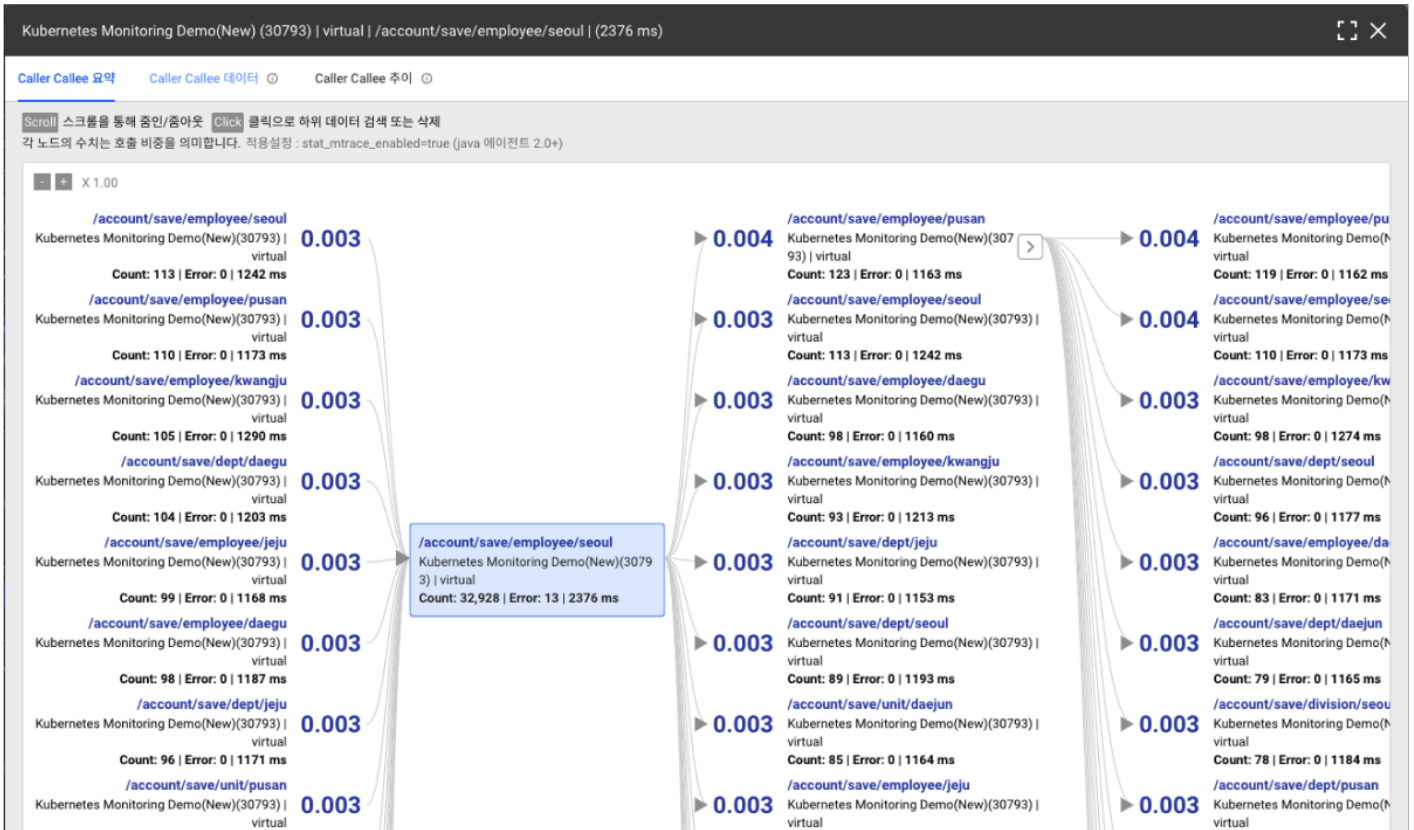
Caller Callee 요약

Caller Callee 요약 탭은 다음과 같이 해당 URL의 Caller, Callee 사이의 의존성 정보를 표시합니다. 이때 최대 0.004, 0.003 등 최대 1인 숫자 정보가 함께 표시되는데, 이것은 전체 Caller들 가운데 차지하는 비중을 표시합니다. 마찬가지로 각 Callee들의 비중이 오른쪽에 함께 표시됩니다.



위 화면을 통해 `/account/save/employee/seoul` 이라는 트랜잭션을 기준으로 Caller와 Callee들 사이의 의존성을 분석할 수 있습니다.

Caller나 Callee 노드는 다음과 같이 하위 노드로 상세 전개가 가능합니다.



Caller & Callee 데이터

기준 URL(예: /account/save/employee/seoul)에 대한 Caller와 Callee들의 상관관계는 다음과 같이 표 형태로 표시할 수 있습니다.

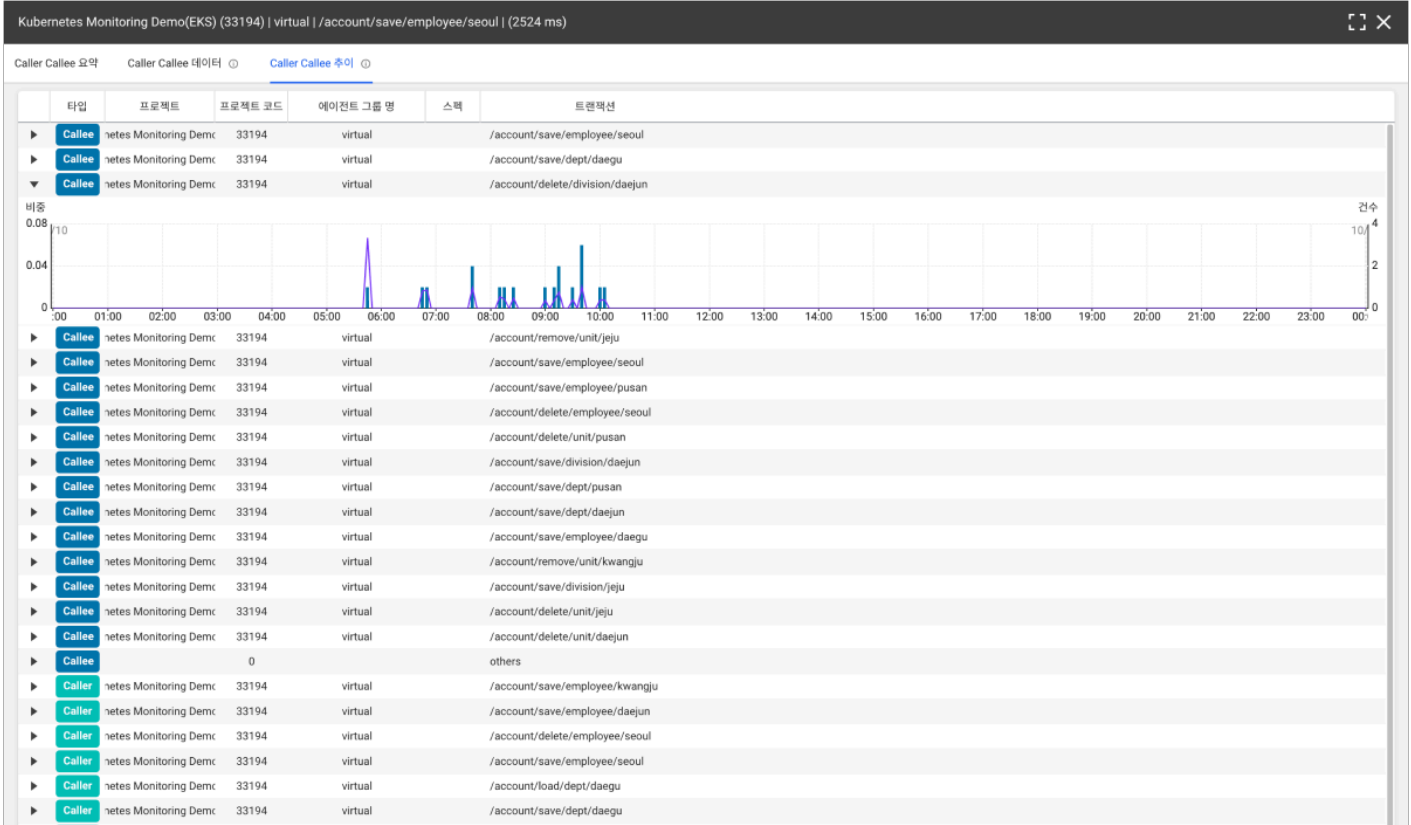
Kubernetes Monitoring Demo(EKS) (33194) | virtual | /account/save/employee/seoul | (2534 ms)

Caller Callee 요약 [Caller Callee 데이터](#) [Caller Callee 추이](#) [CSV 데이터 다운로드](#)

아이디	타입	프로젝트 코드	프로젝트	에이전트 그룹 명	트랜잭션	깊이	건수	에러	평균 응답 시간	비중	Parent ID
▼ id-0-1485863373		33194	Kubernetes Mon	virtual	/account/save/employee/seoul	0	5108	1	2534		
▶ id-caller1-541458444	Caller	33194	Kubernetes Mon	virtual	/account/save/employee/kwangju	1	19	0	1236	0.0037	id-0-1485...
▶ id-caller1-806644592	Caller	33194	Kubernetes Mon	virtual	/account/save/employee/daejun	1	19	0	1211	0.0037	id-0-1485...
▶ id-caller1-517953067	Caller	33194	Kubernetes Mon	virtual	/account/delete/employee/seoul	1	18	0	1233	0.0035	id-0-1485...
▶ id-caller1-1485863373	Caller	33194	Kubernetes Mon	virtual	/account/save/employee/seoul	1	18	0	1149	0.0035	id-0-1485...
▶ id-caller1-1166726544	Caller	33194	Kubernetes Mon	virtual	/account/save/dept/daegu	1	16	0	1185	0.0031	id-0-1485...
▶ id-caller1-869650383	Caller	33194	Kubernetes Mon	virtual	/account/load/dept/daegu	1	15	0	1170	0.0029	id-0-1485...
▶ id-caller1-552405637	Caller	33194	Kubernetes Mon	virtual	/account/save/unit/seoul	1	15	0	1201	0.0029	id-0-1485...
▶ id-caller1-1816628101	Caller	33194	Kubernetes Mon	virtual	/account/save/unit/daegu	1	15	0	1180	0.0029	id-0-1485...
▶ id-caller1-1172305335	Caller	33194	Kubernetes Mon	virtual	/account/kill/employee/kwangju	1	14	0	1184	0.0027	id-0-1485...
▶ id-caller1-486287746	Caller	33194	Kubernetes Mon	virtual	/account/save/division/daejun	1	14	0	1172	0.0027	id-0-1485...
▶ id-caller1-339409101	Caller	33194	Kubernetes Mon	virtual	/account/save/employee/daegu	1	14	0	1168	0.0027	id-0-1485...
▶ id-caller1-493967926	Caller	33194	Kubernetes Mon	virtual	/account/read/employee/seoul	1	14	0	1195	0.0027	id-0-1485...
▶ id-caller1-1302490095	Caller	33194	Kubernetes Mon	virtual	/account/remove/employee/seoul	1	14	0	1183	0.0027	id-0-1485...
▶ id-caller1-1144242220	Caller	33194	Kubernetes Mon	virtual	/account/remove/dept/pusan	1	13	0	1141	0.0025	id-0-1485...
▶ id-caller1-1820299183	Caller	33194	Kubernetes Mon	virtual	/account/delete/dept/kwangju	1	13	0	1183	0.0025	id-0-1485...
id-caller1-others	Caller	0		others		1	4165	0	1195	0.8154	id-0-1485...
▶ id-callee1-1485863373	Callee	33194	Kubernetes Mon	virtual	/account/save/employee/seoul	1	18	0	1149	0.0035	id-0-1485...
▶ id-callee1-1166726544	Callee	33194	Kubernetes Mon	virtual	/account/save/dept/daegu	1	18	0	1187	0.0035	id-0-1485...
▶ id-callee1-254345476	Callee	33194	Kubernetes Mon	virtual	/account/delete/division/daejun	1	17	0	1192	0.0033	id-0-1485...
▶ id-callee1-1907829325	Callee	33194	Kubernetes Mon	virtual	/account/remove/unit/jeju	1	17	0	1267	0.0033	id-0-1485...
▶ id-callee1-517953067	Callee	33194	Kubernetes Mon	virtual	/account/delete/employee/seoul	1	16	0	1197	0.0031	id-0-1485...
▶ id-callee1-1699366673	Callee	33194	Kubernetes Mon	virtual	/account/save/employee/pusan	1	15	0	1245	0.0029	id-0-1485...
▶ id-callee1-486287746	Callee	33194	Kubernetes Mon	virtual	/account/save/division/daejun	1	15	0	1175	0.0029	id-0-1485...
▶ id-callee1-703921046	Callee	33194	Kubernetes Mon	virtual	/account/delete/unit/pusan	1	15	0	1155	0.0029	id-0-1485...

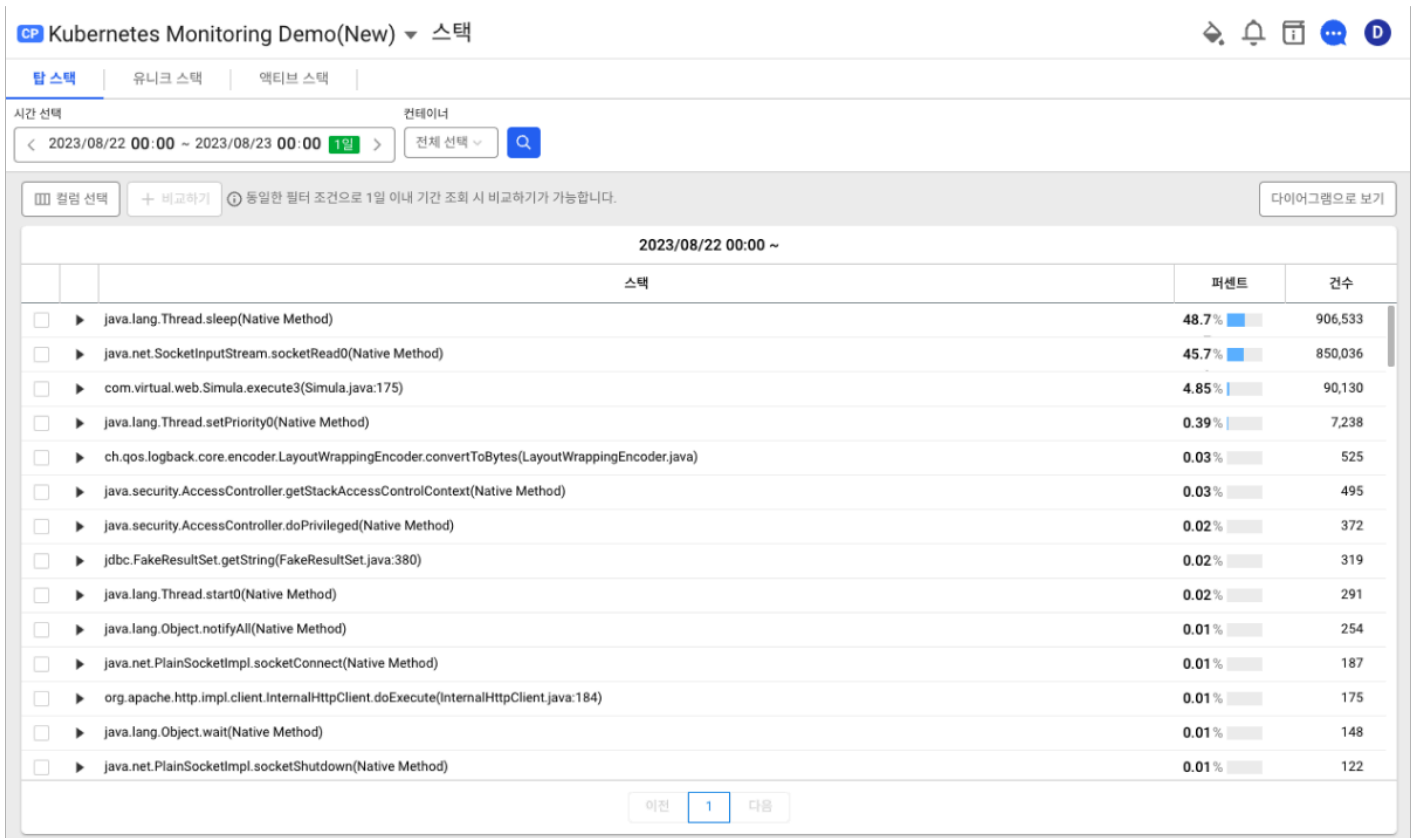
Caller Callee 추이

기준 URL(/account/save/employee/seoul)과 하나의 Caller나 Callee와의 호출 추이는 다음과 같이 시계열로 분석할 수 있습니다.



애플리케이션 스택

홈 화면 > 프로젝트 선택 > 분석 > 스택



와탭은 10초(기본값) 간격으로 수집한 스레드 스택을 활용하여 메소드 레벨의 성능 지연 구간을 분석합니다. 이 기능을 통해 각 컨테이너의 **탭 스택**, **유니크 스택**, **액티브 스택**을 확인할 수 있습니다.

ⓘ **탭 스택**, **유니크 스택**, **액티브 스택**에 대한 자세한 내용은 다음 문서를 참조하세요.

- [Java](#)
- [Python](#)

메트릭스

메트릭스란?

와탭은 모니터링 대상으로부터 데이터를 수집해서 사용자에게 제공합니다. 에이전트로부터 수집되는 데이터를 **메트릭스**라고 표기합니다.

메트릭스는 사용자 환경을 한 눈에 살펴볼 수 있는 기준 요소를 제공합니다. 예를 들어 서버별 메모리 사용률 평균, DB 평균 연결 시간 등을 원본 데이터 목록이나 시각화한 차트 뷰를 통해 간편하게 확인할 수 있습니다. 문제 요소를 찾은 후에는 로그와 트레이스 등을 통해 상세 분석을 확인할 수 있습니다.

메트릭스는 또한 사용자 환경의 스케일을 조절하는 것에도 도움을 줍니다. 자원 사용량 통계를 통해 필요 자원량을 확정하는 것은 성능 향상과 비용 효율성 측면에서 중요한 기준입니다.

와탭의 메트릭스 수집 방식



와탭 에이전트는 모니터링 대상으로부터 모니터링 지표를 수집해 메트릭 데이터의 형태로 와탭 수집 서버에 전송합니다. 와탭 수집 서버는 관련 데이터를 카테고리별로 저장하고 관리합니다.

와탭의 수집 서버는 다양한 모니터링 대상에서 메트릭스를 수집합니다. 사용자는 원하는 메트릭스에 접근하기 위해 해당 상품별 안내 화면으로 이동해 기술된 과정을 따라야 합니다.

예를 들어 [Kubernetes](#) 모니터링을 시작하고 싶다면 먼저 와탭 에이전트를 설치해야 합니다. [다음 문서](#)를 참조하세요. 관련 메트릭스 지표 안내 또한 [다음의 상품별 페이지](#)에서 확인할 수 있습니다.

와탭의 메트릭스 구성 요소

와탭의 **메트릭스**는 다음의 정보들로 구성되어 있습니다.

- **Category:** 관련된 지표들을 묶는 단위로 메트릭스를 구분하는 Key를 의미합니다.
- **Tags:** 수집 대상을 구분할 수 있는 고유 정보를 포함하는 데이터입니다. 변경이 드문 IP, Oname, Host 정보 등의 항목을 저장합니다. Map 형태로 Multi Tag가 존재합니다.
- **Fields:** 에이전트로부터 수집된 모든 지표 값을 저장합니다. Map 형태로 Multi Field가 존재합니다.
- **Time:** 메트릭스가 수집된 시간입니다.
- **Oid:** 메트릭스를 수집한 에이전트의 고유 번호입니다.
- **Oname:** 메트릭스를 수집한 에이전트의 명칭입니다.

메트릭스 데이터 조회 및 시각화

와탭은 사용자가 지정한 조건에 따라 수집한 원본 데이터 목록과 편의성을 위해 다양하게 시각화한 차트를 다음과 같이 제공합니다. 메트릭스의 원본 데이터를 조회할 수 있는 **메트릭스 조회**, 시각화한 차트를 통해 메트릭스 데이터를 조회할 수 있는 **메트릭스 차트**, 시가 학습한 메트릭스 지표의 패턴과 비교해 예상 패턴을 벗어난 이상을 탐지할 수 있는 **메트릭스 이상 탐지** 메뉴를 확인해 보세요.

The screenshot displays the Metrics tool interface with several panels:

- 메트릭스 조회 (Metrics Query):** A table listing metrics with columns for Time, Oid, Tags (alias, container, containerKey, host_ip, okindName, oname, onodeName, pid, type), and Fields (dbc, httpc, http, met).
- 메트릭스 이상 탐지 (Metrics Anomaly Detection):** A panel for detecting anomalies in metrics like ApplicationActiveStat, DbConnection, HttpCall, and Socket.
- 메트릭스 차트 (Metrics Chart):** A central area with an 'Overlay Chart' showing TPS (Transaction Per Second) and Think Time. The TPS chart shows a peak of 22.05, and the Think Time chart shows a peak of 92.87s.
- 인스턴스 응답 시간 (Instance Response Time):** A chart showing response times with a peak of 15.8s.
- 서비스 시간 (Service Time):** A chart showing service times with a peak of 57.82s.
- Call and Sql:** Additional charts showing call and SQL activity patterns.

쿠버네티스 지표

쿠버네티스에서 수집되는 지표 일부는 서버 모니터링, 애플리케이션과 동일합니다.

컨테이너(container) 지표

`container` 카테고리는 컨테이너의 pod에 설정된 모든 사용자 정의 라벨들을 태그로 수집합니다.

- 대상: 클러스터 프로젝트, 네임스페이스 프로젝트
- 수집 간격: 5초
- 통계 데이터: 5분

Tags

태그명	설명	비고
agentOid	노드 에이전트 고유 ID	고유값
agentPcode	프로젝트 코드	고유값
command	실행 명령어	-
containerId	컨테이너 ID	고유값
containerKey	컨테이너 Key	-
created	컨테이너가 생성된 TimeStamp	-
image	컨테이너 이미지명	-
imageHash	이미지 해시 값	-
imageId	이미지 ID	-
k8s-app	Pod의 레이블 k8s-app에 대한 값	-

태그명	설명	비고
microOid	컨테이너에 설치된 와탭 APM 에이전트의 고유 ID	-
name	컨테이너 이름	-
namespace	컨테이너가 소속된 네임스페이스	-
namespaceHash	컨테이너가 소속된 네임스페이스 해시 값	-
okind	컨테이너에 설치된 와탭 APM 에이전트에 지정한 OKIND 고유 ID	-
okindName	컨테이너에 설치된 와탭 APM 에이전트에 지정한 OKIND 이름	-
oname	컨테이너에 설치된 와탭 APM 에이전트 이름	-
onode	컨테이너가 작동 중인 노드 에이전트의 고유 ID	-
onodeName	컨테이너가 작동 중인 노드 이름	-
podHash	컨테이너의 Pod 해시 값	-
podName	컨테이너의 Pod 이름	-
replicaSetHash	컨테이너의 레플리카셋 해시 값	-
replicaSetName	컨테이너의 레플리카셋 이름	-
whatap_project	컨테이너가 속한 와탭 프로젝트 이름	-

Fields

필드명	단위	설명 (Shortname, Name, Description)
blkio_rbps	바이트	IoReadBytes

필드명	단위	설명 (Shortname, Name, Description)
		Container Block I/O Read Byte
		컨테이너 전체 블록 디바이스의 초당 읽은 바이트 합
blkio_riops	건수	IoReadIops
		Container Block I/O Read IOPS
		컨테이너 전체 블록 디바이스의 초당 읽은 건수 합
blkio_wbps	바이트	IoWriteBytes
		Container Block I/O Write Byte
		컨테이너 전체 블록 디바이스의 초당 쓴 바이트 합
blkio_wiops	건수	IoWriteIops
		Container Block I/O Write IOPS
		컨테이너 전체 블록 디바이스의 초당 쓴 건수 합
cpu_per_quota	퍼센트	CpuByLimit
		Container CPU Usage by Limit (%)
		CPU Limit 기준 CPU 전체 사용률
cpu_quota	밀리 코어	CpuLimit
		Container CPU Limit (core)
		컨테이너 CPU Limit 할당량 Limit 미설정인 경우 컨테이너가 작동 중인 해당 노드의 CPU 전체 코어가 밀리코어 단위로 표시된다.

필드명	단위	설명 (Shortname, Name, Description)
cpu_quota_percent	퍼센트	CpuLimitByNode
		Container CPU Limit by Node (%)
		노드 CPU 대비 컨테이너 CPU Limit 할당량 Limit 미설정인 경우 컨테이너가 작동 중인 해당 노드의 CPU 전체 코어가 퍼센트로 표시된다.
cpu_sys	퍼센트	CpuSysByNode
		Container CPU Sys Usage by Node (%)
		노드 CPU 대비 컨테이너 CPU System 사용률
cpu_throttledperiods	건수	CpuThrottledCnt
		Container CPU Throttling Count
		컨테이너 CPU Throttled 건수
cpu_throttledtime	나노세컨드	CpuThrottledTime
		Container CPU Throttling Time
		컨테이너 CPU Throttled 시간
cpu_total	퍼센트	CpuByNode
		Container CPU Usage by Node (%)
		노드 CPU 대비 컨테이너 CPU 사용률
cpu_total_milli	밀리코어	CpuTotUsage
		Container CPU Usage (millicore)

필드명	단위	설명 (Shortname, Name, Description)
		컨테이너 CPU 사용량
cpu_user	퍼센트	CpuUserByNode
		Container CPU User Usage by Node (%)
		노드 CPU 대비 컨테이너 CPU User 사용률
cpu_request	밀리코어	CpuRequest
		Container CPU Request (core)
		컨테이너 CPU 요청
cpu_per_request	퍼센트	CpuByRequest
		Container CPU Usage by Request (%)
		컨테이너 CPU 요청 대비 사용률 = $\text{cpu_total_milli} / \text{cpu_request} * 100$
mem_failcnt	건수	MemFailCnt
		Container Memory Failure Count
		컨테이너 메모리 Limit 도달 건수
mem_limit	바이트	MemLimit
		Container Memory Limit (byte)
		컨테이너 메모리 Limit 크기
mem_maxusage	바이트	MemMaxUsage
		Container Memory Max Usage (byte)

필드명	단위	설명 (Shortname, Name, Description)
		컨테이너 메모리 최대 사용량 기록 값
mem_percent	퍼센트	MemWsByLimit
		Container Memory Working Set by Limit (%)
		컨테이너 메모리 Limit 기준 working set 사용량 = mem_working_set / mem_limit * 100
mem_totalcache	바이트	MemTotCache
		Container Memory Total Cache (byte)
		컨테이너 전체 캐시 크기
mem_totalpgfault	횟수	MemTotPageFaultCnt
		Container Memory Total Page Fault Count
		컨테이너 Page Fault 횟수
mem_totalrss	바이트	MemTotRss
		Container Memory Total RSS (byte)
		컨테이너 RSS 메모리 전체 크기
mem_totalrss_percent	퍼센트	MemTotRssByLimit
		Container Memory Total RSS by Limit (%)
		컨테이너 RSS 메모리 전체 사용률
mem_totalunevictable	바이트	MemTotUnevictable
		Container Memory Total Unevictable (byte)

필드명	단위	설명 (Shortname, Name, Description)
		컨테이너 Unevictable Memory 전체 크기
mem_usage	바이트	MemUsage
		Container Memory Usage (byte)
		컨테이너 메모리 사용량
mem_working_set	바이트	MemWs
		Container Memory Working Set (byte)
		컨테이너 메모리 working set = mem_usage - inactive file
mem_working_set_percent	퍼센트	MemWsByLimit
		Container Memory Working Set by Limit (%)
		컨테이너 메모리 Limit 기준 working set 사용량 = mem_working_set / mem_limit * 100
mem_request	바이트	MemRequest
		Container Memory Request (byte)
		컨테이너 메모리 Request 크기
mem_per_request	퍼센트	MemWsByRequest
		Container Memory Working Set by Request (%)
		컨테이너 메모리 Request 기준 working set 사용량 = mem_working_set / mem_request * 100
network_rbps	바이트	NetRxBytes

필드명	단위	설명 (Shortname, Name, Description)
		Container Network Receive Byte
		컨테이너 전체 블럭 디바이스의 초당 읽기 바이트 합
network_rdropped	바이트	NetRxDropped
		Container Network Receive Dropped
		컨테이너 네트워크 수신 dropped 건수
network_rerror	바이트	NetRxError
		Container Network Receive Error
		컨테이너 네트워크 수신 에러 건수
network_riops	바이트	NetRxIops
		Container Network Receive IOPS
		컨테이너 네트워크 수신 건수
network_wbps	바이트	NetTxBytes
		Container Network Transmit Byte
		컨테이너 네트워크 송신 데이터 크기
network_wdropped	건수	NetTxDropped
		Container Network Transmit Dropped
		컨테이너 네트워크 송신 dropped 건수
network_werror	건수	NetTxError

필드명	단위	설명 (Shortname, Name, Description)
		Container Network Transmit Error
		컨테이너 네트워크 송신 에러 건수
network_wiops	건수	NetTxlops
		Container Network Transmit IOPS
		컨테이너 네트워크 송신 건수
node_cpu	퍼센트	ConNodeCpu
		Container Work Node CPU Usage (%)
		컨테이너가 실행 중인 노드의 CPU 사용량
node_mem	퍼센트	ConNodeMem
		Container Work Node Memory Usage (%)
		컨테이너가 실행 중인 노드의 메모리 사용량
phase	문자열	Pod 라이프사이클 ① PENDING ② RUNNING ③ SUCCEEDED ④ FAILED ⑤ UNKNOWN
restart_count	정수	ConRestartCnt
		Container Restart Count
		컨테이너 다시 시작 횟수
state	정수	ConState

필드명	단위	설명 (Shortname, Name, Description)
		Container Current State
		컨테이너 상태 코드 ① RUNNING = 114 ② PAUSE = 112 ③ RESTARTING = 101 ④ OOMKILLED = 111M ⑤ DEAD = 100 ⑥ WAITING = 119
status	문자열	ConStatus
		Container Current Status
		컨테이너 상태 정보 ① running 상태: uptime 정보 표시 ② waiting/terminated 상태: 상태에 대한 reason 정보 표시

쿠버네티스 노드(kube_node) 지표

`kube_node` 카테고리는 노드에 설정된 모든 사용자 정의 라벨들을 태그로 수집합니다.

- 대상: 클러스터 프로젝트, 네임스페이스 프로젝트
- 수집 간격: 5초
- 통계 데이터: 5분, 1시간

Tags

태그명	설명	비고
nodeName	노드 이름	-

Fields

필드명	단위	설명	비고
allocatable_cpu	밀리 코어	노드 할당 가능한 CPU량	-
allocatable_memory	바이트	노드 할당 가능한 메모리량	-
allocatable_pods	정수	노드 할당 가능한 Pod 수	-
limit_cpu	밀리 코어	노드 CPU Limit 합계	-
limit_memory	바이트	노드 메모리 Limit 합계	-
pods	정수	노드 Pod 총개수	-
request_cpu	밀리 코어	노드 CPU Request 합계	-
request_memory	바이트	노드 메모리 Request 합계	-

쿠버네티스 이벤트(kube_event) 지표

`kube_event` 카테고리는 클러스터 프로젝트의 경우 클러스터 전체를 대상으로 데이터를 수집하며 네임스페이스 프로젝트는 해당 네임스페이스에서 발생한 이벤트에 대해서만 수집합니다.

- 대상: 클러스터 프로젝트, 네임스페이스 프로젝트
- 수집 간격: 5초
- 통계 데이터: 5분, 1시간

Tags

태그명	설명	비고
field_path	Field Path	-

태그명	설명	비고
kind	종류	이벤트가 발생한 오브젝트 종류
name	오브젝트 이름	이벤트가 발생한 쿠버네티스 오브젝트 이름
namespace	네임스페이스 이름	이벤트가 발생한 네임스페이스
reason	이벤트 발생 사유	-
type	이벤트 유형	Warning 또는 Normal
uid	UID	이벤트가 발생한 오브젝트

Fields

필드명	단위	설명	비고
action	문자열	액션 이름	-
count	건수	이벤트 발생 건수	-
event_time	정수	이벤트 최초 발생 TimeStamp	-
first_timestamp	정수	이벤트 최초 발생 시간	-
last_timestamp	정수	이벤트 마지막 발생 시간	-
message	문자열	이벤트 메시지	-
reasonFiled	문자열	이벤트 Reason	-
reporting_component	문자열	현재 이벤트를 보고하는 컴포넌트	-
reporting_instance	문자열	현재 이벤트를 보고하는 인스턴스	-

필드명	단위	설명	비고
series_last_observed_time	정수	series last observed time	-

쿠버네티스 클러스터(kube_stat) 지표

kube_stat 카테고리는 클러스터 프로젝트의 경우 클러스터 전체 대상으로 수집하고 네임스페이스 프로젝트는 해당 네임스페이스와 연관된 오브젝트를 대상으로만 수집합니다.

- 대상: 클러스터 프로젝트, 네임스페이스 프로젝트
- 수집 간격: 5초
- 통계 데이터: 5분, 1시간

Tags

태그명	설명	비고
name	kube_stat	고정 값

Fields

필드명	단위	설명	비고
alloctable_cpu	밀리 코어	클러스터 전체 코어 수	클러스터 프로젝트 Only
alloctable_ephemeral-storage	바이트	클러스터 전체 할당 가능한 ephemeral storage	클러스터 프로젝트 Only
alloctable_hugepages-1gi	바이트	클러스터 전체 할당 가능한 hugepages-1Gi	클러스터 프로젝트 Only

필드명	단위	설명	비고
allocatable_hugepages-2mi	바이트	클러스터 전체 할당 가능한 hugepages-2Mi	클러스터 프로젝트 Only
allocatable_memory	바이트	클러스터 전체 할당 가능한 메모리	클러스터 프로젝트 Only
allocatable_pods	정수	할당 가능한 Pod 수	-
available_pod	정수	Phase가 Running 상태인 Pod의 수	-
desired_pod	정수	metadata.ownerReferences 없이 배포된 Pod 수와 쿠버네티스 오브젝트(ReplicaSet, Daemonset, StatefulSet)에 정의된 Desired Pod 수의 합	-
		kubectl get pods -A로 조회한 Pod 수와 동일	-
nodes	정수	노드 수	-
pod_phase_Pending	정수	Pending 상태 Pod 수	-
pod_phase_Running	정수	Running 상태 Pod 수	-
running_containers	정수	Running 컨테이너 수	-
stopped_containers	정수	Stopped 컨테이너 수	-
total_available_cpu	정수	할당 가능한 CPU 총합	-
total_available_memory	정수	할당 가능한 Memory 총합	-
total_limit_cpu	밀리 코어	Limit CPU 총합	-

필드명	단위	설명	비고
total_limit_memory	바이트	Limit Memory 총합	-
total_request_cpu	밀리 코어	Request CPU 총합	-
total_request_memory	바이트	Request Memory 총합	-
unavailable_pod	정수	Phase가 Running 상태가 아닌(Pending, Failed, Succeeded) Pod 수	-
waiting_containers	정수	Waiting Container 수	-

Pod(kube_pod) 지표

`kube_pod` 카테고리는 Pod에 설정된 모든 사용자 정의 라벨들을 태그로 수집합니다.

- 대상: 마스터(클러스터) 프로젝트, 네임스페이스 프로젝트
- 수집 간격: 5초
- 통계 데이터: 5분

Tags

태그명	설명	비고
agentOid	노드 에이전트 고유 ID	고유값
agentPcode	프로젝트 코드	고유값
command	실행 명령어	-
containerIds	Pod 에 속해있는 컨테이너 ID	-
containerIdsCount	containerIds 의 개수	-

태그명	설명	비고
containerKeys	Pod 에 속해있는 컨테이너 ID 의 해시 값	-
containerKeysCount	containerKeys 의 개수	-
DaemonSet	Pod의 DaemonSet 이름	-
Deployment	Deployment	-
k8s-app	Pod의 레이블 k8s-app에 대한 값	-
microOid	Pod 의 컨테이너 내부 애플리케이션에서 실행중인 에이전트의 id	-
microOids	Pod 안의 복수개 컨테이너 내부 애플리케이션에서 실행 중인 에이전트의 복수개 id	-
microOidsCount	microOids 의 개수	-
name	Pod 이름	-
onames	Pod 의 컨테이너 내부 애플리케이션에서 실행중인 에이전트의 명칭	-
onamesCount	onames 의 개수	-
podName	Pod 이름	-
namespace	Pod가 소속된 네임스페이스	-
namespaceHash	Pod가 소속된 네임스페이스 해시 값	-
replicaSetHash	Pod의 ReplicaSet 해시 값	-
replicaSetName	Pod의 ReplicaSet 이름	-
whatap_project	Pod가 속한 와탭 프로젝트 이름	-

Fields

필드명	단위	(Shortname, Name, Description)
blkio_rbps	바이트	IoReadBytes
		Pod Block I/O Read Byte
		Pod 전체 블럭 디바이스의 초당 읽은 바이트 합
blkio_riops	건수	IoReadIops
		Pod Block I/O Read IOPS
		Pod 전체 블럭 디바이스의 초당 읽은 건수 합
blkio_wbps	바이트	IoWriteBytes
		Pod Block I/O Write Byte
		Pod 전체 블럭 디바이스의 초당 쓴 바이트 합
blkio_wiops	건수	IoWriteIops
		Pod Block I/O Write IOPS
		Pod 전체 블럭 디바이스의 초당 쓴 건수 합
cpu_per_limit	퍼센트	CpuByLimit
		Pod CPU Usage by Limit (%)
		CPU Limit 기준 CPU 전체 사용률
cpu_per_request	퍼센트	CpuByRequest
		Pod CPU Usage by Limit (%)

필드명	단위	(Shortname, Name, Description)
		CPU Limit 기준 CPU 전체 사용률
cpu_quota_percent	퍼센트	CpuLimitByNode
		Pod CPU Limit by Node (%)
		노드 CPU 대비 Pod CPU Limit 할당량 Limit 미설정인 경우 Pod가 작동 중인 해당 노드의 CPU 전체 코어가 퍼센트로 표시된다.
cpu_sys	퍼센트	CpuSysByNode
		Pod CPU Sys Usage by Node (%)
		노드 CPU 대비 Pod CPU System 사용률
cpu_throttledperiods	건수	CpuThrottledCnt
		Pod CPU Throttling Count
		Pod CPU Throttled 건수
cpu_throttledtime	나노세컨드	CpuThrottledTime
		Pod CPU Throttling Time
		Pod CPU Throttled 시간
cpu_total	퍼센트	CpuByNode
		Pod CPU Usage by Node (%)
		노드 CPU 대비 Pod CPU 사용률
cpu_total_milli	밀리코어	CpuTotUsage

필드명	단위	(Shortname, Name, Description)
		Pod CPU Usage (millicore)
		Pod CPU 사용률
cpu_user	퍼센트	CpuUserByNode
		Pod CPU User Usage by Node (%)
		노드 CPU 대비 Pod CPU User 사용률
cpu_request	밀리코어	CpuRequest
		Pod CPU Request (core)
		Pod CPU 요청
cpu_per_request	퍼센트	CpuByRequest
		Pod CPU Usage by Request (%)
		Pod CPU 요청 대비 사용률 = $\text{cpu_total_milli} / \text{cpu_request} * 100$
mem_totalcache	바이트	MemTotCache
		Pod Memory Total Cache (byte)
		Pod 전체 캐시 크기
mem_totalpgfault	횟수	MemTotPageFaultCnt
		Pod Memory Total Page Fault Count
		Pod Page Fault 횟수
mem_totalrss	바이트	MemTotRss

필드명	단위	(Shortname, Name, Description)
		Pod Memory Total RSS (byte)
		Pod RSS 메모리 전체 크기
mem_totalrss_percent	퍼센트	MemTotRssByLimit
		Pod Memory Total RSS by Limit (%)
		Pod RSS 메모리 전체 사용률
mem_totalunevictable	바이트	MemTotUnevictable
		Pod Memory Total Unevictable (byte)
		Pod Unevictable Memory 전체 크기
mem_usage	바이트	MemUsage
		Pod Memory Usage (byte)
		Pod 메모리 사용량
mem_working_set	바이트	MemWs
		Pod Memory Working Set (byte)
		Pod 메모리 working set = mem_usage - inactive file
memory_request	바이트	MemRequest
		Pod Memory Request (byte)
		Pod 메모리 요청량
memory_limit	바이트	MemLimit

필드명	단위	(Shortname, Name, Description)
		Pod Memory Limit (byte)
		Pod 메모리 Limit 할당량
memory_per_request	퍼센트	MemByRequest
		Pod Memory by Request (%)
		Pod 메모리 요청량 기준 사용량
memory_per_limit	퍼센트	MemByLimit
		Pod Memory by Limit (%)
		Pod 메모리 Limit 기준 사용량
network_rbps	바이트	NetRxBytes
		Pod Network Receive Byte
		Pod 전체 블럭 디바이스의 초당 읽기 바이트 합
network_rdropped	바이트	NetRxDropped
		Pod Network Receive Dropped
		Pod 네트워크 수신 dropped 건수
network_rerror	바이트	NetRxError
		Pod Network Receive Error
		Pod 네트워크 수신 에러 건수
network_riops	바이트	NetRxIops

필드명	단위	(Shortname, Name, Description)
		Pod Network Receive IOPS
		Pod 네트워크 수신 건수
network_wbps	바이트	NetTxBys
		Pod Network Transmit Byte
		Pod 네트워크 송신 데이터 크기
network_wdropped	건수	NetTxDropped
		Pod Network Transmit Dropped
		Pod 네트워크 송신 dropped 건수
network_werror	건수	NetTxError
		Pod Network Transmit Error
		Pod 네트워크 송신 에러 건수
network_wiops	건수	NetTxlops
		Pod Network Transmit IOPS
		Pod 네트워크 송신 건수
phase	문자열	Phase
		Pod Current Phase
		Pod 라이프사이클 ① PENDING ② RUNNING ③ SUCCEEDED

필드명	단위	(Shortname, Name, Description)
		④ FAILED ⑤ UNKNOWN

다음 필드는 내부용으로 예약되었습니다.

필드명	설명	비고
kube_sless_normal	쿠버네티스 정보성 이벤트 발생 건수	-
kube_sless_warning	쿠버네티스 경고 이벤트 발생 건수	-
micro_sful_critical	apm 심각 상태 기반 이벤트 발생 건수	-
micro_sful_info	apm 정보성 상태 기반 이벤트 발생 건수	-
micro_sful_warning	apm 경고 상태 기반 이벤트 발생 건수	-
micro_sless_critical	apm 심각 무상태 이벤트 발생 건수	-
micro_sless_info	apm 정보성 무상태 이벤트 발생 건수	-
micro_sless_warning	apm 경고 무상태 이벤트 발생 건수	-
sful_critical	메트릭스 심각 상태 기반 이벤트 발생 건수	-
sful_info	메트릭스 정보성 상태 기반 이벤트 발생 건수	-
sful_warning	메트릭스 경고 상태 기반 이벤트 발생 건수	-
sless_critical	메트릭스 심각 무상태 이벤트 발생 건수	-
sless_info	메트릭스 정보성 무상태 이벤트 발생 건수	-
sless_warning	메트릭스 경고 무상태 이벤트 발생 건수	-

쿠버네티스 Pod 통계(kube_pod_stat) 지표

kube_pod_stat 카테고리 클러스터 프로젝트는 클러스터 전체를 대상으로 데이터를 수집하며 네임스페이스 프로젝트는 해당 네임스페이스에 소속된 Pod에 대해서만 수집합니다.

- 대상: 클러스터 프로젝트, 네임스페이스 프로젝트
- 수집 간격: 5초
- 통계 데이터: 5분, 1시간

Tags

태그명	설명	비고
kind	유형	클러스터 프로젝트의 경우 cluster 고정 값을 가지며 네임스페이스 프로젝트의 경우 Deployment 또는 ReplicaSet에 대해서만 수집합니다.
name	쿠버네티스 자원 이름	클러스터 프로젝트는 name 값이 없으며 네임스페이스 프로젝트는 Deployment 또는 ReplicaSet의 이름입니다.

Fields

필드명	단위	설명	비고
available_pod	정수	Phase가 Running 상태인 Pod의 수	-
desired_pod	정수	metadata.ownerReferences 없이 배포된 Pod 수와 쿠버네티스 오브젝트(ReplicaSet, Daemonset, StatefulSet)에 정의된 Desired Pod 수의 합	-
		kubectl get pods -A로 조회한 Pod 수와 동일	-
limit_cpu	밀리 코어	CPU Limit 사용량	-
limit_memory	바이트	Memory Limit 사용량	-
request_cpu	밀리 코어	CPU Request 사용량	-

필드명	단위	설명	비고
request_memory	바이트	Memory Request 사용량	-
running_container	정수	Running Container 수	-
stopped_container	정수	Stopped Container 수	-
waiting_container	정수	Waiting Container 수	-

쿠버네티스 Horizontal Pod Autoscaler(HPA)(kube_hpa_stat) 지표

와탭에서 사용하는 ClusterRole에 HPA가 추가되어야 지표 수집이 시작됩니다.

- 대상: 클러스터 프로젝트
- 수집 간격: 5초
- 통계 데이터: 5분, 1시간

Tags

태그명	설명	비고
name	HPA 이름	-

Fields

필드명	단위	설명	비고
currentReplicas	정수	현재 레플리카 수	-
desiredReplicas	정수	Desired 레플리카 수	-
lastScaleTime	정수	마지막으로 스케일이 변경된 TimeStamp	-
maxReplicas	정수	최대 레플리카 수	-

필드명	단위	설명	비고
minReplicas	정수	최소 레플리카 수	-

메트릭스 조회

홈 화면 > 프로젝트 선택 > 분석 > 메트릭스 조회

메트릭스 조회 메뉴에서 태그 기반으로 특정 메트릭스를 조회할 수 있습니다.

The screenshot shows the '메트릭스 조회' (Metrics Search) interface. It is divided into three numbered sections:

- 1 시간 선택** (Time Selection): Includes a date range selector (2023/08/17 07:56 ~ 2023/08/17 08:56), a '최대 개수' (Max Count) dropdown set to 100, and a '카테고리' (Category) dropdown set to 'app_counter'.
- 2 태그 전체** (All Tags): A list of tags with 'active_tx_3' selected. A '필드 전체' (All Fields) section shows various metrics like 'arrival_rate', 'tx_error', etc.
- 3**: A table with columns for 'Time', 'Oid', 'Tags', and 'Fields'. The 'Tags' column lists attributes like 'container', 'host_ip', etc. The 'Fields' column lists metrics like 'active_tx_3', 'arrival_rate', etc.

	Time	Oid	Tags							Fields					
			container	containerKey	host_ip	okindName	oname	onodeName	pid	type	active_tx_3	arrival_rate	sql_time	tps	tx_error
1	2023-08-17 07:56:45	-877561626	agent.service	1395943798	10.21.1.26	demo-okind-1	demo-8101	node-1	3284967	java	11	24.390244	5.7834587	20.68982	1
2	2023-08-17 07:56:45	-1128904592	agent.service	1395943798	10.21.1.26	demo-okind-0	demo-8100	node-0	3284934	java	5	15.993603	9.613368	18.623732	0

1 시간과 카테고리 선택

1 영역에서 메트릭스가 수집된 시간과 최대 개수 및 카테고리를 지정할 수 있습니다. 시간과 카테고리는 반드시 지정해야 합니다.

- 시간**: 메트릭스가 수집된 시간을 지정해 조회할 수 있습니다. 기본값은 1시간 입니다. 기본 옵션으로 제공하는 조회 시간 외 사용자가 직접 시간 탭을 선택해 날짜와 시간을 지정할 수 있습니다.
- 최대 개수**: 3 목록에 조회할 메트릭스 최대 개수를 지정할 수 있습니다. 10, 50, 100, 200, 300, 1000, 2000, 3000 개까지 설정할 수 있습니다.
- 카테고리**: 유관 지표들의 분류 단위입니다. 카테고리 탭을 선택해 원하는 카테고리를 지정할 수 있습니다.
- 새로 고침**: 새로 고침 아이콘을 선택하면 카테고리, 태그 및 필드 옵션을 다시 불러올 수 있습니다.

2 태그와 필드 선택

2 영역에서 태그와 필드를 선택합니다. 사용자가 개별적으로 지정하지 않는다면 기본 설정은 전체 선택입니다.

- 태그**: 수집된 대상을 구분할 수 있는 고유 정보 데이터입니다.

- **필드**: 모니터링 대상으로부터 수집된 지표입니다.
- **필터**: ▾ **태그값으로 필터링** 버튼을 선택하고 태그 값을 설정해 필터링할 수 있습니다.
예시, `oname` 의 값을 `demo-8101` 로 설정해 필터링한 데이터를 조회할 수 있습니다.
- 🔍 **검색**: 조건을 설정 후 **검색** 아이콘을 선택하면 ③ 영역에서 해당 메트릭스의 원본 데이터를 조회할 수 있습니다.
- 📄 **CSV 다운로드**: **CSV 다운로드** 버튼을 선택해 해당 메트릭스 원본 데이터를 CSV 파일로 다운로드할 수 있습니다.

3 메트릭스 테이블

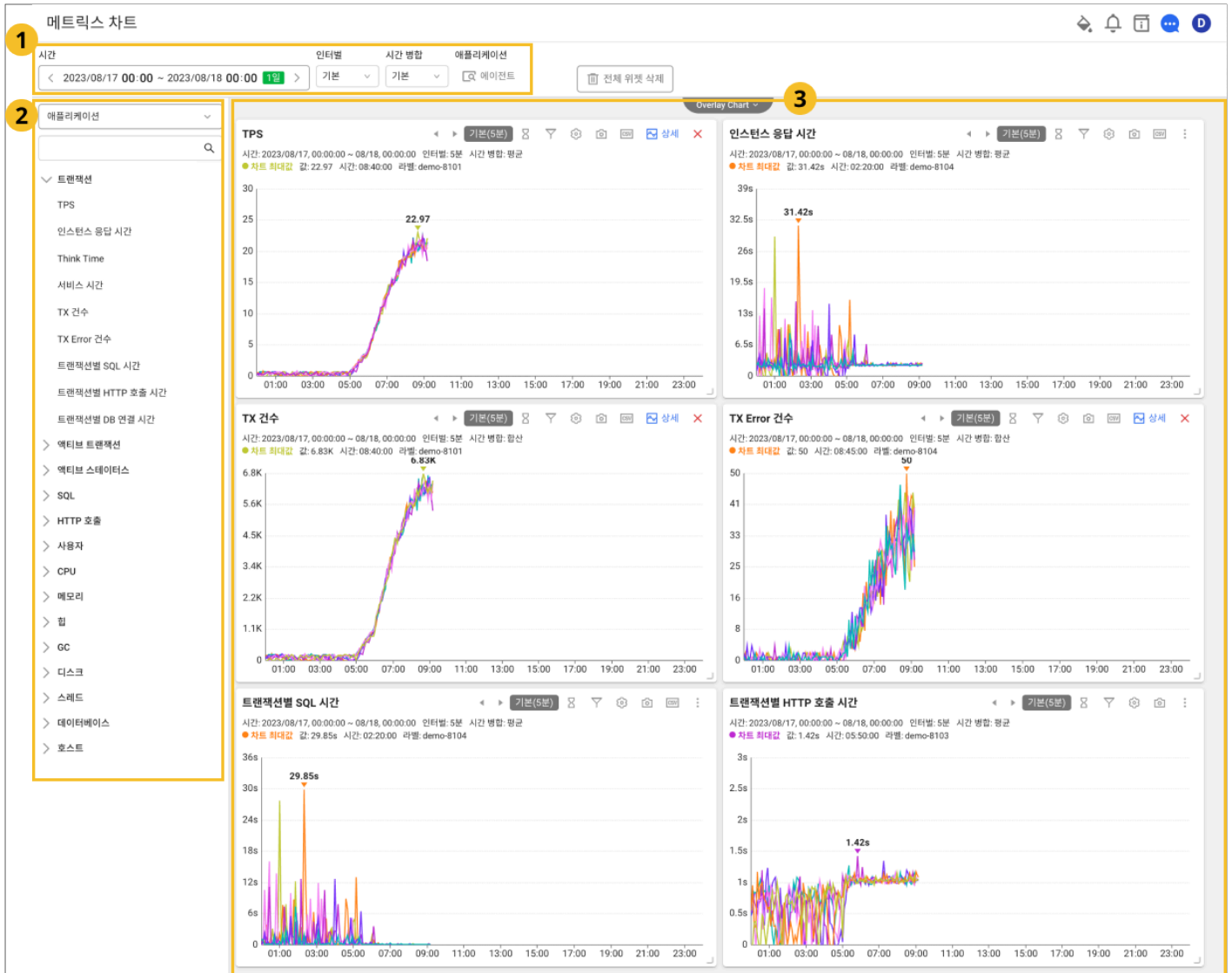
수집되는 메트릭스를 사전에 특정할 수 없기에 수집 중인 모든 메트릭스의 원본 데이터를 확인하는 것이 중요합니다. 위의 조건 영역에서 원하는 조건을 설정 후 ③ 영역에서 해당 메트릭스의 원본 데이터를 테이블 형식으로 조회할 수 있습니다. 사용자가 **태그**와 **필드** 각 조건을 지정함에 따라 테이블의 컬럼이 변경됩니다.

- ① • 메트릭스 조회 시 **시간**과 **카테고리**는 반드시 지정해야 합니다.
- 메트릭스 조회 시 **태그**와 **필드** 지정은 선택 사항입니다.

메트릭스 차트

홈 화면 > 프로젝트 선택 > 분석 > 메트릭스 차트

메트릭스 차트 메뉴에서 모니터링 대상에서 수집된 메트릭스 데이터를 다음과 같이 차트로 조회할 수 있습니다. 시간과 지표를 지정하는 것은 필수입니다.



상단 옵션

- 1 영역의 **메트릭스 차트**의 상단 옵션을 통해 차트의 시간 범위와 모니터링 대상 에이전트를 지정할 수 있습니다.
 - **시간**: 시간의 총 범위로 X축의 시작과 끝을 지정할 수 있습니다.
 - **인터벌**: 시간 간격으로 X축 데이터 간격을 지정할 수 있습니다.
 - **시간 병합**: 데이터 병합 방식 중 하나로 인터벌로 지정한 시간 내의 데이터를 병합할 수 있습니다.
 - 예, **평균**은 1시간 안 데이터의 평균값을 말합니다.
 - **에이전트**: 조회할 에이전트를 지정할 수 있습니다. 지정하지 않으면 전체가 조회됩니다.

지표 목록

- 2 영역은 옵션을 조회할 지표 목록입니다. 먼저 **카테고리**를 선택하세요. 선택한 **카테고리** 하위의 지표를 조회한 후 원하는 지표를 선택하세요. **카테고리**와 지표를 선택하면 1 영역의 상단 메뉴에서 지정한 시간 범위의 데이터를 바탕으로 3 영역에서 차트 위젯을 조회할 수 있습니다.

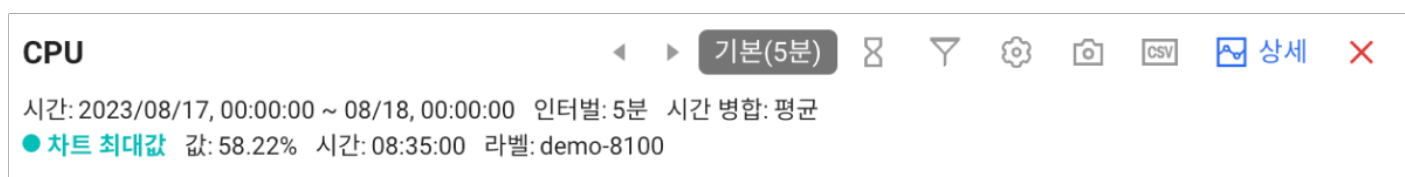
! 데이터 병합

데이터 병합은 **시간 병합**과 **오브젝트 병합**을 제공합니다.

- **시간 병합**은 원본 데이터에서 필드 값이 같은 데이터끼리 일정한 간격으로 데이터를 병합합니다.
- **오브젝트 병합**은 서로 다른 필드 값을 가진 데이터들 중에서 태그가 일치하는 경우 해당 데이터를 병합합니다.

차트 위젯

- 3 영역 차트 위젯의 좌측 상단에서 지표명을 확인할 수 있습니다. 차트 위젯의 우측 상단에서 다음과 같은 옵션을 확인할 수 있습니다.



- **시간 이동**: ◀ ▶ 왼쪽 화살표 또는 오른쪽 화살표 버튼을 통해 선택한 시간 범위만큼 -1, +1 씩 이동 가능합니다.

예, 시간 범위가 2월 13일 00:00~2월 14일 00:00일 때, ◀ 왼쪽 화살표를 선택하면 2월 12일 00:00~2월 13일 00:00 데이터를 조회할 수 있습니다.

- 인터벌/시간 병합: ① 상단 메뉴에서 지정한 인터벌과 시간 병합을 수정할 수 있습니다.
- 모니터링 대상: ▾ 아이콘을 선택해 모니터링 대상을 지정할 수 있습니다. 선택하지 않으면 전체를 대상으로 조회합니다.
- 시간 비교: ⏮ 아이콘을 선택하면 동일한 지표의 이전 시간대의 추이를 비교할 수 있습니다.
- 스냅샷: 📷 아이콘을 선택해 위젯의 옵션을 제외한 차트를 스냅샷 할 수 있습니다.
- CSV: 📄 아이콘을 선택해 차트를 그리는 데이터를 CSV 파일로 다운로드할 수 있습니다.
- 상세 보기: 🔍 아이콘을 선택해 상세 조회가 가능합니다. 모니터링 대상이 많은 경우 유용하며, 모니터링 대상의 지표 추이를 개별로 확인 할 수 있습니다.

ⓘ 메트릭스 차트 위젯 상단에서 옵션이 보이지 않을 경우 ⋮ 아이콘을 선택하세요.

위젯 삭제하기

화면에 배치한 모든 위젯을 삭제하려면 화면 위에 위치한 ☰ **전체 위젯 삭제** 버튼을 선택하세요.

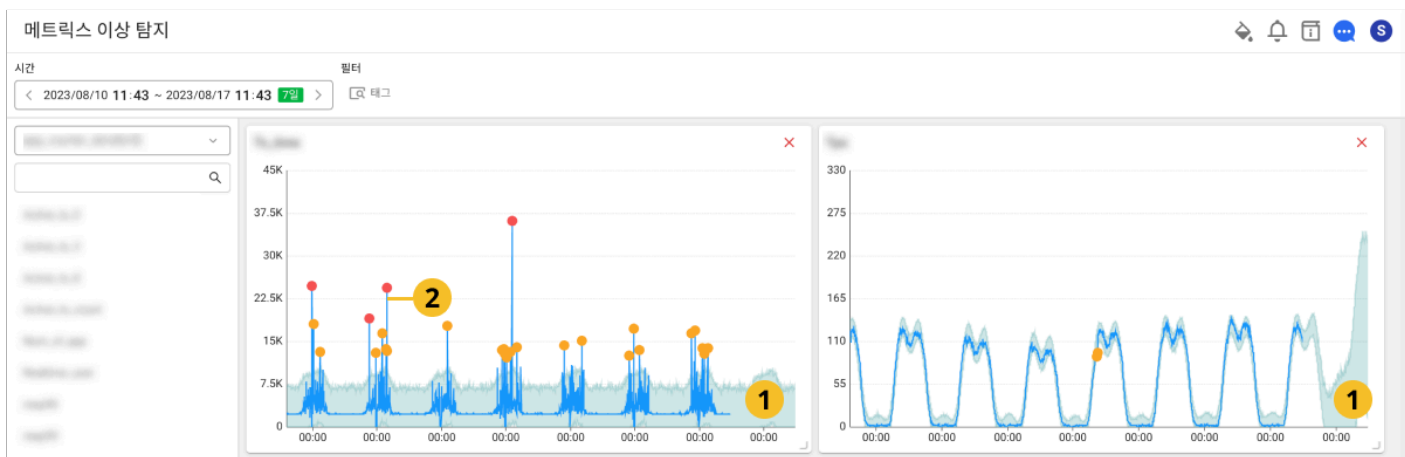
메트릭스 이상 탐지

홈 화면 > 프로젝트 선택 > 분석 > 메트릭스 이상 탐지

다양한 메트릭스 지표의 패턴을 시가 학습한 예상 패턴과 비교해 볼 수 있습니다. 예상 패턴을 벗어난 이상 탐지를 그래프 차트를 통해 확인할 수 있습니다. 과거 데이터를 바탕으로 반복되는 패턴을 확인하고 향후 지표 값 예측에 활용할 수 있습니다.

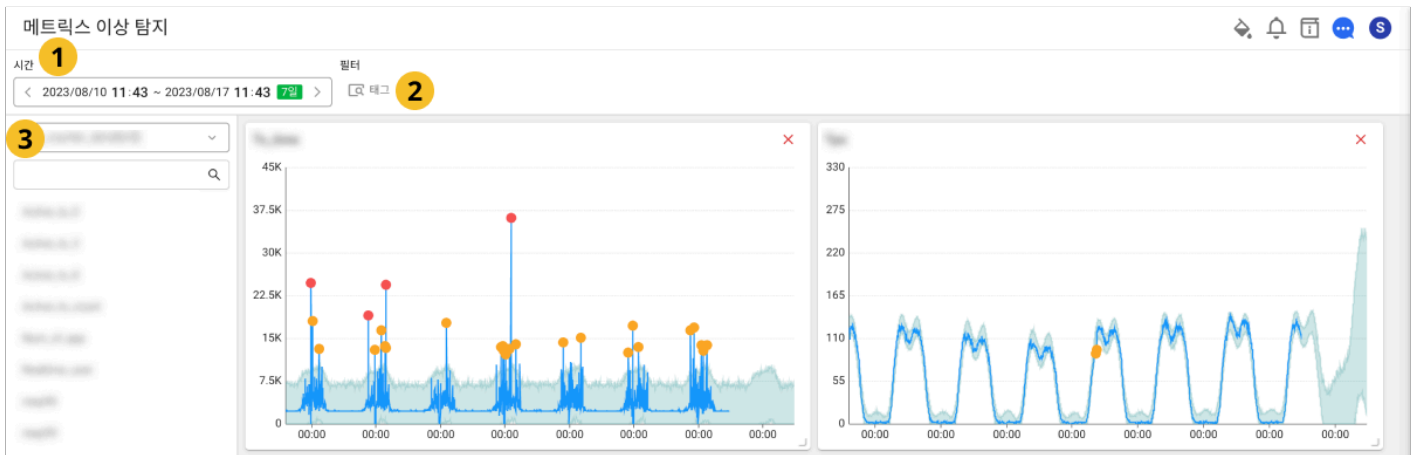
- ❗ • 화면에 배치한 위젯은 다른 메뉴로 이동할 경우 저장되지 않고 초기화합니다.
- 패턴 표시와 이상치 표시 기능을 제외하면 **분석 > 메트릭스 차트** 메뉴와 유사합니다.
- **이상치 탐지(Anomaly Detection)** 경고 알림 기능의 기술 근간은 **메트릭스 이상 탐지**입니다. **이상치 탐지** 경고 알림 설정에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

위젯 확인하기



- ❶ 밝은 색상의 그래프 영역은 시가 분석한 예상 패턴입니다.
- ❷ 파랑색 그래프는 프로젝트의 메트릭스 지표 추이입니다.
- 시가 분석한 예상 패턴을 벗어나면 **주황색**, **빨간색**의 단계로 그래프에 점을 표시합니다. 예상 패턴 범위를 크게 벗어난 값을 **빨간색**으로 표시합니다.

위젯 배치하기

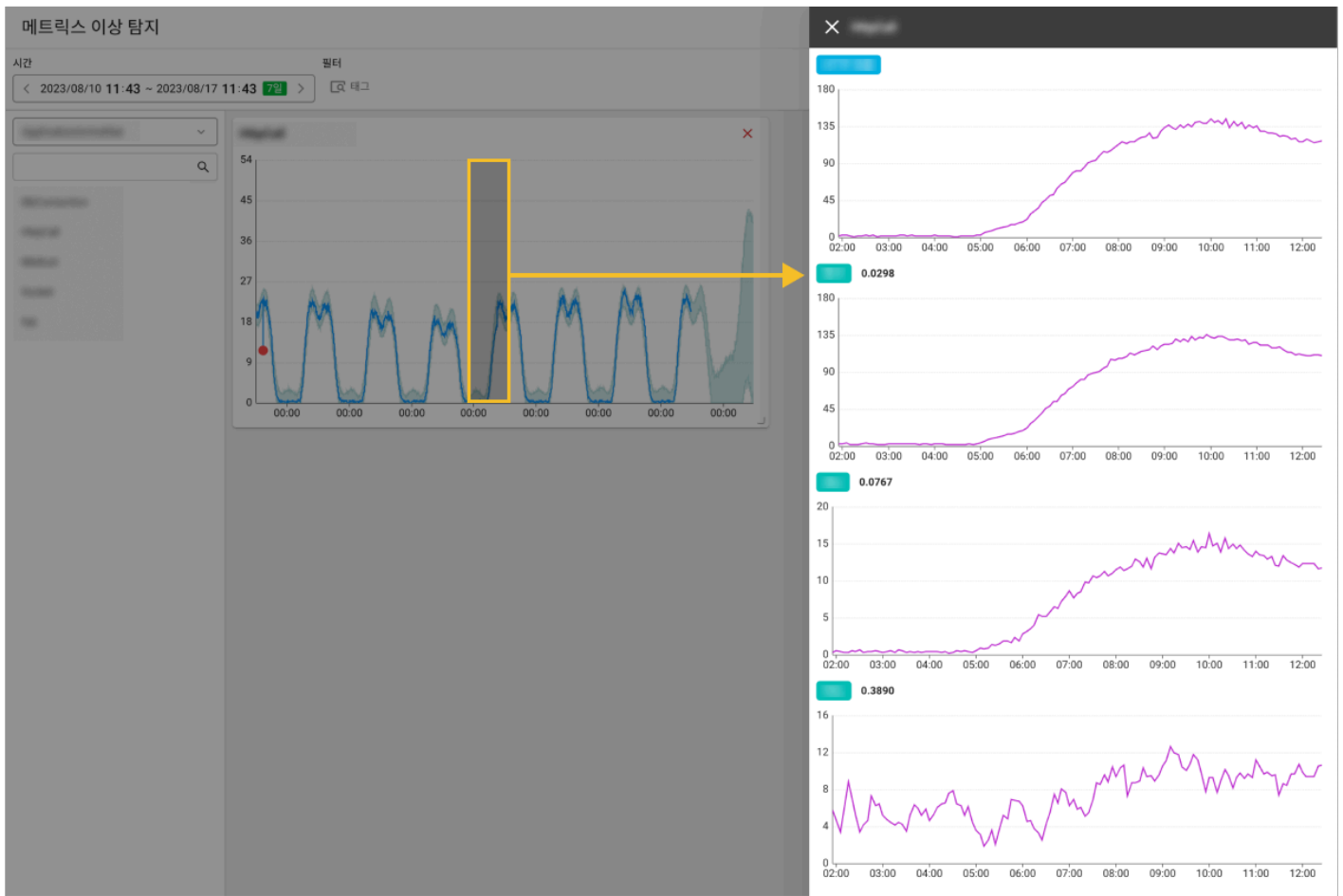


1. ① **시간**에서 원하는 시간 간격을 설정하세요. 최대 1개월 간격까지 설정할 수 있습니다.
2. ② **필터**에서 메트릭스 지표의 범위를 선택하세요.
3. 아래 목록에서 모니터링하길 원하는 메트릭스 지표를 선택하세요.

선택한 메트릭스 지표를 화면 오른쪽에 배치합니다.

- ⓘ • 모니터링 대상을 선택해 메트릭스 지표를 구분해서 확인하려면 ② **태그**를 선택하세요. 화면 오른쪽에 태그 선택 목록이 나타납니다. 원하는 항목을 선택한 다음 위젯을 추가하세요.
 - 화면에 배치한 위젯은 **시간** 또는 **태그** 값을 변경해도 차트에 반영되지 않습니다.
 - 화면에 배치한 위젯을 삭제하려면 위젯 오른쪽 위에 **X** 버튼을 선택하세요.
 - 위젯의 왼쪽 위를 선택한 상태에서 드래그해 위젯 위치를 변경할 수 있습니다.
 - 위젯 오른쪽 아래를 선택한 상태에서 드래그해 위젯 크기를 조절할 수 있습니다.

연관 지표 확인하기



위젯에서 차트의 일부 영역을 드래그하세요. 선택한 영역에서 해당 메트릭스 지표와 관련한 지표를 같이 확인할 수 있습니다.

로그

홈 화면 > 프로젝트 선택 > [로그](#)

쿠버네티스 관련 다양한 로그를 확인할 수 있습니다. 와탭 쿠버네티스에서 제공하는 모니터링 카테고리는 다음과 같습니다.

카테고리	설명
#K8sEvent	<ul style="list-style-type: none"> 쿠버네티스에서 발생하는 이벤트가 저장된 로그 사용자 설정과 무관하게 기본 생성
#WhatapEvent	<ul style="list-style-type: none"> 와탭 이벤트 설정에 의해 발생한 이벤트가 저장된 로그 사용자 설정과 무관하게 기본 생성
containerStdout	<ul style="list-style-type: none"> 컨테이너 Standard Out 로그 사용자 설정 시 생성 노드 에이전트에 <code>logsink_enabled=true</code> 설정 추가 시
AppLog	<ul style="list-style-type: none"> 컨테이너 내 애플리케이션 로그 사용자 설정 시 생성 애플리케이션 에이전트에 <code>logsink_enabled=true</code> 설정 추가 시

ⓘ 설정에 따라 중복된 로그 내용이 저장될 수 있으니 반드시 중복 여부를 확인하세요.

라이브 테일

홈 화면 > 프로젝트 선택 > [로그](#) > [라이브 테일](#)

와탭 쿠버네티스 모니터링 카테고리의 내용을 실시간으로 확인할 수 있습니다. [라이브 테일](#)에 대한 자세한 내용은 [다음 문서](#)를

참조하세요.

로그 트렌드

홈 화면 > 프로젝트 선택 > [로그](#) > [로그 트렌드](#)

시간 범위를 설정하여 저장된 로그 추이를 확인할 수 있습니다. [로그 트렌드](#)에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

로그 검색

홈 화면 > 프로젝트 선택 > [로그](#) > [로그 검색](#)

시간 범위를 설정하여 저장된 로그의 내용을 검색할 수 있습니다. 예를 들어, 특정 컨테이너의 Standard Out 로그를 확인하거나, Deployment에 속하는 모든 컨테이너의 Standard Out 로그를 한번에 확인할 수 있습니다. 특정 키워드를 하이라이트하여 표시할 수도 있습니다.

[로그 트렌드](#)에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

로그 설정하기

홈 화면 > 프로젝트 선택 > [로그](#) > [설정](#)

로그 활성화를 포함한 각종 로그 관련 설정을 할 수 있습니다. 로그 설정에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

보고서

홈 화면 > 프로젝트 선택 > 보고서

CP Kubernetes Monitoring Demo(EKS) ▾ 보고서
🔊 🔍 🔔 📄 ⋮ 👤

보고서
시작일 2023/09/14 📅
시작 시간 00:00 ⌚
종료 시간 23:59 ⌚
필터
🔍
🔄 재생성
📄 다운로드
🖨 인쇄
✉ 보고서 메일 발송 예약
👉 가이드

일간 애플리케이션 보고서 ✕

일간 애플리케이션 보고서

조회조건 2023-09-14(Thu) 00:00:00 ~ 2023-09-14(Thu) 23:59:59 (GMT+9)

▪ 개요

방문자(DAU)	요청	에러비율(%)	이벤트 건수	애플리케이션 서버	CPU코어	APDEX*100
1,007	4,130,099	0.01	24	3	4	71.748

▪ 피크타임 성능요약(08:45)

처리량	동시접속 사용자	요청	평균응답시간(ms)	에러비율(%)	애플리케이션 서버	CPU코어
64.84	1,007	32,338	2,246	0.0	3	3

▪ 지표별 평균/최대값

	동시접속 사용자	시간당 방문자 수	처리량	평균응답시간 (ms)	서버 CPU 사용률	서버 메모리 사용률	자바 CPU 사용률	자바 힙 메모리 사용량(MiB)	에러건수
평균	768.76	1,001.33	36.56	2,916.01	15.3	41.66	0.0	44.52	1.28
최대 (발생시간)	1,007 (06:50)	1,007 (05:00)	87.19 (10:00)	42,353.82 (23:45)	49.14 (09:20)	69.26 (16:00)	0.0 (00:00)	51.47 (18:25)	4 (18:55)

▪ 성능추이차트

동시접속 사용자 CSV

보고서는 개별 프로젝트에 대해 서비스 이용 현황과 장애 발생 기록을 보고하는 문서입니다. 서비스 모니터링 담당자는 보고서를 통해 관련 부서 담당자들과 현황을 공유합니다. 모니터링 데이터 분석은 서비스의 개선 방향을 정하는 지표가 되기 때문에 중요합니다. 하지만 여러 대시보드의 데이터를 취합해서 문서화하는 일은 번거롭습니다.

와탭의 [보고서](#) 메뉴는 [보고서 작성 업무 자동화](#)를 지원합니다. 매주 보고서를 작성해야 하는 일, 정해진 시간에 보고서를 공유하는 일, 여러 가지 서식을 관리하는 일 모두 [보고서](#) 메뉴에서 할 수 있습니다.

- ✔ • 보고서를 다운로드하려면 [다운로드](#) 버튼을 선택하세요. html 파일 형식으로 다운로드 받을 수 있습니다.
- 보고서를 인쇄하려면 [인쇄](#) 버튼을 선택하세요.
- 보고서를 예약 발송하려면 [보고서 메일 발송 예약](#) 버튼을 선택하세요.

❗ 보고서와 통합 보고서의 차이

보고서는 하나의 프로젝트에 대한 보고서를 생성하는 메뉴입니다. 따라서 프로젝트를 선택 후 왼쪽에 보고서 메뉴를 통해 접근이 가능합니다. 통합 보고서는 여러 프로젝트에 대한 보고서를 생성하는 메뉴입니다. 프로젝트를 선택하지 않고 홈 화면에서 기본 메뉴를 통해 접근이 가능합니다. 통합 보고서에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

보고서 종류

쿠버네티스 관련 다양한 보고서를 확인할 수 있습니다. 와탭 쿠버네티스에서 제공하는 보고서 유형은 다음과 같습니다.

- [일간 애플리케이션 보고서](#)
- [일간 애플리케이션 종류별 임계치 초과 트랜잭션 리스트](#)
- [일간 메트릭 추이현황](#)
- [일간 애플리케이션 에러통계 보고서](#)
- [주간 애플리케이션 보고서](#)
- [월간 애플리케이션 보고서](#)
- [POD별 일간 Cpu/Memory 보고서](#)
- [POD별 주간 Cpu/Memory 보고서](#)
- [일간 서버 보고서](#)
- [주간 서버 보고서](#)
- [월간 서버 보고서](#)

메일 발송 예약

정기적으로 보고서를 email로 받고 싶으시면 [보고서 메일 발송 예약](#)을 선택하세요. 출근 직후 수행하던 여러 가지 서비스 점검 절차를 메일 확인으로 대체할 수 있습니다.

보고서 메일 발송 예약

① 등록 다음날 부터 04~06시에 리포트가 발송됩니다.
② 최대 5건의 리포트 메일 설정이 가능합니다.

보고서 종류 **보고 시간** 00:00 ~ 23:59

발송 요일 Mon Tue Wed Thu Fri Sat Sun

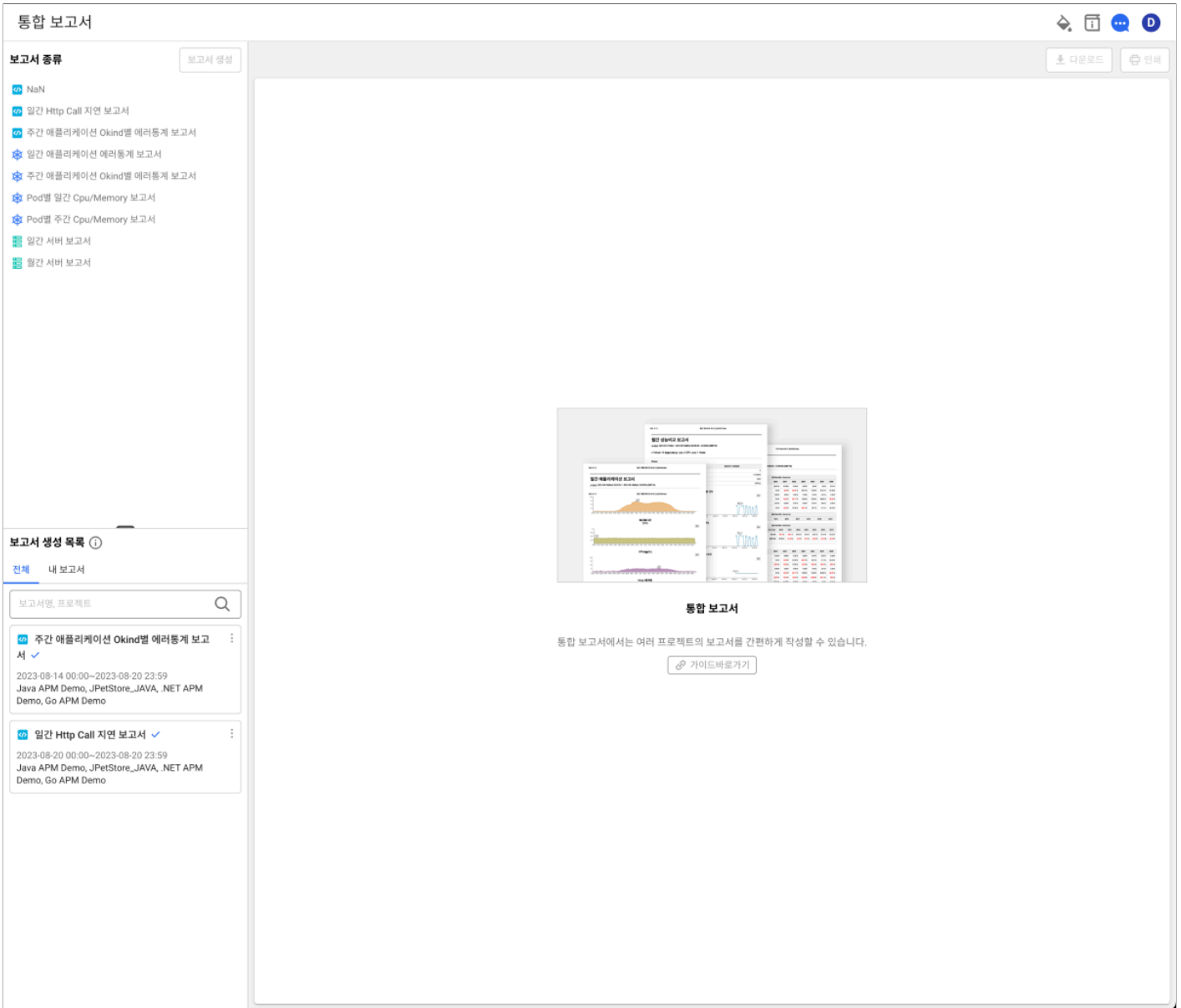
메일

통합 보고서

홈 화면 > [통합 보고서](#)

와탭 모니터링 서비스 초기 화면에서 프로젝트를 선택하지 않고 기본 [메뉴](#) 하위에 [통합 보고서](#)를 선택하세요. [통합 보고서](#)는 여러 프로젝트에 대한 보고서를 간편하게 생성할 수 있습니다.

보고서 종류와 원하는 프로젝트를 선택하여 보고서를 생성할 수 있고, 보고서 생성 목록에서 작업 진행 상태를 확인할 수 있습니다. 보고서 생성 중 특정 프로젝트에서 에러가 발생한 경우 에러 내용을 확인할 수 있습니다. 작업이 완료된 보고서를 바로 조회하거나, 다운로드 및 인쇄 기능을 통해 다른 사람에게 보고서를 공유할 수 있습니다.



통합 보고서의 기능은 다음과 같습니다.

- 멀티 프로젝트 지원

통합 보고서 메뉴에서 여러 프로젝트를 선택해 한 번에 보고서를 생성할 수 있어 편리합니다. 기존 보고서 메뉴는 프로젝트마다 보고서를 생성해야 했습니다.

- 업무 단위 보고서 생성

보고서 생성 메뉴에서 업무 단위로 프로젝트를 선택해 템플릿으로 저장할 수 있습니다. 보통 업무 단위로 프로젝트를 나누어 사용하는 마이크로 서비스 아키텍처(MSA, Micro Service Architecture) 환경에서 유용한 기능입니다.

- **대용량 데이터 최적화**

통합 보고서 메뉴는 보고서 생성 시간이 획기적으로 줄어들었습니다. 그리고 일부 프로젝트 보고서 작성 과정에서 오류가 발생하더라도 보고서 결과를 조회할 수 있습니다.

- **보고서 생성 중 동시 작업 가능**

통합 보고서 메뉴에서 보고서를 생성하는 동시에 다른 업무를 보는 것이 가능합니다. 기존 **보고서** 메뉴는 생성 작업이 완료될 때까지 기다려야만 했습니다.

통합 보고서 메뉴에서 보고서 생성을 시작한 후 다른 페이지로 이동할 수 있습니다. 또한 보고서 결과 목록에서 진행 상태와 완료된 보고서를 조회할 수 있습니다.

- **보고서 작업 공유**

동일 프로젝트 권한을 가진 사용자라면 누구든 생성한 보고서를 조회할 수 있습니다. 관련 보고서가 이미 생성됐는지 확인할 수 있어 중복으로 작성하지 않을 수 있습니다.

보고서와 통합 보고서의 차이

보고서는 하나의 프로젝트에 대한 보고서를 생성하는 메뉴입니다. 따라서 프로젝트를 선택 후 좌측의 **통계/보고서** 메뉴를 통해 접근이 가능합니다.

통합 보고서는 여러 프로젝트에 대한 보고서를 생성하는 메뉴입니다. 프로젝트를 선택하지 않고 기본 메뉴에서 접근이 가능합니다. **보고서**의 상세 내용은 [다음 문서](#)에서 확인할 수 있습니다.

보고서 생성 방법

1. 보고서 종류 선택

보고서 종류

보고서 생성

 **일간 Http Call 지연 보고서**

 주간 애플리케이션 Okind별 에러통계 보고서

 일간 애플리케이션 에러통계 보고서

 주간 애플리케이션 Okind별 에러통계 보고서




 Pod별 일간 Cpu/Memory 보고서

 Pod별 주간 Cpu/Memory 보고서

 일간 서버 보고서

 월간 서버 보고서

보고서 종류 메뉴에서 원하는 보고서 유형을 선택하면 **보고서 생성** 버튼이 활성화됩니다. 보고서 이름 앞의 **아이콘**은 보고서의 양식이 지원하는 프로젝트 플랫폼을 의미합니다.

-  애플리케이션 프로젝트 플랫폼을 나타냅니다.
-  쿠버네티스 프로젝트 플랫폼을 나타냅니다.
-  서버 프로젝트 플랫폼을 나타냅니다.

2. 상세 설정

× 보고서 생성

타입 🔗 일간 Http Call 지연 보고서 ▼

제목 일간 Http Call 지연 보고서

시간 < 2023/08/22 00:00 ~ 2023/08/22 23:59 1일 >

프로젝트

검색 🔍

▼ 프로젝트 전체 선택

🔗 [30092] .NET APM Demo

🔗 [31130] Go APM Demo

🔗 [5490] Java APM Demo

🔗 [6969] Node.js APM Demo

🔗 [31324] PHP APM Demo

🔗 [29744] Python APM Demo

보고서 양식 저장

보고서 생성

보고서 종류 선택 후 [보고서 생성](#) 버튼을 클릭하세요. 다음과 같이 상세 설정을 할 수 있습니다.

- **보고서 제목**

같은 타입의 보고서가 여러 개인 경우 이름을 지정하면 보고서 결과를 구분하기 쉽습니다.

- **시간**

보고서에 사용될 데이터의 기간을 선택하세요. 기간은 보고서 타입(일, 주, 월)에 따라 달라집니다.

- 프로젝트 선택

보고서를 작성할 프로젝트를 선택하세요. 그룹 및 개별 프로젝트 단위로 선택이 가능합니다.

- 보고서 양식 저장과 불러오기

지금 입력된 보고서의 설정을 템플릿으로 저장하고 다음 생성 시 타입 목록에서 불러올 수 있습니다. 업무 단위로 프로젝트를 자주 사용하시는 분들에게 유용한 기능입니다.

3. 보고서 생성

설정을 완료했다면 [보고서 생성](#) 버튼을 눌러 보고서를 생성하세요.

작업 목록 확인

보고서 생성 목록 (i)

[전체](#) [내 보고서](#)

🔍

↔

일간 Http Call 지연 보고서 ✓

2023-08-22 00:00~2023-08-22 23:59

Java APM Demo, .NET APM Demo, Go APM Demo

⋮

[보고서 생성 목록](#)에서 이전에 만들었던 보고서와 현재 작업 중인 보고서를 확인할 수 있습니다.

[전체](#) 목록에서는 내가 생성한 보고서 외에도 나와 같은 권한을 가진 사람이 생성한 보고서를 함께 볼 수 있습니다. [내 보고서](#) 목록에서는 내가 생성한 보고서만 필터링하여 볼 수 있습니다.

에러 없이 생성된 보고서는 **✓ 체크** 아이콘으로 표시되며, 에러가 포함된 경우는 **⚠ 경고** 아이콘이 함께 표시됩니다. **⚠ 경고** 아이콘을 클릭하시면 어떤 프로젝트에서 에러가 발생했는지 확인할 수 있습니다.

조회 및 공유

↓ 다운로드 ☰ 인쇄

일간 Http Call 지연 보고서

조회조건 :2023-08-22(Tue) 00:00:00 ~ 2023-08-22(Tue) 23:59:59 (GMT+9)

[5490] Java APM Demo

일간 Http Call 지연 보고서

조회조건 2023-08-22(Tue) 00:00:00 ~ 2023-08-22(Tue) 23:59:59 (GMT+9)

Http Call 지연 Top 5 (1초 이상)

번호	URL	호출 URL	호스트	포트	전체	에러	평균 시간(ms)	합계 시간(ms)	최대 시
1	/remote/account/save/dept/kwangj	/account/load/dept/daejun	127.0.0.1	8105	2264	0	1247	2823259	

보고서는 3가지 방식으로 제공됩니다.

- **화면 조회**

작업 목록에서 생성한 보고서를 선택하면 화면에서 바로 조회하여 볼 수 있습니다.

- **다운로드**

[다운로드](#) 버튼을 클릭하면 바로 보기의 보고서와 같은 파일이 html 형식으로 다운로드 됩니다. 모니터링 관리 권한이 없는 사람에게 보고서를 공유할 수 있습니다.

- **인쇄 및 PDF 저장**

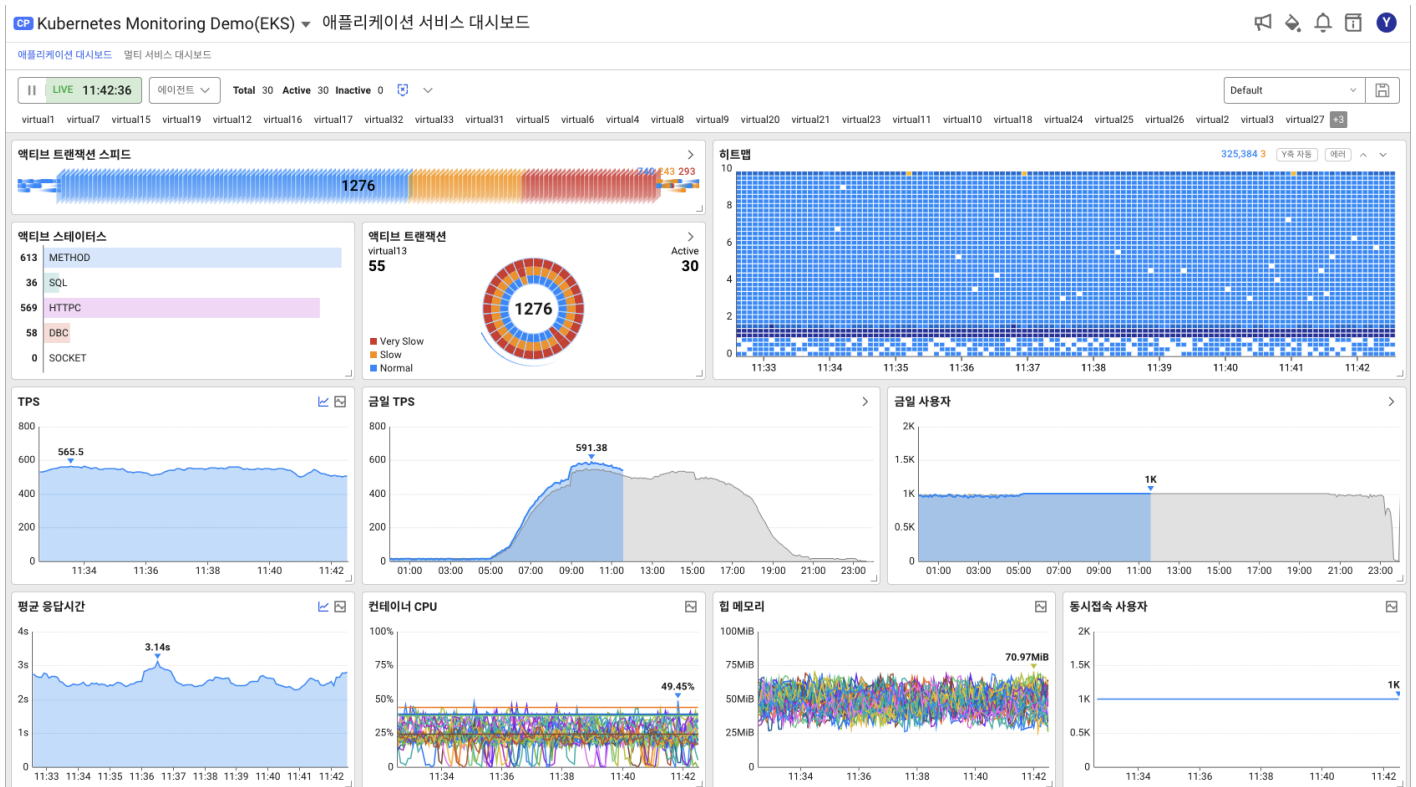
[인쇄](#) 버튼을 누르면 컴퓨터에 연결된 프린터로 인쇄하거나 또는 PDF 형식으로 저장할 수 있습니다.

애플리케이션 서비스 대시보드

홈 화면 > 프로젝트 선택 > 애플리케이션 > 대시보드 > 애플리케이션 서비스 대시보드

컨테이너 내부에서 실행 중인 애플리케이션 상태를 실시간으로 확인할 수 있는 대시보드입니다. 각 위젯의 정상적인 데이터 출력을 위해서는 쿠버네티스 환경에 노드 에이전트를 설치해야 하고, 컨테이너 내부의 애플리케이션에도 와탭 에이전트를 설치해야 합니다.

- ❗ [Java](#), [Node.js](#), [Python](#), [PHP](#)로 개발한 애플리케이션에 와탭 에이전트 설치를 지원합니다.
- 대시보드 메뉴 또는 애플리케이션 메뉴 하위의 대시보드 메뉴를 통해 접근할 수 있습니다.

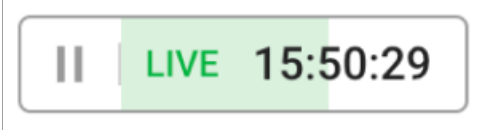


애플리케이션 대시보드에서는 애플리케이션의 주요 지표들을 에이전트를 통해 수집하고 차트를 구성해 실시간으로 모니터링하고 과거의 데이터를 조회할 수 있습니다. 대시보드에 배치된 위젯들을 통해 애플리케이션의 문제를 빠르게 파악하고 분석할 수 있습니다. CPU, Memory를 제외한 일반적인 차트의 경우 안정적인 데이터는 파란색 계열로 표현되고, 문제로 식별되는 요소들은 붉은색 계열로 표시되어 현황을 쉽게 인지할 수 있습니다.

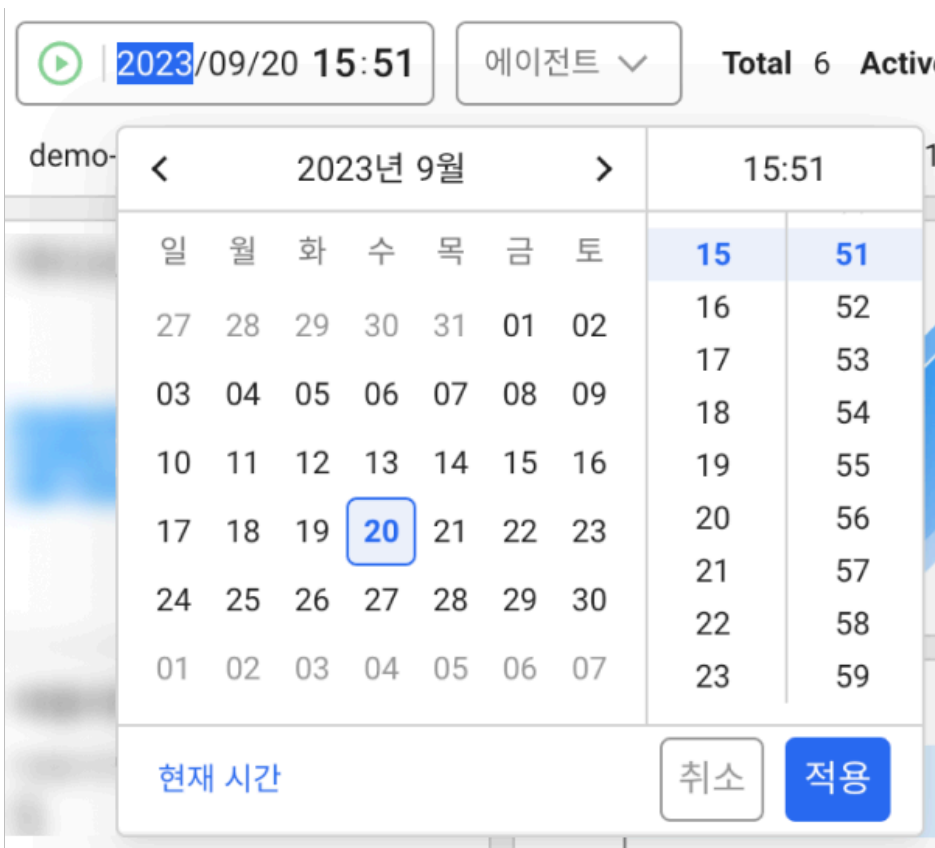
애플리케이션 대시보드

과거 데이터 조회하기

애플리케이션 대시보드는 기본적으로 실시간 모니터링 기능을 제공하지만 과거 시간의 데이터를 조회할 수도 있습니다.

실시간 모드	비실시간 모드
	

1. 화면 왼쪽 위에 시간 선택자에서 || 버튼을 선택하세요.
2. 날짜 및 시간 영역을 클릭해 원하는 시간대를 설정하세요.

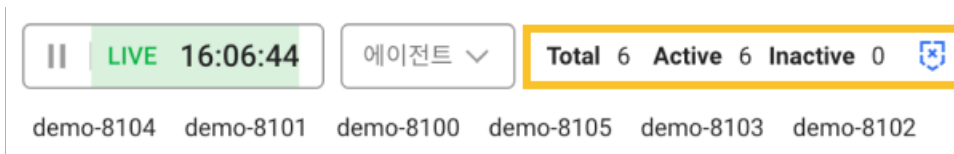


3. 적용 버튼을 선택하세요.


사용자가 설정한 시간을 기준으로 대시보드에 배치한 위젯의 데이터를 갱신합니다. 실시간 모드로 전환하려면 ▶ 버튼을 선택하세요.

에이전트 확인하기

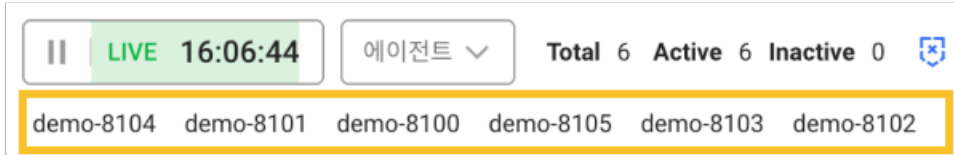
에이전트 연결 상태 확인하기



화면 왼쪽 위, 시간 선택자의 오른쪽에서는 해당 프로젝트와 연결된 에이전트의 상태를 확인할 수 있는 정보를 제공합니다. 이를 통해 애플리케이션 서버의 동작 여부를 바로 확인할 수 있습니다.

- **Total**: 프로젝트와 연결된 모든 에이전트의 수
- **Active**: 활성화된 에이전트의 수
- **Inactive**: 비활성화된 에이전트의 수
- : 비활성화된 에이전트를 표시하거나 감출 수 있습니다.

에이전트별 모니터링



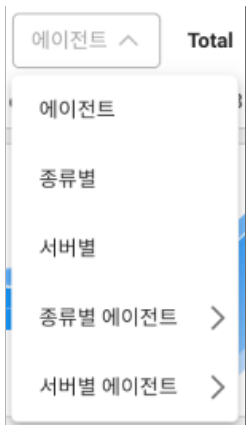
The screenshot shows a control bar with a pause icon, a 'LIVE' indicator, a timer '16:06:44', a dropdown menu for '에이전트', and summary statistics: 'Total 6 Active 6 Inactive 0'. Below this is a list of agent IDs: 'demo-8104 demo-8101 demo-8100 demo-8105 demo-8103 demo-8102'.

기본적으로 대시보드에는 모든 에이전트로부터 수집한 지표들을 차트에 표시하지만 에이전트별로 데이터를 조회할 수도 있습니다. 시간 선택자 아래에 위치한 에이전트를 하나 또는 둘 이상을 선택하세요. 선택한 에이전트의 지표들로 대시보드에 배치된 위젯의 데이터를 갱신합니다.

✔ 에이전트를 하나 또는 둘 이상을 선택한 상태에서 다시 모든 에이전트를 선택하려면 선택을 해제하거나 **Total**을 선택하세요.

ⓘ 프로젝트에 연결된 에이전트의 수가 많을 경우 에이전트의 이름을 짧게 설정하는 것이 효율적입니다. 에이전트 이름 설정에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

분류별 에이전트 모니터링



The screenshot shows a dropdown menu for '에이전트' with a 'Total' button. The menu items are: '에이전트', '종류별', '서버별', '종류별 에이전트 >', and '서버별 에이전트 >'.

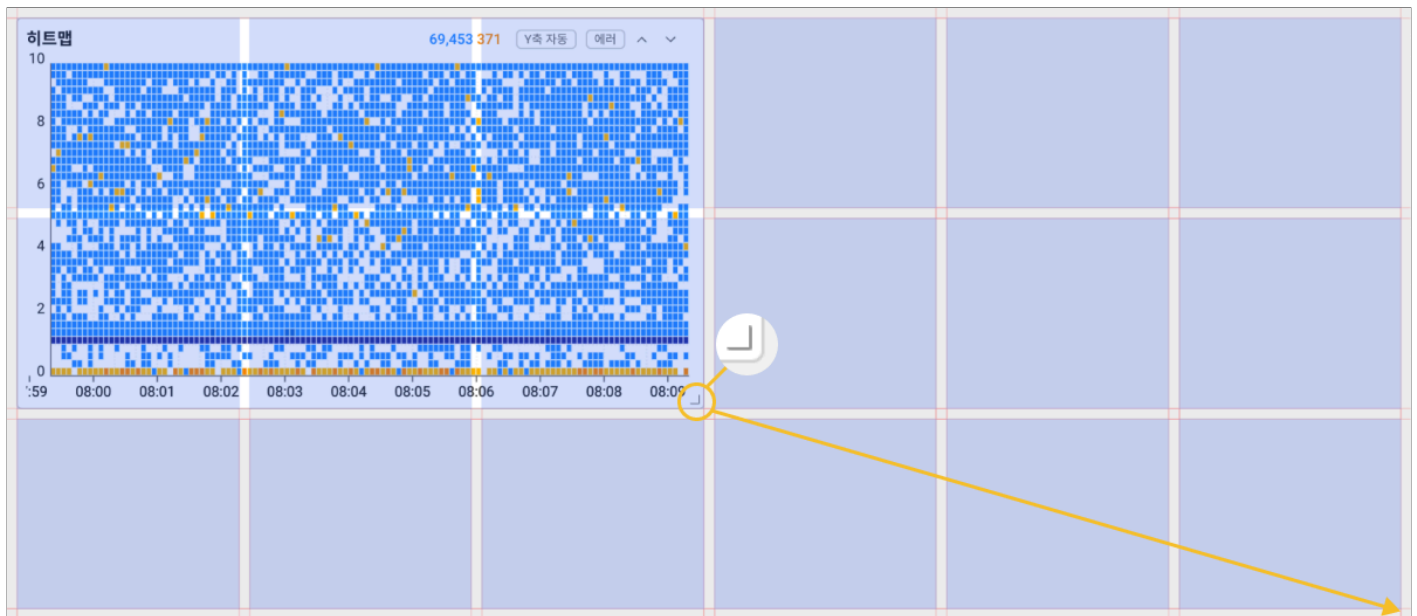
에이전트 설정을 통해 분류한 그룹 단위로 에이전트를 선택해 모니터링할 수 있습니다.


- **에이전트**: 개별 에이전트를 선택하거나 모두 선택할 수 있습니다.
- **종류별**: 에이전트 설정에서 `whatap.okind` 로 분류된 그룹 단위로 에이전트를 모니터링합니다.
- **서버별**: 에이전트 설정에서 `whatap.onode` 로 분류된 그룹 단위로 에이전트를 모니터링합니다.
- **종류별 에이전트**: 에이전트 설정에서 `whatap.okind` 로 분류된 에이전트 목록만을 모니터링합니다.
- **서버별 에이전트**: 에이전트 설정에서 `whatap.onode` 로 분류된 에이전트 목록만을 모니터링합니다.

대시보드 위젯 편집하기

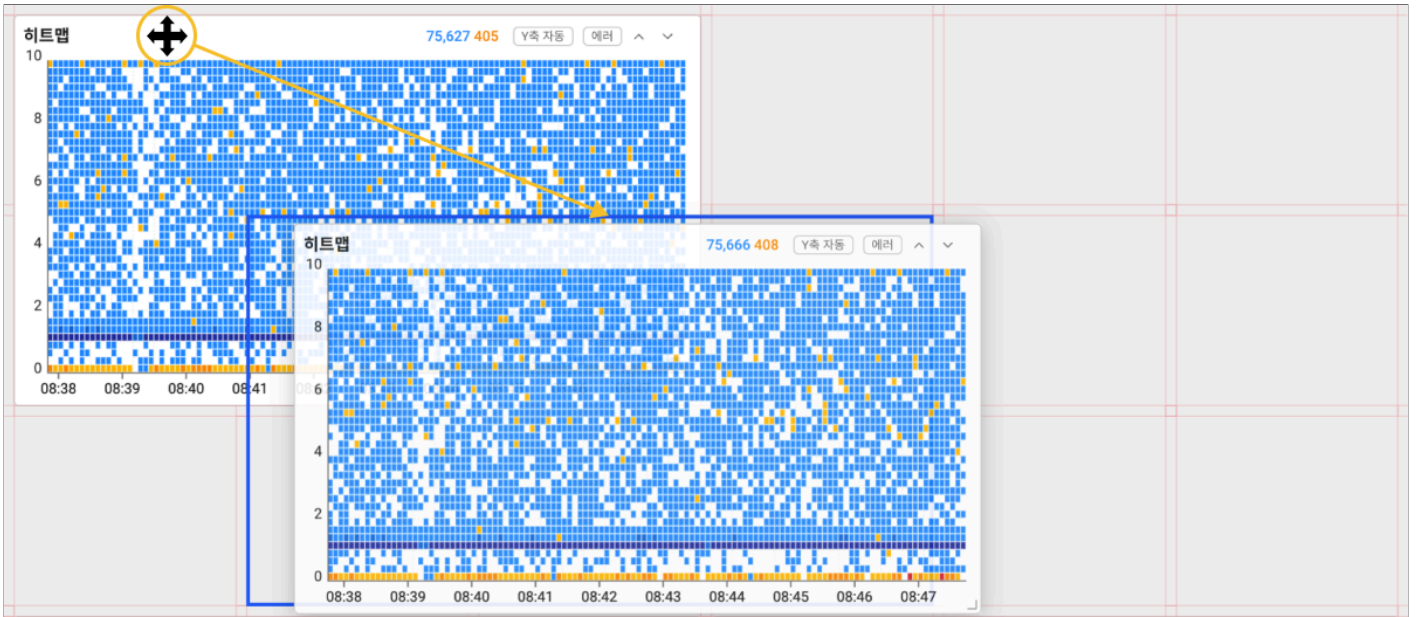
대시보드에 배치한 위젯은 사용자가 원하는 크기로 조절할 수 있고, 원하는 위치에 배치할 수 있습니다. 불필요하다고 생각되는 위젯은 삭제하고 다시 추가할 수도 있습니다.

위젯 크기 조절하기



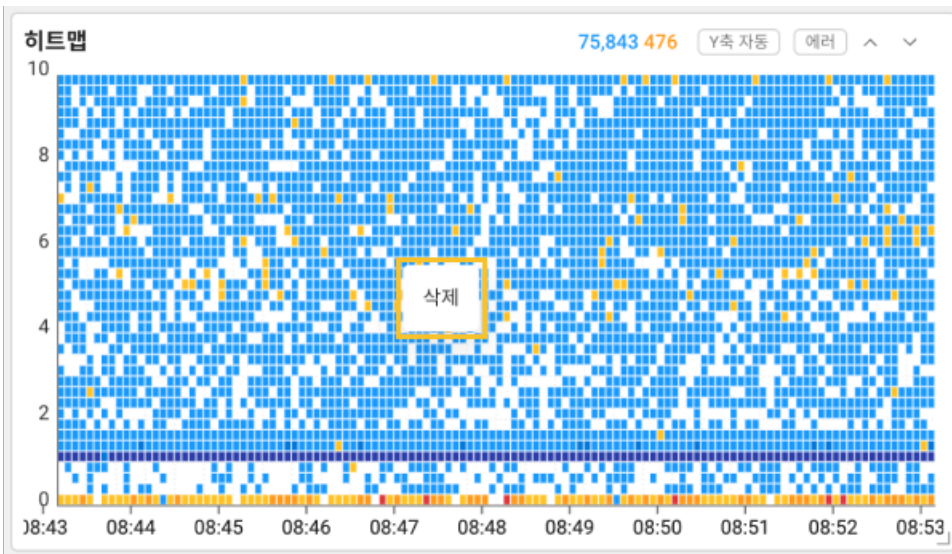
위젯의 오른쪽 아래에  요소를 마우스로 클릭한 상태에서 원하는 크기로 드래그하세요. 균일한 가로, 세로 비율의 격자가 표시되고, 격자 단위로 위젯의 크기를 조절할 수 있습니다.

위젯 이동하기



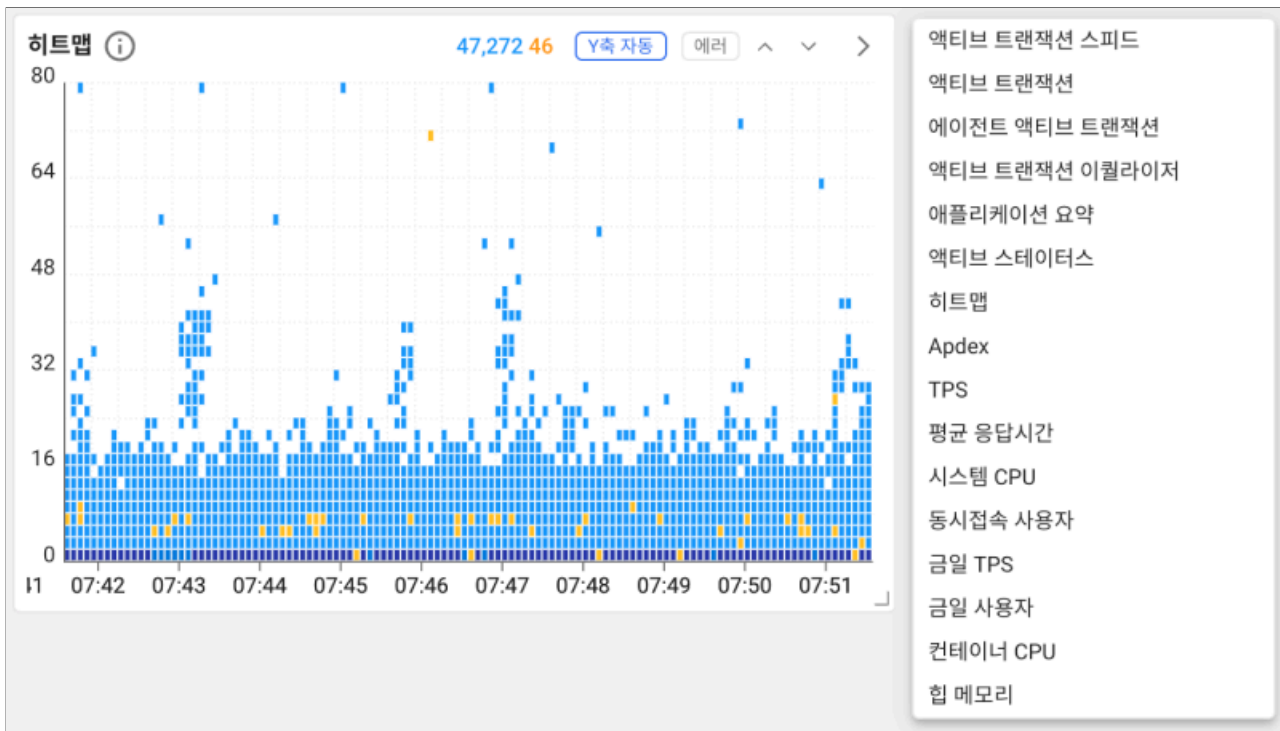
위젯의 위 부분으로 마우스 커서를 이동하면 커서 모양이 **+** 모양으로 변경됩니다. 이때 마우스 왼쪽 버튼을 클릭한 상태로 원하는 위치로 드래그하여 위젯을 이동할 수 있습니다.

위젯 삭제하기



삭제하길 원하는 위젯에서 마우스 오른쪽 버튼을 클릭하세요. 삭제 버튼을 선택하면 해당 위젯이 대시보드에서 삭제됩니다.

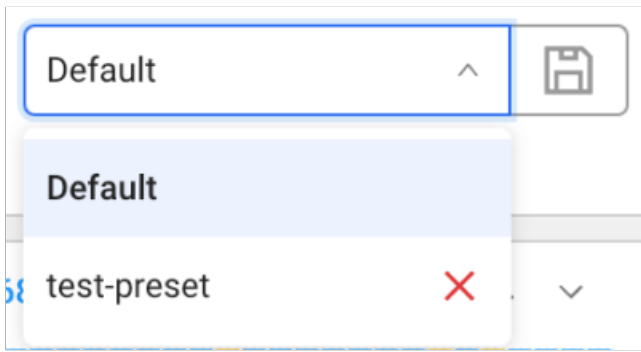
위젯 추가하기



대시보드에서 빈 공간으로 마우스 커서를 이동한 다음 마우스 오른쪽 버튼을 클릭하세요. 팝업 메뉴에서 추가하려는 위젯을 선택하세요. 원하는 위치로 위젯을 배치하고 크기를 조절하세요.


ⓘ 현재 추가할 수 있는 위젯은 고정적이지만 향후 업데이트를 통해 위젯 지원을 늘려갈 계획입니다.

프리셋



대시보드에서 사용자가 설정한 위젯의 설정과 레이아웃 상태를 저장하고 불러올 수 있습니다. 기본 프리셋(Default)은 변경할 수 없으며, 원하는 형식으로 위젯의 크기를 조절하고, 배치해 새로운 프리셋을 만들 수 있습니다.


새로운 프리셋 만들기

1. 대시보드에서 원하는 형식으로 위젯을 배치해 보세요. 크기를 조절하고 자주 확인하는 위젯만 배치할 수도 있습니다.
2. 화면 오른쪽 위에  버튼을 선택하세요.
3. 새로운 프리셋 이름을 입력하세요.


에이전트 선택 내역을 같이 저장하려면 [에이전트 선택 내역 포함](#)을 선택하세요.

4. [저장](#) 버튼을 선택하세요.

프리셋 목록에서 새로 저장한 프리셋을 확인할 수 있습니다.



- ① 새로 만든 프리셋에 변경 사항이 생겼다면 다시 프리셋을 저장해야 합니다.  버튼을 선택한 다음 같은 이름으로 프리셋을 저장하세요. 기존의 프리셋에 변경 사항을 덮어쓰기합니다.
- 대시보드의 변경 사항을 저장하지 않고 다른 메뉴로 이동하면 변경 사항은 저장되지 않습니다.
- 프리셋은 사용자 계정으로 저장되며 현재는 다른 사용자와 공유할 수 없습니다. 더 나은 기능을 제공하도록 업데이트 예정입니다.

프리셋 삭제하기

사용하지 않는 프리셋이 있다면 프리셋 목록에서 삭제할 수 있습니다. 프리셋 목록에서 삭제하려는 항목의 오른쪽에  버튼을 선택하세요.

위젯 옵션 이용하기

위젯의 위에 표시된 아이콘 버튼의 기능은 다음과 같습니다.

- ① : 주요 위젯에 대한 기능 및 정보를 확인할 수 있습니다.
-  병합 / 개별로 보기: 해당 위젯 항목의 에이전트 데이터를 개별 또는 병합해 그래프로 표시합니다.
-  상세: 해당 위젯 항목의 데이터를 에이전트별로 구분해 조회할 수 있는 모달 창이 나타납니다.

- ① 위젯에 따라 제공되는 옵션은 다를 수 있습니다.

위젯 알아보기

- ① 이 문서에서는 [애플리케이션 대시보드](#)에 배치된 위젯에 대한 설명을 주로 다룹니다. [애플리케이션 대시보드](#)를 통해 트랜잭션 및 사용자, 서비스, 리소스 분석 방법에 대한 자세한 내용은 다음 문서를 참조하세요.

- [애플리케이션 대시보드 살펴보기](#)
- [히트맵 트랜잭션](#)
- [DB 연결 지연과 커넥션 풀](#)

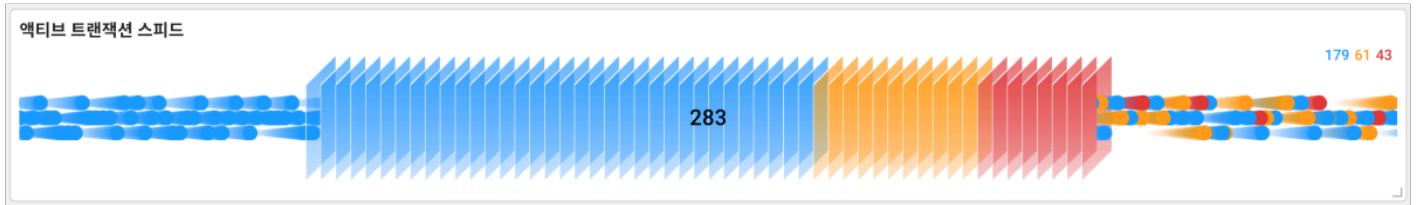
에이전트 관련 위젯

애플리케이션에 설치한 에이전트를 통해서 애플리케이션의 정보를 확인할 수 있습니다.

트랜잭션 관련 위젯

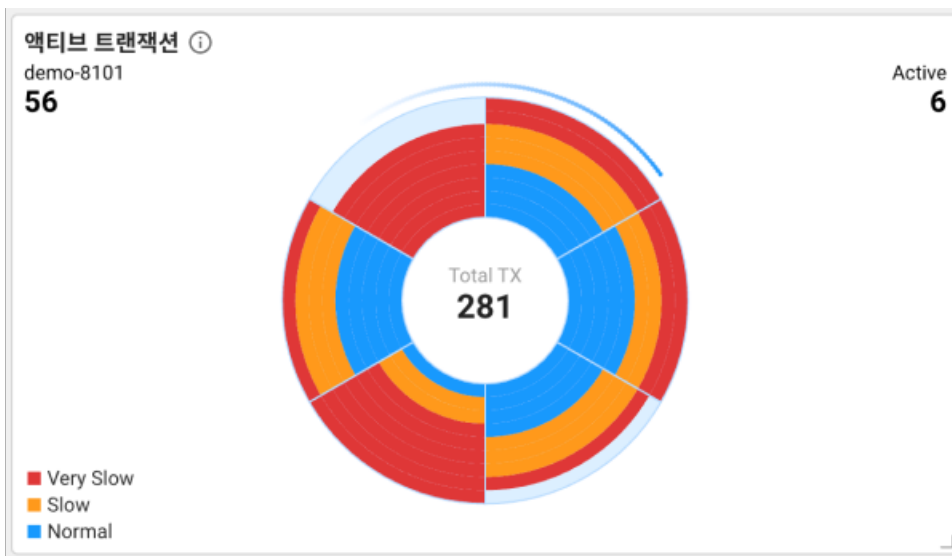
트랜잭션을 통해 확인할 수 있는 장애의 현황은 우선 응답시간을 통해 알 수 있습니다. 또한 진행 중인 트랜잭션이 종료되지 않는다면 이 또한 장애로 인식해야 합니다. 와탭은 진행 중인 상태의 시간에 따라 구간을 나누어 표시합니다. **파랑색(Normal)**은 응답 시간이 정상인 트랜잭션, **주황색(Slow)**은 응답 시간이 8초 정도의 느린 트랜잭션, **빨간색(Very slow)**은 응답 시간이 보통의 2배 이상으로 느린 트랜잭션을 의미합니다. 이를 통해 사용자는 직관적으로 가장 빨리 장애를 인지할 수 있습니다.

액티브 트랜잭션 스피드



현재 진행 중인 트랜잭션(가운데 영역)과 종료된 트랜잭션(오른쪽 영역) 현황을 확인할 수 있습니다. 왼쪽에서 오른쪽으로 지나가는 물방울 모양의 아이콘은 트랜잭션을 의미합니다. 액티브 트랜잭션의 상태를 속도(Normal, Slow, Very Slow)로 색상으로 분류하여 우측 상단에 개수로 표시합니다.

액티브 트랜잭션



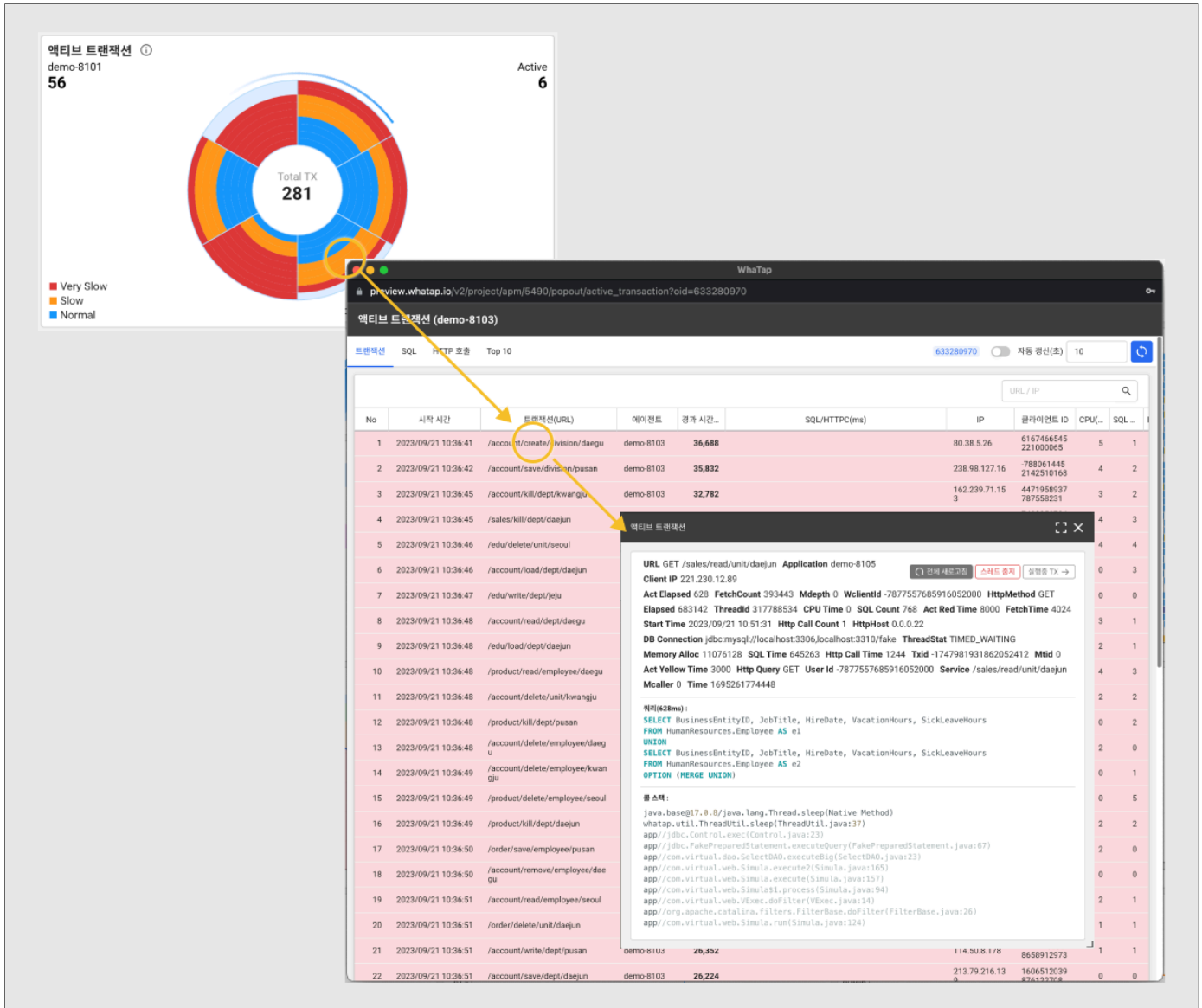
액티브 트랜잭션 위젯에서는 진행 중인 트랜잭션을 **아크 이퀄라이저** 차트로 표현합니다. 지연 발생 현황을 에이전트별로 확인할 수 있습니다. 에이전트 수만큼 아크가 분할됩니다. 5초마다 현재 서버에서 처리 중인 요청의 수를 표현해서 해당 요청이 각각 어느 정도의 시간 동안 처리 중인지 알 수 있습니다. 5초 간격의 시간에 감지된 요청들은 위험 여부를 파악할 수 있도록 다음과 같이 색으로

분류합니다.

- 빨간색(**Very Slow**): 8초 이상
- 주황색(**Slow**): 3초 ~ 8초
- 파란색(**Normal**): 0초 ~ 3초

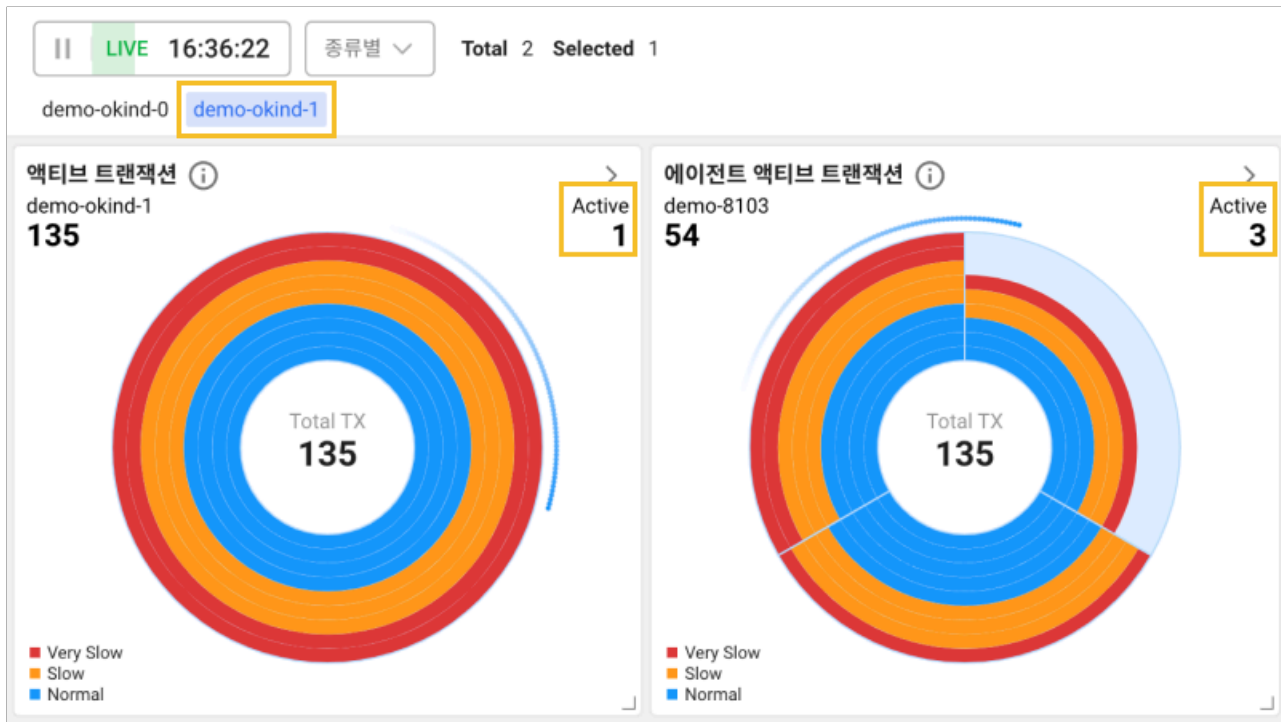
파란색이 많이 표현되는 상황은 문제가 되지 않습니다. 이 중 일부가 주황색이나 빨간색으로 변경되는지 추이를 지켜봐야 합니다. 장애 상황에는 진행 중 트랜잭션 수가 증가하고 빨간색 비율이 높아집니다. 반면, 응답이 매우 빠른 시스템의 경우 처리하는 트랜잭션 수(TPS)는 높아도 진행 중인 트랜잭션 수는 낮을 수 있습니다.

- 왼쪽 위에는 진행 중인 트랜잭션이 가장 많은 에이전트 이름을 표시합니다.
- 가운데 숫자는 전체 에이전트의 액티브 트랜잭션 수의 합계입니다.
- 액티브 트랜잭션 둘레에 두 개의 바는 처리량(TPS)에 따라 3단계 속도로 차트 주변을 회전합니다.
- 분할된 아크 영역을 선택하면 해당 에이전트에서 진행 중인 트랜잭션 목록 팝업창이 나타납니다. 상세한 트랜잭션 정보를 확인하려면 트랜잭션 목록에서 원하는 항목을 하나 선택하세요.



- ① 트랜잭션 목록에서 선택한 항목이 이미 종료된 트랜잭션일 경우 **트랜잭션 보기** 버튼을 선택하세요. **트랜잭션 정보** 창이 나타납니다. **트랜잭션 정보** 창에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.
- 트랜잭션에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

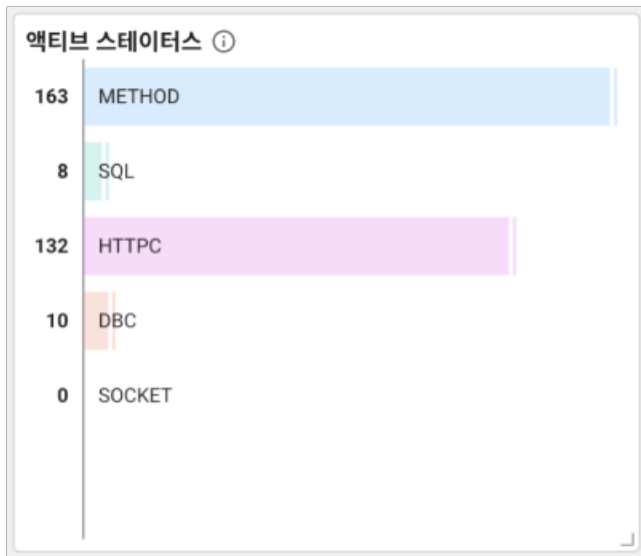
에이전트 액티브 트랜잭션



액티브 트랜잭션 위젯은 에이전트 선택 옵션에서 선택한 기준에 따라 병합된 데이터를 차트에 표시합니다. 반면 에이전트 액티브 트랜잭션 위젯은 선택한 에이전트 분류 기준에 포함된 에이전트의 현황을 개별로 확인할 수 있습니다.

대시보드에서 마우스 오른쪽 버튼을 클릭한 다음 위젯 목록에서 에이전트 액티브 트랜잭션 위젯을 선택하세요.

액티브 스테이터스



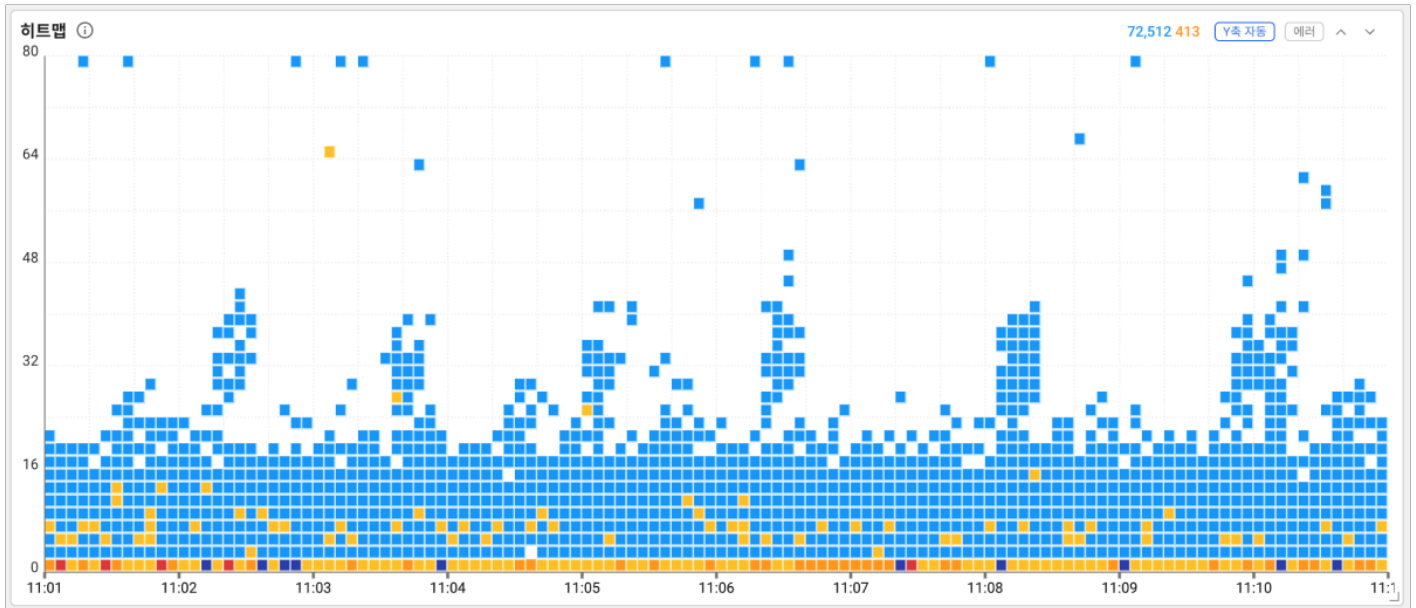
프로젝트 내의 모든 에이전트는 5초마다 액티브 트랜잭션 수를 수집함과 동시에 진행 상태를 수집합니다. 진행 상태는 **METHOD**, **SQL**, **HTTPC**, **DBC**, **SOCKET**으로 분류할 수 있습니다.

- **METHOD**: 메소드를 수행 중인 상태입니다. 일반적인 상황으로 파란색으로 표시됩니다.
- **SQL**: SQL을 수행 중인 상태입니다. **SQL** 수치가 증가한다면 데이터베이스 서버와의 연결에 문제를 확인해 보세요.
- **HTTPC**: 외부 API를 호출 중인 상태입니다. **HTTPC** 수치가 증가한다면 외부와 연결된 서버의 응답이 제대로 이루어지지 않는다고 봐야 합니다.
- **DBC**: 트랜잭션이 Connection Pool로부터 새로운 Connection을 획득(get)하려는 상태입니다. DB Connection Pool의 개수가 부족하면 새로운 연결 요청이 발생할 때마다 지연이 되면서 성능 장애의 원인이 됩니다. 이 경우 **DBC** 수치가 증가합니다.

ⓘ DB 연결 지연과 관련한 자세한 내용은 [다음 문서](#)를 참조하세요.

- **SOCKET**: 외부 시스템과의 TCP 연결 시도를 의미합니다. **SOCKET** 수치가 지속적으로 증가한다는 것은 외부 시스템과의 연결이 되지 않아 장애가 발생 중일 가능성이 높습니다.

히트맵



히트맵 차트는 종료된 트랜잭션 응답시간을 분포도로 표현합니다. 가로축은 트랜잭션 종료 시간입니다. 세로축은 수행 시간입니다. 수행 시간이 긴 트랜잭션은 분포도 상단에 위치합니다. 히트맵 내의 차트 영역을 드래그하면 세부 트랜잭션 정보를 확인할 수 있는 [트레이스 분석](#) 창이 나타납니다. [트레이스 분석](#)에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

- 정상 트랜잭션은 하늘색 → 파란색 → 남색 순으로 표현됩니다.
- 에러 트랜잭션은 노란색 → 주황색 → 빨간색 순으로 표현됩니다.
- 오른쪽 위에 위치한 숫자는 차트 내 **트랜잭션 건 수**/**에러 건 수**를 의미합니다.
- **Y축 자동** 버튼을 선택하면 수행 시간을 나타내는 Y축을 자동 조정합니다.
- **에러** 버튼을 선택하면 에러 트랜잭션만 표시합니다.
- ^ 또는 v 버튼을 선택해 5초 ~ 80초까지 차트를 확대/축소할 수 있습니다.

- ⓘ • 오류가 발생하지 않았고 수행 시간 500ms 이하 트랜잭션의 세부 정보는 url 당 5분에 1건만 수집됩니다.
 - TPS나 평균 응답시간 같은 통계정보에는 영향을 미치지 않습니다.
 - 에이전트 설정을 통해 정책을 설정할 수 있습니다. 자세한 내용은 [다음 문서](#)를 참조하세요.
 - 히트맵 위젯을 통해 트랜잭션을 분석하는 과정은 [다음 문서](#)를 참조하세요.

서비스 관련 위젯

서비스 관련 위젯을 통해 애플리케이션 성능 튜닝을 위한 지표로 활용할 수 있습니다.

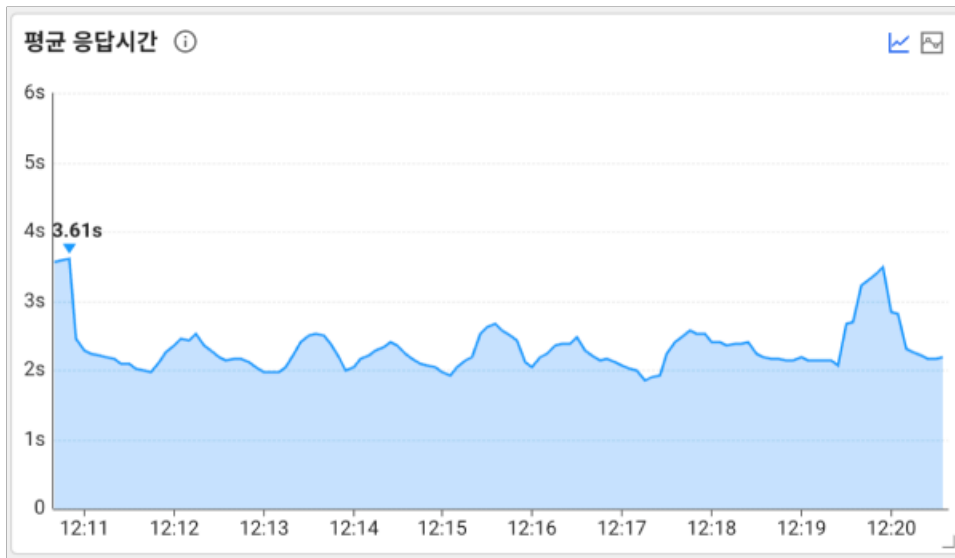
TPS



Transaction Per Second(TPS)는 초당 처리한 트랜잭션 건수를 의미하며 서비스 성능 지표의 기준이 됩니다. 5초마다 처리한 트랜잭션의 수를 초당 건수로 환산해 차트로 표현합니다. 최근 10분간의 TPS를 보여줍니다.

ⓘ TPS, 응답시간, CPU 사용률 분석에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

평균 응답시간

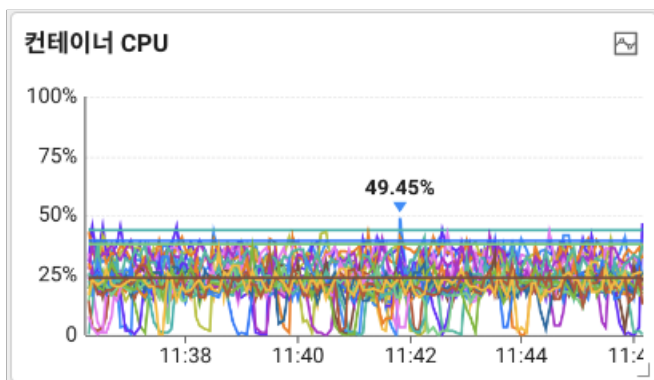


애플리케이션 서버가 사용자에게 요청 결과를 반환하는 데 걸리는 시간을 실시간으로 모니터링합니다. **평균 응답시간**은 튜닝 지표로서 의미를 가집니다. 와탭의 모니터링 서비스는 5초 간격으로 트랜잭션의 평균 응답 시간을 계산합니다.

❗ TPS, 응답시간, CPU 사용률 분석에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

리소스 관련 위젯

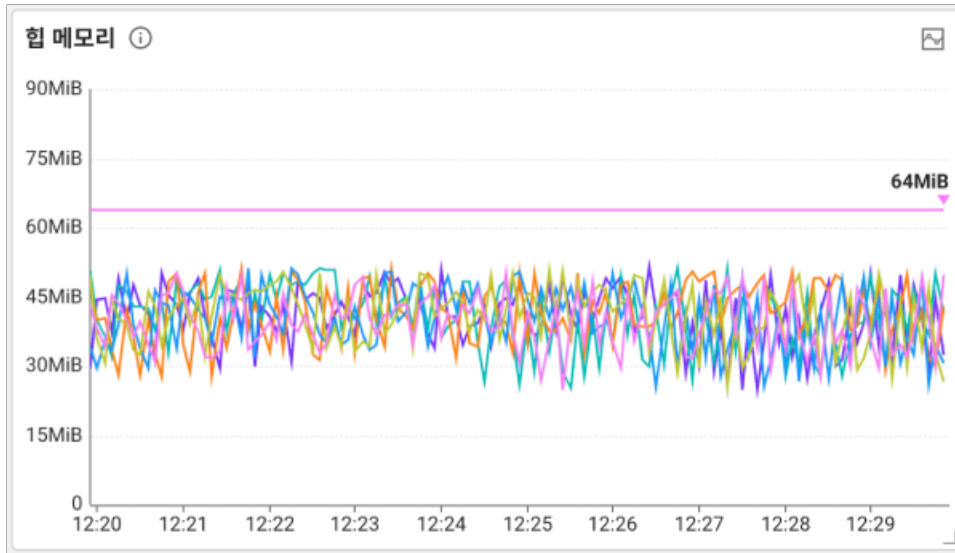
컨테이너 CPU



컨테이너 CPU 사용량입니다. 실시간으로 CPU 사용량 변화 추이를 파악할 수 있습니다.

❗ TPS, 응답시간, CPU 사용률 분석에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

힙 메모리



각 서버당 사용할 수 있는 최대 메모리와 현재 메모리를 표시해 위험 수치에 있는 서버를 확인할 수 있습니다. 시간에 따른 메모리 사용량 변화를 실시간으로 볼 수 있습니다.

메모리 라인 차트는 보통 계속해서 물결칩니다. 애플리케이션 서버가 요청을 처리하기 위해 메모리를 사용할 때 증가합니다. GC(Garbage Collection)를 통해서 메모리를 정리할 경우에는 감소합니다. [힙 메모리](#) 지표는 평균값을 사용해 표기합니다.

- ❗ • Java와 Node.js는 힙 메모리 총량과 사용량을 보여줍니다.
- PHP 및 Python, .Net은 프로세스 메모리 사용량을 보여줍니다.

- ✅ 힙 메모리 및 힙 메모리 차트 분석에 대한 자세한 내용은 다음 링크를 참조하세요.
 - [학습하기: 힙 메모리 매트릭스 및 매트릭스 알림 설정](#)
 - [월간 와탭: 모니터링에 주목해야할 지표](#)
 - [Java 힙 메모리 차트 분석: Ch.1 힙차트 관찰하기](#)

- [JAVA 힙메모리 차트 분석: Ch.2 메모리 릭, 그리고 힙덤프 분석](#)

사용자 관련 위젯

동시접속 사용자



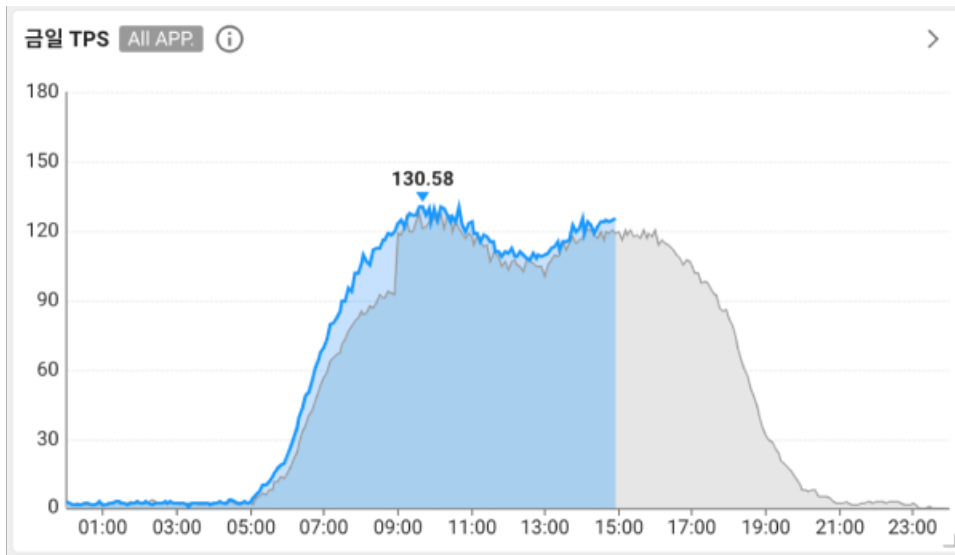
실시간 브라우저 사용자 수를 보여줍니다. 5초마다 최근 5분 이내에 트랜잭션을 일으킨 사용자를 중복없이 카운팅해 표시합니다. 사용자 브라우저의 IP를 기반으로 카운팅합니다.

- ① 일반적으로 접속 사용자는 현재 네트워크로 접속해 연결된 사용자를 의미합니다. 하지만 웹 시스템은 비 연결 네트워크를 사용하기 때문에 다릅니다. 접속되어 있다는 의미보다는 최근 요청을 보낸 사용자가 측정 대상이 될 수 있습니다.
- 실시간(동시접속) 사용자는 같은 시간대에 시스템을 사용하고 있는 사용자를 의미합니다. 실시간(동시접속) 사용자를 산정하거나 측정하는 방식은 다양합니다.
- 사용자 수는 단순히 합산하는 것이 아니라 **HyperLogLog**로 산출합니다. **HyperLogLog**는 매우 적은 메모리로 집합의 원소 개수를 추정하는 확률적 자료 구조를 의미합니다.

1일 기준 비교 위젯

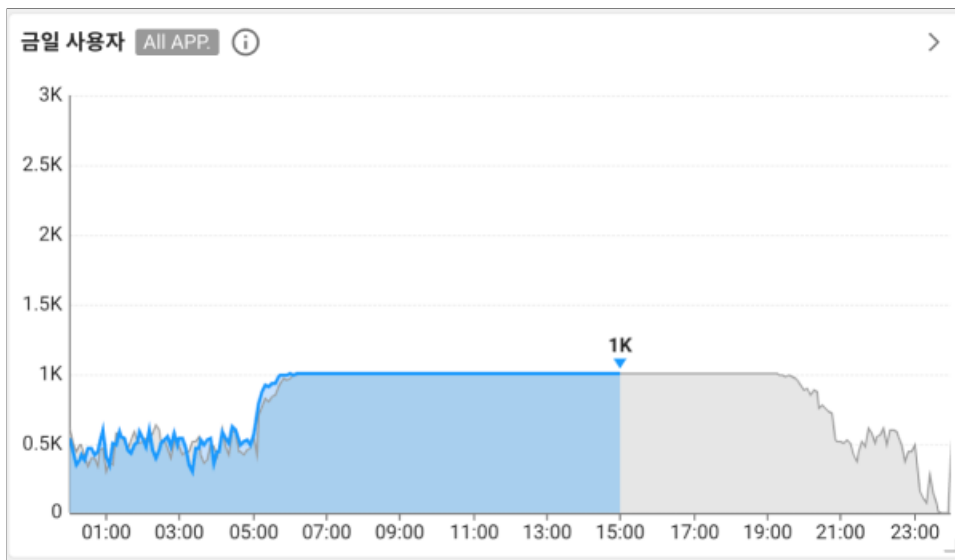
- ① All APPS. 태그가 표시된 위젯은 항상 모든 에이전트에 대한 데이터를 표시합니다.

금일 TPS




금일 하루 동안의 TPS를 시간대별로 표시합니다. 어제 동시간대의 TPS(회색)와 비교해 확인할 수 있습니다. > 버튼을 클릭하면 [통계/보고서 > 일자별 애플리케이션 현황](#) 메뉴로 이동합니다.

금일 사용자



하루 동안의 [동시접속 사용자](#) 수를 시간대별로 나타낸 지표입니다. 어제 동시간대의 사용자 수(회색)와 비교해 확인할 수 있습니다. > 버튼을 클릭하면 [통계/보고서 > 일자별 애플리케이션 현황](#) 메뉴로 이동합니다.

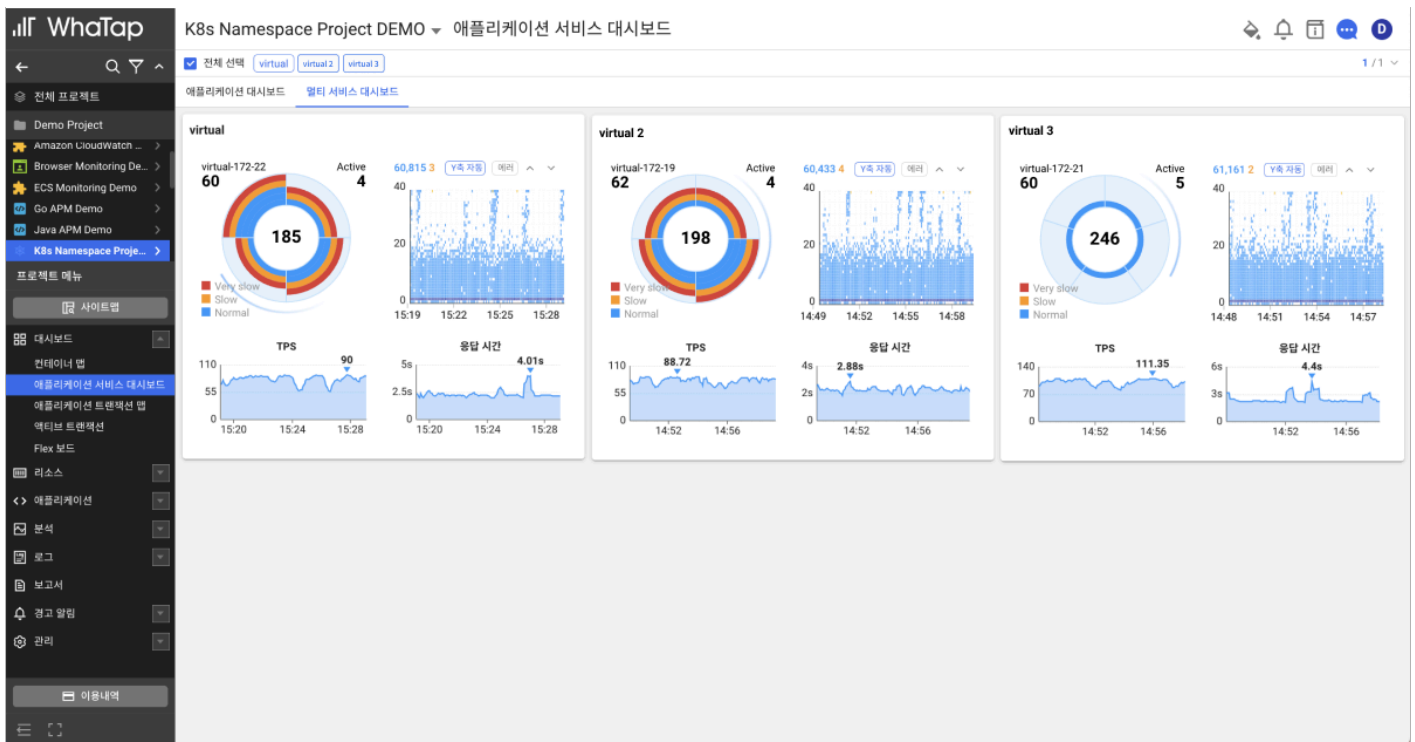
실시간 알림 확인하기

화면 오른쪽 위에  실시간 알림 버튼을 선택하면 최근 발생한 이벤트를 확인할 수 있습니다. 토크 메뉴를 클릭해 브라우저 알림을 켜거나 끌 수 있습니다.

❗ 화면 가장 위에 고정 메뉴 영역의 기본 요소들에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

멀티 서비스 대시보드

선택한 애플리케이션을 개별 카드 형태로 조회, 비교할 수 있습니다.

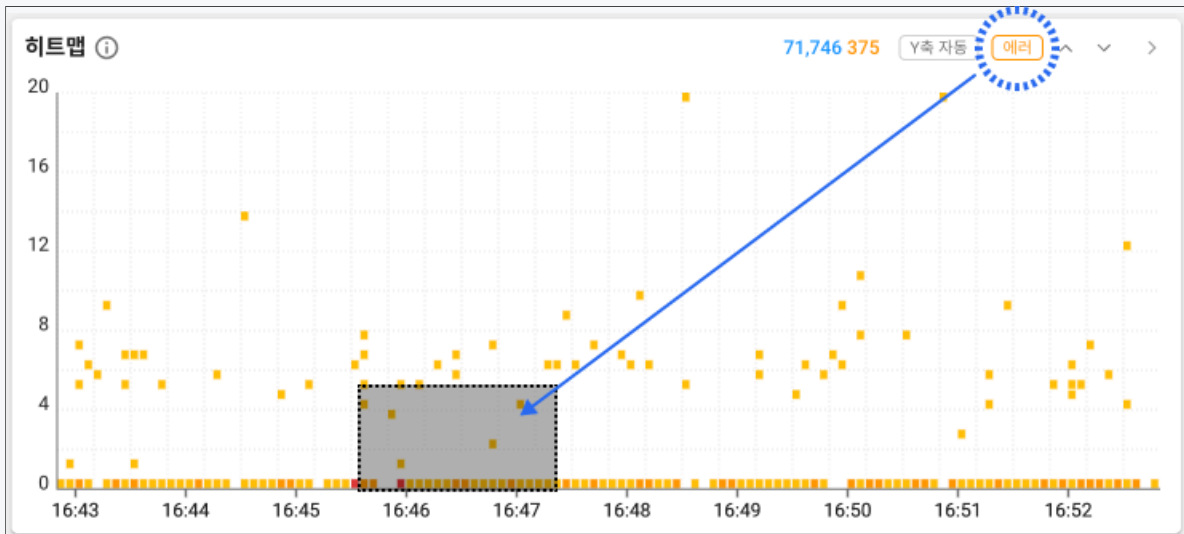


히트맵 - 트레이스 분석

애플리케이션 대시보드의 히트맵 위젯을 통해 트레이스 분석 창을 열어 트랜잭션 목록과 각 트랜잭션 하위의 스텝 정보를 한 번에 확인할 수 있습니다. 히트맵 위젯에서 원하는 영역을 드래그하세요. 선택한 영역의 트랜잭션 목록과 하위 스텝 정보를 포함하는 트레이스 분석 창이 나타납니다.

ⓘ 지난 2023년 7월 25일, 릴리스된 트레이스 분석 기능의 변경 사항을 확인하려면 [다음 문서](#)를 참조하세요.

✔ 에러가 발생한 트랜잭션 정보만 분류해 확인하려면 애플리케이션 대시보드의 히트맵 위젯의 오른쪽 위에 에러 버튼을 선택한 다음 차트 영역을 드래그하세요.



기본 화면 안내

트레이스 분석 1000건 조회되었습니다. ①

전체 ② [필터링 설정] [TXT] 검색어를 입력하세요

No	에이전트 명 (oname)	트랜잭션	경과 시간	시작 시간	종료 시간	HTTP 호출 ...	HTTP 호출 시간 (...)	SQL 시간	SQL 건수	DB 연결 시간 (...)	SQL 패치 건수	에러 메시지	클라이언트 IP	WClientID	멀티 트랜잭...
1	demo-8100	/edu/load/divis... A	13,919	23/10/06 08:27:15.989	23/10/06 08:27:29.908	9	13,154	714	6	20	1,918		246.159.226.226	6299760544220	
2	demo-8103	/order/write/de... A	13,271	23/10/06 08:27:18.842	23/10/06 08:27:32.113	10	12,285	790	5	192	0		53.128.58.29	-6966890863916	
3	demo-8105	/edu/kill/divisio... A	12,351	23/10/06 08:27:16.740	23/10/06 08:27:29.091	10	11,836	495	2	3	143		163.234.79.106	79916051556534	
4	demo-8100	/order/write/un... A	12,233	23/10/06 08:27:18.904	23/10/06 08:27:31.137	9	11,635	398	8	184	92		113.217.26.61	18123471058131	
5	demo-8104	/order/pickup/d... A	11,993	23/10/06 08:27:14.781	23/10/06 08:27:26.774	10	11,921	67	3	3	0		165.239.32.206	-8541053171176	
6	demo-8102	/order/remove/... A	11,648	23/10/06 08:27:22.493	23/10/06 08:27:34.141	9	11,365	263	4	4	610		101.243.208.67	1849935236386	
7	demo-8105	/sales/read/uni... A	11,505	23/10/06 08:27:19.740	23/10/06 08:27:31.245	9	11,034	265	4	203	0		130.135.32.247	6289584611395	
8	demo-8101	/order/pickup/e... A	11,495	23/10/06 08:27:18.073	23/10/06 08:27:29.568	9	10,858	426	4	196	1,266		160.106.28.10	-5449598593135	
9	demo-8104	/order/read/em... A	11,476	23/10/06 08:27:14.258	23/10/06 08:27:25.734	8	10,572	838	7	43	1,645		62.26.92.37	11623328420977	
10	demo-8100	/edu/write/divi... A	11,204	23/10/06 08:27:14.097	23/10/06 08:27:25.301	9	10,944	154	4	97	413		161.42.184.55	-601513188945	

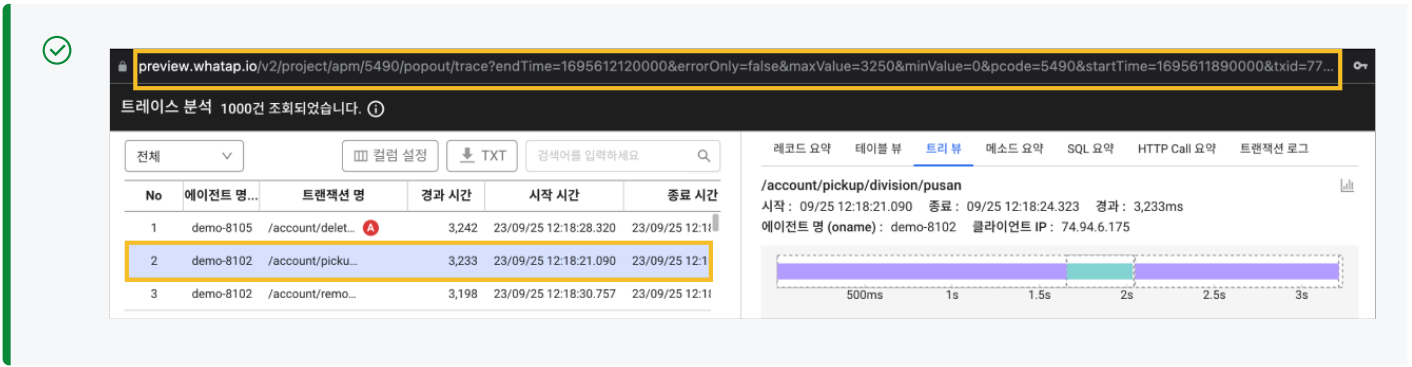
- ① **트랜잭션 조회 건수**: 히트맵 위젯에서 드래그한 영역의 트랜잭션 목록의 개수를 확인할 수 있습니다. 최대 1000건까지 조회할 수 있습니다.
- ② **트랜잭션 분류**: **A** **액티브 스택**과 **M** **멀티 트랜잭션**, 에러 항목을 구분하여 목록을 확인할 수 있습니다.
- ③ **트랜잭션 검색**: 트랜잭션 명과 에이전트 명, 에러 메시지를 기준으로 트랜잭션 목록을 검색할 수 있습니다.
- ④ **목록 정렬하기**: 테이블의 헤더 컬럼 항목을 클릭하면 선택한 항목을 기준으로 트랜잭션 목록을 정렬할 수 있습니다.
- 에러가 발생한 트랜잭션 항목은 목록에서 빨간색으로 표시됩니다.

트레이스 분석 92건 조회되었습니다. ①

전체 [필터링 설정] [TXT] 검색어를 입력하세요

No	에이전트 명 (oname)	트랜잭션	경과 시간	시작 시간	종료 시간	HTTP 호출 ...	HTTP 호출 시간 (...)	SQL 시간	SQL 건수	DB 연결 시간 (...)	SQL 패치 건수	에러 메시지	클라이언트 IP	WClientID	멀티 트랜잭...
1	demo-8102	/account/delet... A	9,637	23/10/06 08:45:29.188	23/10/06 08:45:38.825	7	8,802	648	3	184	0	Sql Exception	11.105.116.110	-5540024635227	
2	demo-8104	/edu/read/unit/... A	8,918	23/10/06 08:44:21.258	23/10/06 08:44:30.176	7	8,867	45	1	1	0	Sql Exception	53.199.161.32	22360232179385	
3	demo-8101	/product/create... A	8,384	23/10/06 08:45:57.515	23/10/06 08:46:05.899	7	8,167	193	3	4	709	Sql Exception	5.217.123.24	7443623688153	
4	demo-8102	/account/remo... A	7,810	23/10/06 08:45:48.766	23/10/06 08:45:56.576	6	7,145	651	2	2	220	Sql Exception	246.254.84.54	1849935236386	
5	demo-8103	/order/delete/u... A	7,762	23/10/06 08:45:16.257	23/10/06 08:45:24.019	6	7,290	458	2	3	0	Sql Exception	131.246.229.18	7124226905278	
6	demo-8104	/product/remov... A	7,289	23/10/06 08:46:19.886	23/10/06 08:46:27.175	6	7,281	3	1	1	0	Sql Exception	68.188.222.114	-1064563861026	

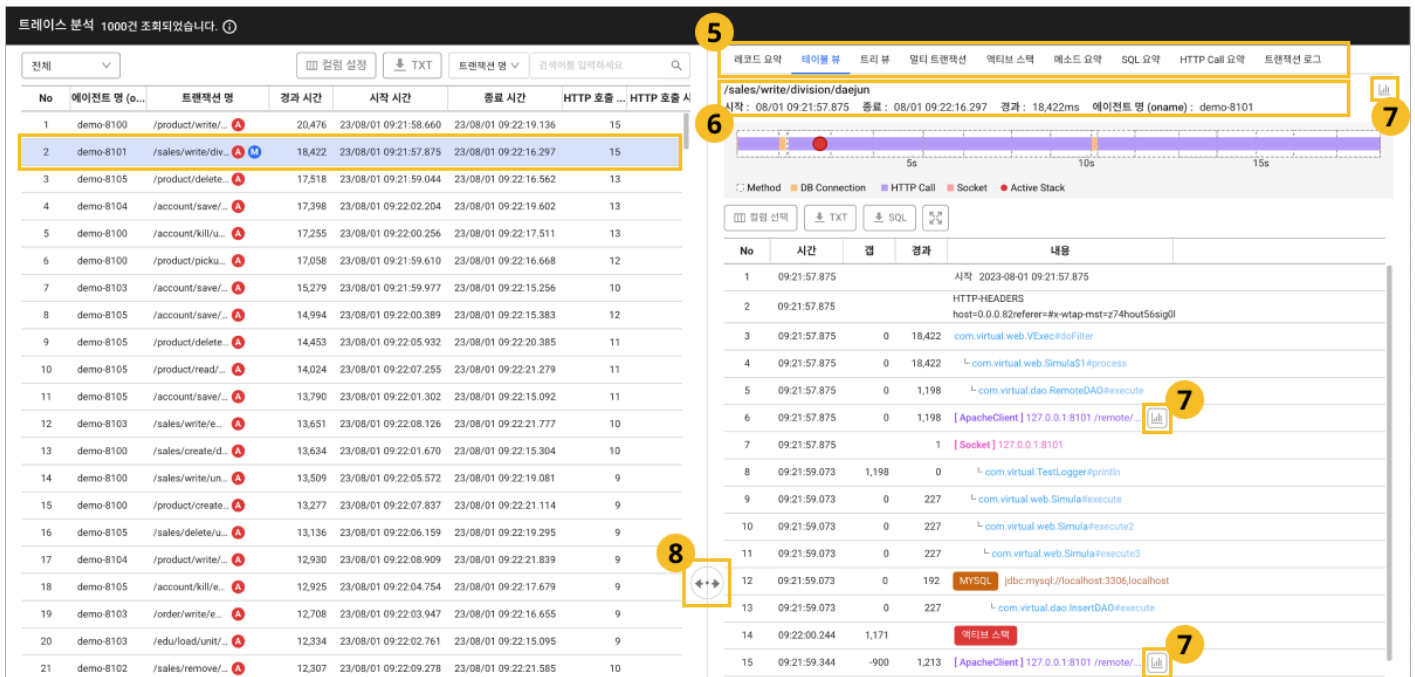
- ☑ 테이블에서 자주 보길 원하는 컬럼 항목을 앞으로 배치하거나 추가, 숨기기할 수 있습니다. 자세한 내용은 [다음 문서](#)를 참조하세요.
- 히트맵 위젯에서 다른 영역을 드래그하면 새로운 트레이스 분석 창을 불러올 수 있습니다. 기존에 열린 트레이스 분석 창과 비교해 확인할 수 있습니다.
- 현재 사용자가 보고 있는 트랜잭션 항목을 다른 사용자에게 공유할 수 있습니다. 트랜잭션 항목을 선택한 상태에서 트레이스 분석 창의 URL 주소를 복사해 다른 사용자에게 전달하세요. URL 주소를 전달 받은 사용자는 브라우저 주소 표시줄에 URL 주소를 입력하면 같은 항목을 빠르게 확인할 수 있습니다.





트레이스 분석하기




트랜잭션 목록에서 상세 내용을 확인할 항목을 선택하세요. 화면 오른쪽에 선택한 트랜잭션에 대한 상세 분석 내용을 표시합니다. 이 상태에서 다른 트랜잭션 항목을 선택하면 선택한 트랜잭션 항목에 대한 상세 분석 내용으로 변경됩니다.

와탭은 트랜잭션의 성능을 분석하기 위해 이름과 클라이언트 정보 등의 속성, 트랜잭션의 처리 성능 그리고 각 구간별 상세 수행 이력을 수집하고 보여줍니다.




- 5 영역에서는 스텝 정보를 다양한 형식으로 확인할 수 있습니다. 원하는 형식의 탭을 선택하세요.

- 6 영역에서는 트랜잭션에 대한 기본 정보를 확인할 수 있습니다. 왼쪽 목록의 컬럼 항목과 일치하는 정보입니다.
- 7 : 해당 URL 또는 각 수행 구간의 통계 데이터 창이 나타납니다. 통계 데이터 창에서 상세 버튼을 선택하면 통계 메뉴로 이동합니다.
- 8  왼쪽 또는 오른쪽으로 드래그해서 화면의 크기를 조절할 수 있습니다. 더 많은 정보를 보길 원할 때 이용하면 유용합니다.

-  통계 메뉴에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.
 -  또는  버튼을 클릭하면 트랜잭션 목록 또는 상세 분석 화면을 전체 크기로 변경할 수 있습니다.

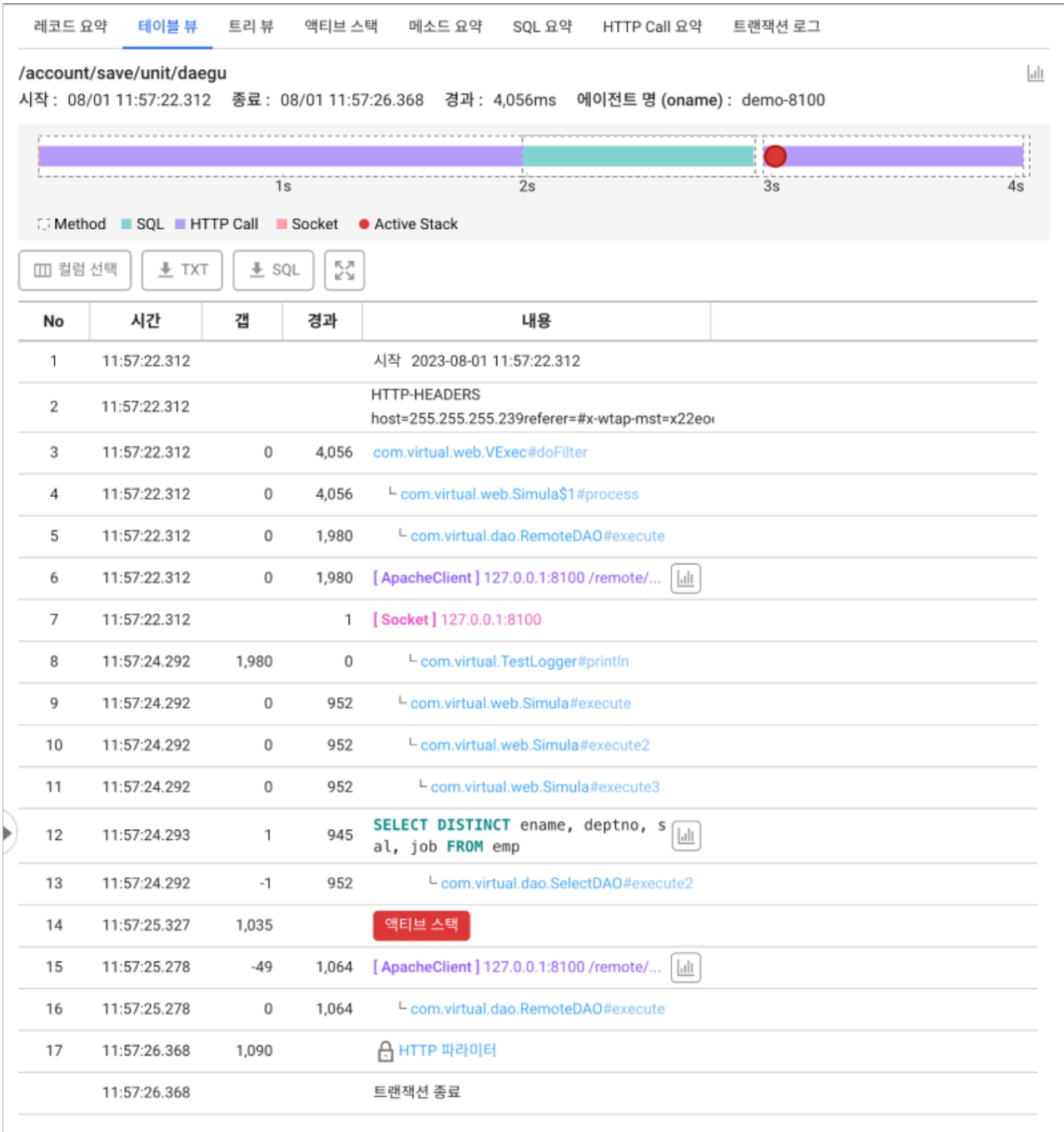
수집하는 스텝(Step)의 종류는 다음과 같습니다. 다이어그램과 텍스트 색상을 구분해 표시합니다.

- DB Connection 스텝 START-END
RDB에 대한 연결에 대한 성능을 포함합니다. 스텝 정보에는 이름, 응답시간, 에러를 포함합니다.
- SQL 스텝 START-END
JDBC SQL에 대한 성능을 포함합니다. 스텝 정보에는 연결 정보, SQL문, 에러가 포함되어 있습니다.
- HTTP Call 스텝 START-END
외부 http 서비스 호출에 대한 성능을 포함합니다. 스텝 정보에는 url, host, port, 응답시간, 에러가 포함됩니다.
- Message 스텝 ADD
트레이스를 수집하는 과정에서 비정형적인 모든 구간에 대한 이력을 수집할 때 메시지 스텝을 사용합니다.
- Socket 스텝 ADD
Socket 오픈을 표현하는 스텝입니다.
- Method 스텝 START-END
메소드 응답시간을 추적합니다.
- Active Stack 스텝 START-END
액티브 스텝에 대한 정보를 포함합니다. 별도 스레드가 생성하여 트레이스에 추가하는 방식으로 수집합니다.

-  와탭 모니터링 서비스에서 스텝(Step)은 스패ن(Span)과 같은 뜻으로 사용됩니다.
 - 지난 2023년 7월 25일, 릴리스된 [트레이스 분석](#) 기능의 변경 사항을 확인하려면 [다음 문서](#)를 참조하세요.

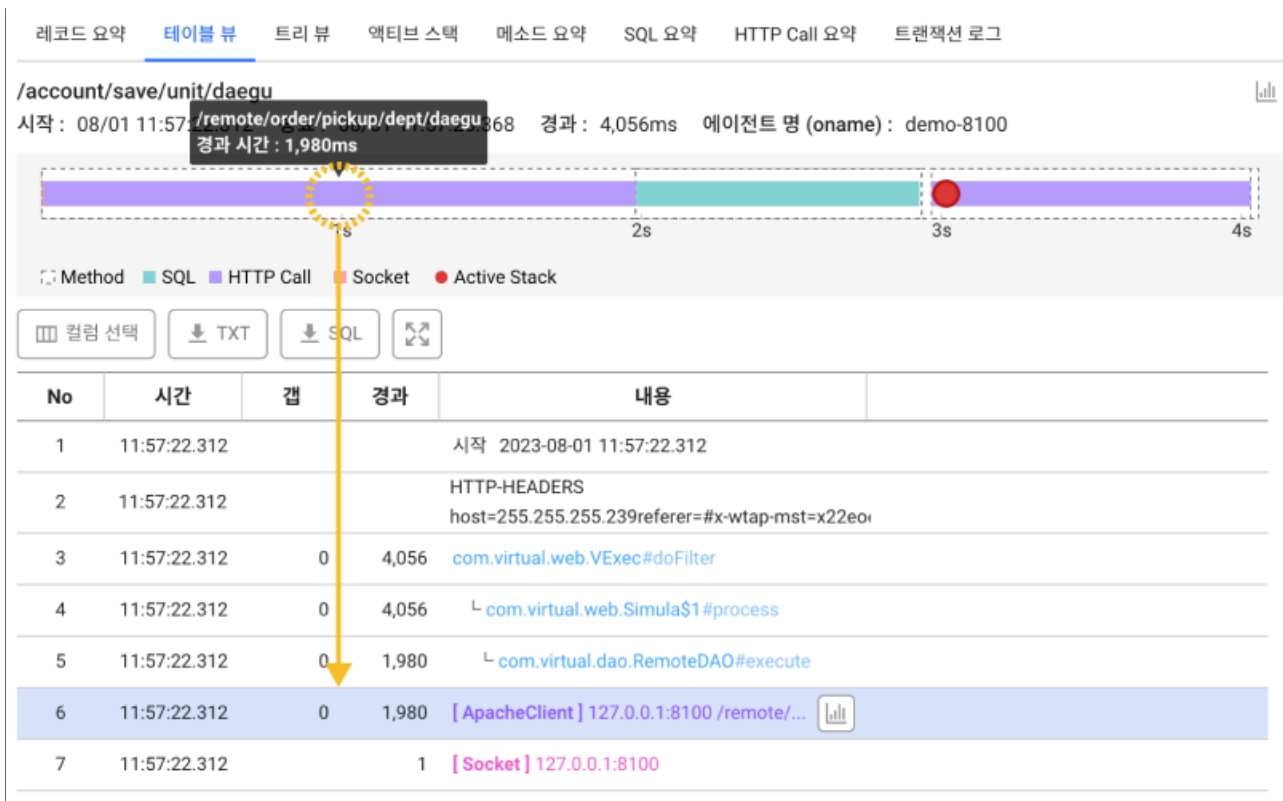
테이블 뷰

테이블 뷰 탭에서는 트랜잭션의 수행 과정을 시간의 순서대로 확인할 수 있습니다.



- 트랜잭션을 수행 구간별로 분류해 다이어그램을 통해 확인할 수 있습니다. 전체 경과 시간 중 각 구간별 소요된 시간, 가장 오래 소요된 구간을 빠르게 파악할 수 있습니다.

- 다이어그램의 각 구간을 선택하면 해당 스텝이 위치한 테이블 목록으로 이동합니다.



- 빨간색 원(●)이 위치한 영역은 액티브 스택이 수집된 순간입니다. 빨간색 원을 선택하면 **액티브 스택** 버튼이 위치한 테이블 목록으로 이동할 수 있습니다. **액티브 스택** 버튼을 선택하면 해당 구간 동안 수행된 스텝 정보를 확인할 수 있습니다.

ⓘ ○ 트랜잭션 목록에서 **A** 아이콘이 표시된 항목에서 확인할 수 있습니다.
○ 액티브 스택에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

- **☰ 컬럼 선택**: 테이블 목록에 컬럼으로 메모리 누적 정보 및 CPU 누적 정보를 추가하거나 감출 수 있습니다. 컬럼 정보는 다음과 같습니다.
 - **No**: 스텝의 발생 순서
 - **시간**: 각 스텝의 시작 시각
 - **갭**: 해당 스텝의 시작 시각부터 다음 스텝으로 넘어가기까지 대기 시간, 외부 요인으로 지연될 경우 경과 시간과 차이가 있을 수 있습니다.
 - **경과**: 각 메소드 시작부터 종료까지 총 소요 시간
 - **내용**: 해당 스텝의 세부 수행 내용

- **↓ TXT**: 트랜잭션 기본 정보 및 구간별 수행 정보를 txt 형식의 파일로 다운로드할 수 있습니다.
- **↓ SQL**: 트랜잭션 기본 정보 및 SQL 수행 정보를 txt 형식의 파일로 다운로드할 수 있습니다.
- **🔍**: 트랜잭션 요약 정보와 다이어그램을 감추고 테이블 목록만 확인할 수 있습니다. **⌵** 버튼을 선택하면 감춰진 정보를 다시 표시합니다. 테이블 목록이 긴 경우 이 기능을 이용하면 유용합니다.
- SQL 스텝을 선택하면 파라미터를 조회할 수 있는 **SQL** 창이 나타납니다. 자세한 내용은 [다음 문서](#)를 참조하세요.

레코드 요약

레코드 요약	테이블 뷰	트리 뷰	메소드 요약	SQL 요약	HTTP Call 요약	트랜잭션 로그
/account/remove/dept/pusan 📄						
에이전트 명 (oname) : demo-8101			에이전트 ID (oid) : -877561626			
에이전트 그룹 명 : demo-okind-1			에이전트 그룹 ID : -628198688			
에이전트 서버 명 : node-1			에이전트 서버 ID : 334634079			
프로젝트 코드 : 5490						
<hr/>						
트랜잭션 ID : -3538973756714763020			시작 시간 : 23/10/06 08:48:11.706			
종료 시간 : 23/10/06 08:48:14.585			경과 시간 : 2,879ms			
HTTP 메소드 : GET			HTTP 호출 시간 : 2,409ms			
HTTP 호출 건수 : 2			DB 연결 시간 : 139ms			
SQL 시간 : 324ms			SQL 건수 : 2			
SQL 패치 시간 : 6ms			SQL 패치 건수 : 647			
<hr/>						
CPU 사용 시간 : 2ms			메모리 할당량 : 197,232byte			
<hr/>						
클라이언트 IP : 168.196.231.239			도메인 : 255.255.255.159			
WClientID : 4545076460797978934			국가 : BR			
도시 : Acailandia			운영체제 : Windows			
클라이언트 타입 : IE			클라이언트 명 : Other			
상태 : 200			Referer : http://www.test.com/account/load/employee/seoul			
원본 URL/트랜잭션 : /account/remove/dept/pusan						
유저 에이전트 : Mozilla/5.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4325) 📄						

트랜잭션의 기본 정보를 확인하려면 **레코드 요약** 탭을 선택하세요. 트랜잭션의 기본 정보 및 해당 트랜잭션이 수행된 에이전트 관련

정보, 메모리 할당 사용량, 클라이언트 관련 정보를 확인할 수 있습니다.

트랜잭션 속성

분류	속성	설명
에이전트	에이전트 명(oname)	에이전트 오브젝트 이름
	에이전트 ID (oid)	에이전트 오브젝트 아이디
	에이전트 그룹 명	okind 옵션으로 설정된 에이전트 그룹 이름
	에이전트 그룹 ID	에이전트 그룹 아이디
	에이전트 서버 명	ondoe 옵션으로 설정한 에이전트 서버 이름
	에이전트 서버 ID	에이전트 서버 아이디
	프로젝트 코드	와탭 모니터링 서비스에 등록된 프로젝트의 식별 코드
트랜잭션	트랜잭션 ID	트랜잭션 아이디
	시작 시간	트랜잭션 시작 시각
	종료 시간	트랜잭션 종료 시각
	경과 시간	트랜잭션 수행 시간
	HTTP 메소드	HTTP 메소드 - GET, POST, PUT, HEAD 등
	HTTP 호출 시간	외부 Http Call 시간
	HTTP 호출 건수	외부 HTTP Call 건수
	DB 연결 시간	DB와 연결된 시간
	SQL 시간	SQL 수행 시간

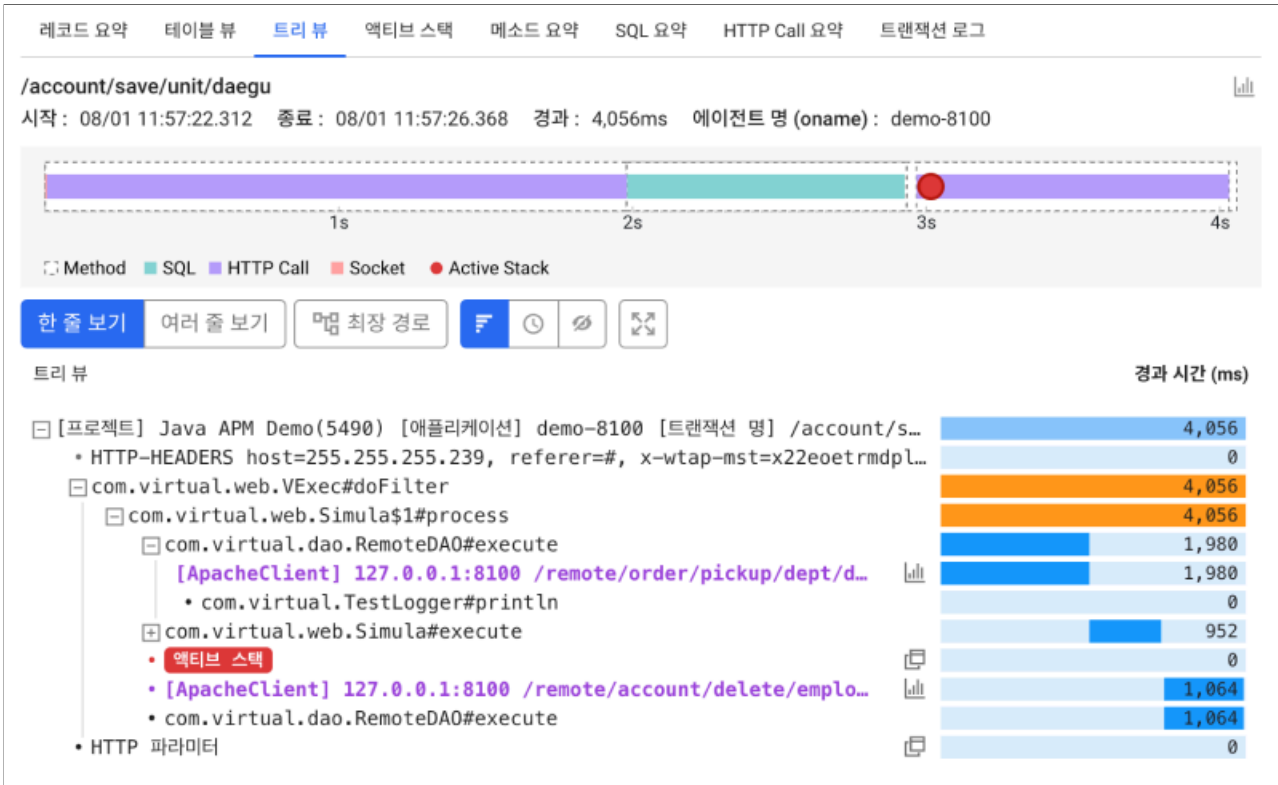
분류	속성	설명
	SQL 건수	SQL 수행 건수
	SQL 패치 시간	레코드를 조회하는 데 걸린 시간 (중첩이 발생하거나 타 로직이 포함될 수 있음)
	SQL 패치 건수	SQL에서 데이터를 조회한 레코드 건수
멀티 트랜잭션	CallerUrl	부모 트랜잭션의 트랜잭션 주소
	CallerPcode	부모 트랜잭션이 수집된 프로젝트 코드
	CallerOid	부모 트랜잭션의 에이전트 오브젝트 아이디
	CallerOName	부모 트랜잭션의 에이전트 이름
	CallerOkind	부모 트랜잭션의 에이전트 그룹 ID
	CallerOkindName	부모 트랜잭션의 에이전트 그룹 이름
자원	CPU 사용 시간	트랜잭션이 사용한 CPU 사용량
	메모리 할당량	트랜잭션이 사용한 메모리 사용량
클라이언트	클라이언트 IP	클라이언트의 IP 주소
	도메인	클라이언트가 접속한 IP 주소에 지정된 인터넷 주소
	WClientID	클라이언트의 ID
	국가	클라이언트의 국가 정보
	도시	클라이언트의 국가 내 도시 정보
	운영체제	브라우저가 실행되는 운영 체제 환경
	클라이언트 타입	클라이언트가 이용한 브라우저 종류

분류	속성	설명
	클라이언트 명	클라이언트가 이용한 기기 이름
	상태	클라이언트의 요청에 대한 HTTP 상태 코드
	Referer	클라이언트가 브라우저에서 이전에 이용한 페이지 주소 또는 유입 경로
	원본 URL/트랜잭션	트랜잭션 이름과 원본 URL의 조합 데이터 원본 URL이 있을 경우 원본 URL 표시, 원본 URL이 없을 경우 트랜잭션 표시
	유저 에이전트	클라이언트 관련 정보를 추출하는데 이용하는 브라우저 정보

- ⓘ 애플리케이션 종류나 설정, 스텝의 종류에 따라 수집하는 정보는 달라질 수 있습니다.
- 와탭은 클라이언트와 관련한 정보를 기본 저장합니다. 사용자 데이터 수집과 관련한 에이전트 설정에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

트리 뷰

트랜잭션 수행 과정을 트리 형식으로 확인하려면 **트리뷰** 탭을 선택하세요. 각 트랜잭션과 그에 속한 트race의 세부 정보, 트race의 시작 시간 및 소요 시간, 호출 관계를 확인할 수 있습니다. 다이어그램의 각 구간을 선택하면 해당 스텝이 위치한 리뷰로 이동합니다.




- **한 줄 보기:** 각 구간 별 수행 정보에 표시된 텍스트를 한 줄로 표시해 트리 형식을 간격하게 정리할 수 있습니다.
- **여러 줄 보기:** 각 구간 별 수행 정보에 표시된 텍스트를 줄바꿈해 모두 표시합니다.
- **최장 경로:** 가장 긴 경로로 이동할 수 있습니다.
- **시간바 표시:** 경과 시간을 막대 형식의 차트로 표시합니다.
- **시간 표시:** 각 구간별 타임 스탬프, 갭, 경과 시간을 텍스트 형식으로 표시합니다.
 - 8초 이상: **초과 지연** 상태로 **빨간색**으로 표현합니다.
 - 3초 이상 8초 미만: **지연** 상태로 **주황색**으로 표현합니다.
 - 3초 미만: **정상** 상태로 **파란색**으로 표현합니다.
- **시간 숨기기:** 시간 정보를 숨깁니다.
- **트랜잭션 요약 정보와 다이어그램을 감추고 트리뷰만 확인할 수 있습니다.** 버튼을 선택하면 감춰진 정보를 다시 표시합니다. 트리뷰 목록이 긴 경우 이 기능을 이용하면 유용합니다.
- **SQL 변수와 HTTP 쿼리를 조회할 수 있는 창이 나타납니다.** 자세한 내용은 [다음 문서](#)를 참조하세요.

- ① 시작 및 소요 시간의 경우 트랜잭션 호출 환경에 따라 발생하는 시차를 상위 트랜잭션 내 트레이스와 매핑을 통해 보정하여 표현하기 때문에 실제 수집된 시간 데이터와 차이가 발생할 수 있습니다.

액티브 스택

레코드 요약 테이블 뷰 트리 뷰 **액티브 스택** 메소드 요약 SQL 요약 HTTP Call 요약 트랜잭션 로그

/account/save/unit/daegu 

시작: 08/01 11:57:22.312 종료: 08/01 11:57:26.368 경과: 4,056ms 에이전트 명 (oname): demo-8100

액티브 스택 (1)

No	시간	
14	11:57:25.327	<pre> java.base@17.0.7/sun.nio.ch.SocketDispatcher.read0(Native Method) java.base@17.0.7/sun.nio.ch.SocketDispatcher.read(SocketDispatcher.java:47) java.base@17.0.7/sun.nio.ch.NioSocketImpl.tryRead(NioSocketImpl.java:261) java.base@17.0.7/sun.nio.ch.NioSocketImpl.implRead(NioSocketImpl.java:312) java.base@17.0.7/sun.nio.ch.NioSocketImpl.read(NioSocketImpl.java:350) java.base@17.0.7/sun.nio.ch.NioSocketImpl\$1.read(NioSocketImpl.java:803) java.base@17.0.7/java.net.Socket\$SocketInputStream.read(Socket.java:966) app//org.apache.http.impl.io.SessionInputBufferImpl.streamRead(SessionInputBufferImpl.java:139) app//org.apache.http.impl.io.SessionInputBufferImpl.fillBuffer(SessionInputBufferImpl.java:155) app//org.apache.http.impl.io.SessionInputBufferImpl.readLine(SessionInputBufferImpl.java:284) app//org.apache.http.impl.conn.DefaultHttpResponseParser.parseHead(DefaultHttpResponseParser.java:140) app//org.apache.http.impl.conn.DefaultHttpResponseParser.parseHead(DefaultHttpResponseParser.java:57) app//org.apache.http.impl.io.AbstractMessageParser.parse(AbstractMessageParser.java:261) </pre>

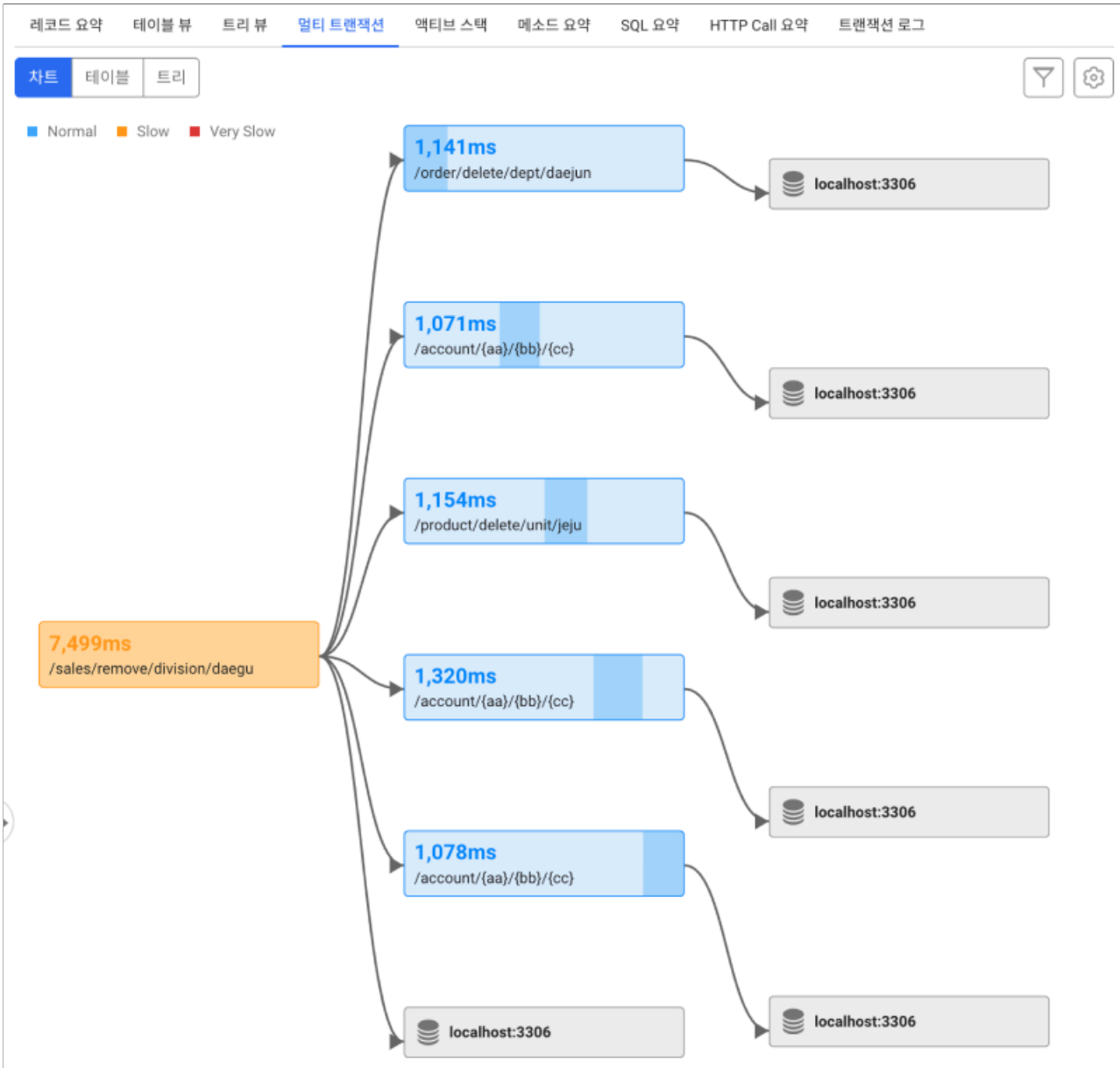
액티브 스택을 수집한 구간과 시간, 관련한 클래스, 메소드 정보를 확인하려면 **액티브 스택** 탭을 선택하세요.


- ① • 트랜잭션 목록에서 **A** 아이콘이 표시된 항목에서 확인할 수 있습니다.
 - 액티브 스택에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

멀티 트랜잭션

멀티 트랜잭션은 다른 에이전트나 프로젝트와의 연관된 트랜잭션을 의미합니다. [멀티 트랜잭션](#) 탭에서는 와탭 모니터링 서비스에 등록된 애플리케이션 간의 호출 관계를 확인할 수 있습니다.

- ✔ 멀티 트랜잭션을 추적하려면 [관리 > 에이전트 설정](#) 메뉴에서 `mtrace_enabled` 옵션을 `true` 로 설정하세요. 에이전트 설정에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.



- **차트**: 각 트랜잭션의 호출 관계를 플로우 차트 형식으로 제공합니다. 트랜잭션 노드를 선택하면 해당 트랜잭션 노드에 대한 트레이스 분석 정보를 확인할 수 있습니다. 차트 모드에서는 마우스를 이용해 원하는 위치로 이동하거나 스크롤을 통해서 확대, 축소할 수 있습니다.
 - **차트 뷰 설정**: 차트에 표시할 요소를 표시하거나 숨길 수 있습니다.
- **테이블**: 테이블 형식으로 멀티 트랜잭션 내에 포함된 각 트랜잭션 별 정보를 확인할 수 있습니다.  컬럼 선택 아이콘을 선택해

테이블 헤더 컬럼을 편집할 수 있습니다. 각 트랜잭션 항목을 선택하면 트레이스 분석 정보를 확인할 수 있습니다.

- **트리**: 트리 형식으로 트랜잭션 간의 호출 관계를 파악할 수 있습니다. 관련한 부가 기능은 **트리뷰** 탭의 기능과 같습니다.
- **프로젝트 선택**: 차트에 표시될 프로젝트를 선택하거나 해제할 수 있습니다.

- ⓘ • 트랜잭션 목록에서 **M** 아이콘이 표시된 항목에서 확인할 수 있습니다.
 - 멀티 트랜잭션에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

메소드 요약

레코드 요약 테이블 뷰 트리 뷰 액티브 스택 메소드 요약 SQL 요약 HTTP Call 요약 트랜잭션 로그							
/account/save/unit/daegu 📊							
No	클래스	메소드	파라미터/리턴	건수	합계 (ms)	최대 (ms)	
1	com.virtual.dao.Remot...	execute	()V	2	3,044	1,980	
2	com.virtual.web.VExec	doFilter	(Ljavax/servlet/Servlet...	1	4,056	4,056	
3	com.virtual.web.Simul...	process	()V	1	4,056	4,056	
4	com.virtual.TestLogger	println	(Ljava/lang/String;)V	1	0	0	
5	com.virtual.web.Simula	execute	()V	1	952	952	
6	com.virtual.web.Simula	execute2	()V	1	952	952	
7	com.virtual.web.Simula	execute3	()V	1	952	952	
8	com.virtual.dao.Select...	execute2	()V	1	952	952	

메소드 정보만을 확인하려면 **메소드 요약** 탭을 선택하세요. 에이전트에 추적이 설정된 메소드 이름과 소요 시간을 표시합니다. 불필요한 로직이 반복 실행되는 경우, 낮은 건수에 비해 실행 시간이 오래 걸리는 경우를 파악해 메소드 로직 개선을 위한 분석 정보로 활용할 수 있습니다.

- ⓘ 메소드(method)와 관련한 에이전트 설정 옵션에 대한 자세한 내용은 [다음 문서](#)를 참조하세요. 이용 중인 상품에 따라 메소드 추적 지원 여부는 다를 수 있습니다.

SQL 요약

레코드 요약 테이블 뷰 트리 뷰 액티브 스택 메소드 요약 SQL 요약 HTTP Call 요약 트랜잭션 로그						
/account/save/unit/daegu 						
No	데이터베이스	SQL	건수	합계 (ms)	최대 (ms)	
1	jdbc:mysql://localhost:...	SELECT DISTINCT ena...	1	945	945	

SQL문에 대한 정보를 확인하려면 [SQL 요약](#) 탭을 선택하세요. 불필요하게 반복 실행되는 경우, 낮은 건수에 비해 실행 시간이 오래 걸리는 경우를 파악해 쿼리 성능 개선을 위한 분석 정보로 활용할 수 있습니다.

HTTP Call 요약

레코드 요약 테이블 뷰 트리 뷰 액티브 스택 메소드 요약 SQL 요약 HTTP Call 요약 트랜잭션 로그						
/account/save/unit/daegu 						
No	원격 서버	URL	건수	합계 (ms)	최대 (ms)	
1	127.0.0.1:8100	/remote/order/pickup/...	1	1,980	1,980	
2	127.0.0.1:8100	/remote/account/delet...	1	1,064	1,064	

HTTP 호출의 호출 건수, 합계 시간, 평균 시간 등을 확인하려면 [HTTP Call 요약](#) 탭을 선택하세요. 불필요한 외부 호출이 반복 실행되는 경우, 낮은 건수에 비해 실행 시간이 오래 걸리는 경우를 파악해 트랜잭션 지연 요인이 외부인지 내부인지를 파악하는 분석 정보로 활용할 수 있습니다.

트랜잭션 로그

트랜잭션과 관련한 로그 정보를 확인하려면 [트랜잭션 로그](#) 탭을 선택하세요.

레코드 요약 테이블 뷰 트리 뷰 액티브 스택 메소드 요약 SQL 요약 HTTP Call 요약 **트랜잭션 로그**

/account/save/unit/daegu

키워드(를) 입력해주세요 🔍 🏠 ⚙️

▶	oname	타임스탬프	로그
▶	demo-8100	2023-08-01 11:57:24.293	@txid 5501194250256405697 pcode 5490 oname demo-8100 onodeName node-0 http://127.0.0.1:8100/remote/order/pickup/dept/daegu status=200
▶	demo-8100	2023-08-01 11:57:25.245	@txid 5501194250256405697 pcode 5490 oname demo-8100 onodeName node-0 select distinct ename, deptno, sal, job from emp
▶	demo-8100	2023-08-01 11:57:25.279	@txid 5501194250256405697 pcode 5490 oname demo-8100 onodeName node-0 select productmodelid, name
▶	demo-8100	2023-08-01 11:57:26.342	@txid 5501194250256405697 pcode 5490 oname demo-8100 onodeName node-0 http://127.0.0.1:8100/remote/account/delete/employee/pusan status=20
▶	demo-8100	2023-08-01 11:57:26.369	@txid 5501194250256405697 pcode 5490 oname demo-8100 onodeName node-0 select ename, sal+1000 from emp

- **키워드 검색:** 키워드 검색란에 검색하려는 텍스트를 입력한 다음 엔터 키를 입력하거나 🔍 버튼을 선택하세요. 입력한 텍스트와 일치하는 키워드를 하이라이트 표시합니다.
- 🏠 **키워드 하이라이트:** 키워드와 색상을 설정하면 자동으로 로그 목록에서 키워드와 일치하는 텍스트를 하이라이트 표시합니다.
- ⚙️ **테이블 설정:** 로그 목록 테이블의 헤더 컬럼을 설정할 수 있습니다.

ⓘ • **트랜잭션 로그** 탭을 활성화하려면 로그와 관련한 에이전트 설정 옵션을 적용해야 합니다. 로그 설정에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

- **트랜잭션 로그** 탭은 **로그 조회** 권한이 가진 멤버만 진입할 수 있습니다. 멤버 권한에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

키워드 하이라이트 설정하기

1. 하이라이트 표시할 키워드를 추가하려면 🏠 버튼을 선택하세요.



2. 원하는 키워드를 입력하세요.
3. 색상 아이콘을 선택한 다음 원하는 색상을 선택하세요.
4. 엔터를 입력하세요.

추가한 키워드가 목록에 생성됩니다.

키워드 하이라이트 삭제하기

하이라이트 목록에서 이용하지 않는 항목을 삭제하려면 — 버튼을 선택하세요.

테이블 컬럼 표시하거나 숨기기

1. 테이블 오른쪽 위에 ⚙ 버튼을 선택하세요.
2. 드롭다운 메뉴에서 **컬럼 설정**을 선택하세요.
3. **컬럼 설정** 창이 나타나면 **컬럼 추가** 섹션에서 원하는 항목을 선택하거나 선택을 해제하세요.
4. 모든 설정을 완료한 다음 **확인** 버튼을 선택하세요.

설정된 내용이 테이블에 적용됩니다.

테이블 컬럼 순서 바꾸기

1. 테이블 오른쪽 위에 ⚙ 버튼을 선택하세요.
2. 드롭다운 메뉴에서 **컬럼 설정**을 선택하세요.

3. 컬럼 설정 창이 나타나면 컬럼 순서 설정 섹션에서 원하는 항목을 드래그해서 순서를 변경하세요.
4. 모든 설정을 완료한 다음 확인 버튼을 선택하세요.

설정된 내용이 테이블에 적용됩니다.

테이블 콘텐츠 표시 설정

1. 테이블 오른쪽 위에 ⚙️ 버튼을 선택하세요.
2. 드롭다운 메뉴에서 테이블 콘텐츠 설정을 선택하세요.
3. 다음 옵션을 설정하세요.
 - 콘텐츠 표시 여부: 테이블 목록에 표시하는 콘텐츠 표시 여부를 선택하세요.
 - 태그 관리: 테이블 목록에 표시하고 싶은 태그를 선택할 수 있습니다. 보이지 않도록 설정하거나 순서를 변경할 수 있습니다.
4. 모든 설정을 완료한 다음 확인 버튼을 선택하세요.

설정된 내용이 테이블에 적용됩니다.

부가 기능

HTTP 파라미터 조회

테이블 뷰 탭에서 해당 트랜잭션의 HTTP 파라미터를 조회할 수 있습니다.

1. 페이지 아래로 스크롤해 트랜잭션 수행의 가장 마지막 단계로 이동하세요.
2. ☰ HTTP 파라미터 항목을 선택하세요.
3. HTTP-PARAMETERS 창이 나타나면 비밀번호 버튼을 선택하세요.
4. 설정한 Param Key를 입력하세요.

가려진 매개 변수를 확인할 수 있습니다.

- ① • HTTP 파라미터와 관련한 에이전트 설정은 [다음 문서](#)를 참조하세요.
 - 비밀번호: 복호화된 파라미터 값을 확인할 수 있습니다. 비밀번호는 `WHATAP_HOME/security.conf(paramkey.txt)` 파일 내 6자리 문자열입니다. 다른 문자열로 변경 가능합니다. SQL 변수 조회, HTTP 쿼리 조회, Thread 중지 필요합니다.

SQL 파라미터 조회

테이블 뷰 탭에서 SQL 스텝을 선택하거나 트리 뷰 탭에서  버튼을 선택하세요. 파라미터를 조회할 수 있는 SQL 창이 나타납니다.

SQL
  

연결 URL jdbc:mysql://localhost:3306,localhost:3310/fake@BasicDataSource

CPU 누적 3ms

메모리 누적 329,752byte

쿼리 (Hash: -2057375474)


SID 107


```
insert into
  dept
values(
  #, '#', '#')
```

치환 변수

파라메타가 존재하지 않거나 복호화에 실패하였습니다.
복호화를 위하여 에이전트 설치 경로의 [paramkey.txt]에서 비밀번호를 확인해주세요.

- **비밀번호:** 복호화된 파라미터 값을 확인할 수 있습니다. 비밀번호는 `WHATAP_HOME` /`paramkey.txt` 파일 내 6자리 문자열입니다. 다른 문자열로 변경 가능합니다.

 `paramkey.txt` 내 키는 SQL 변수 조회, HTTP 쿼리 조회, Thread 중지에도 필요합니다.

- **SQL 포매팅:** SQL 문장을 들여쓰기 및 포매팅하여 가독성을 높일 수 있습니다.
- : SQL 문장을 클립보드에 복사할 수 있습니다.
- **통계:** 통계/보고서 > 통계 메뉴의 SQL 탭으로 이동합니다.

- ✓ SQL 변수와 HTTP 쿼리를 조회하려면 다음 옵션을 에이전트 설정에 추가하세요. DB, SQL과 관련한 에이전트 설정에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

whatap.conf

SQL 파라미터 조회 옵션: 옵션이 적용되면 SQL 파라미터를 암호화하여 수집합니다.

profile_sql_param_enabled=true

HTTP 파라미터 조회 옵션: 옵션이 적용되면 HTTP 쿼리 파라미터를 암호화하여 수집합니다.



profile_http_parameter_enabled=true

테이블 컬럼 설정하기

트랜잭션 목록에서 테이블 헤더 컬럼을 감추거나 원하는 항목을 추가할 수 있습니다. 컬럼 순서를 변경할 수도 있습니다. **▣ 컬럼 설정** 버튼을 선택하세요.

컬럼 설정
✕

표시할 컬럼을 선택해주세요

1

전체 선택 초기화

- 경과 시간
- 국가
- 도메인
- 도시
- 로그인 ID
- 멀티 트랜잭션 ID
- 메모리 할당량
- 시작 시간
- 에러 단계
- 에러 메시지
- 에이전트 그룹 명
- 에이전트 그룹 ID
- 에이전트 명 (oname)
- 에이전트 서버 명
- 에이전트 서버 ID
- 에이전트 ID (oid)
- 운영 체제
- 원본 URL

3 컬럼명 검색

2 표시 항목(18)

No

- || 에이전트 명 (oname) ✕
- || 트랜잭션 ✕
- || 경과 시간 ✕
- || 시작 시간 YY/MM/DD 00:00:00 ▾ ✕
- || 종료 시간 YY/MM/DD 00:00:00 ▾ ✕
- || HTTP 호출 건수 ✕
- || HTTP 호출 시간 ✕
- || SQL 시간 ✕
- || SQL 건수 ✕
- || DB 연결 시간 ✕
- || SQL 패치 건수 ✕
- || 에러 메시지 ✕
- || 클라이언트 IP ✕

취소 확인

- ① • 설정을 완료한 다음에는 **확인** 버튼을 선택해야 설정 사항이 테이블에 반영됩니다.

- ③ 검색란에 텍스트를 입력해 원하는 컬럼 항목을 검색할 수 있습니다. 입력한 텍스트와 매칭되는 컬럼 항목만 표시됩니다.

컬럼 추가하기

- 1 목록에서 테이블 헤더 컬럼으로 추가할 항목을 선택하세요. 모든 항목을 추가하려면 **전체 선택**을 선택하세요.

컬럼 삭제하기

- 1 목록에서 삭제할 컬럼 항목의 체크 박스를 선택 해제하세요. 또는 2 목록에서 삭제할 항목의 오른쪽에 **×** 버튼을 선택하세요.

컬럼 순서 변경하기

- 2 목록에 순서를 변경할 항목을 드래그해서 원하는 위치로 이동할 수 있습니다.

설정 사항 초기화하기

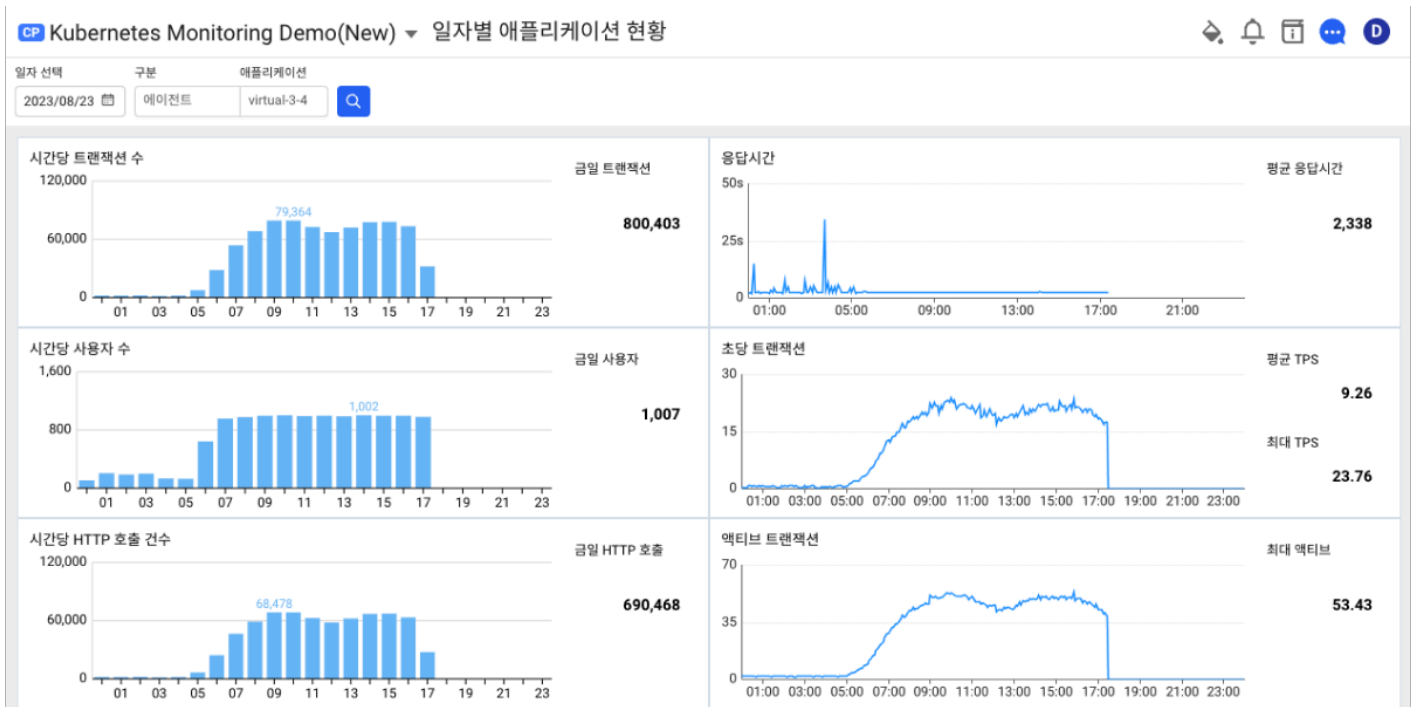
변경 사항은 모두 취소하고 초기화하려면 **초기화** 버튼을 선택하세요.

조회 데이터 다운로드

테이블에 출력된 데이터를 txt 파일 형식으로 다운로드하려면 **↓ TXT** 버튼을 선택하세요.

일자별 애플리케이션 현황

홈 화면 > 프로젝트 선택 > 애플리케이션 > 분석 > 일자별 애플리케이션 현황



일자별 애플리케이션 상태를 확인할 수 있습니다. 검색 조건은 시작일, 구분, 애플리케이션으로 구분 조건은 다음과 같습니다.

분류	설명
에이전트	컨테이너 내부에 설치된 애플리케이션 에이전트 이름
에이전트 종류	에이전트가 그룹화된 에이전트 종류 (OKIND)
에이전트 서버	에이전트가 실행 중인 쿠버네티스 노드 이름

제공하는 차트는 다음과 같습니다.

- 시간당 트랜잭션 수

- 시간당 사용자 수
- 시간당 HTTP 호출 건수
- 시간당 SQL 건수
- 시간당 SQL 패치 건수
- 응답시간
- 초당 트랜잭션
- 액티브 트랜잭션
- 5분 합계 에러 건수
- 에러율

트랜잭션 검색

홈 화면 > 프로젝트 선택 > 애플리케이션 > 분석 > 트랜잭션 검색

트랜잭션 검색 메뉴를 통해 특정 기간 동안 각 트랜잭션의 성능 속성에 대한 통계와 실행 상태를 검색할 수 있습니다. 모든 트랜잭션을 확인하고 정상 및 에러 상태를 빠르게 확인할 수 있습니다. 제공되는 정보를 통해 어떤 트랜잭션의 사용량이 많고 리소스를 많이 소비하는지 알 수 있습니다. 트랜잭션의 속성에 따라 다양하게 필터링할 수 있어 특정 조건의 트랜잭션을 찾는 데 유용합니다.

The screenshot displays the '트랜잭션 검색' (Transaction Search) interface. At the top, there's a breadcrumb 'Kubernetes Monitoring Demo(EKS) > 트랜잭션 검색'. Below it, a search bar shows a time range: '2024/03/07 00:00 ~ 2024/03/07 23:59'. A '컨테이너' (Container) dropdown is set to '전체 선택'. A row of filter buttons includes: '+ 경과 시간', '+ 에러 메시지', '+ HTTP 호출 건수', '+ SQL 건수', '+ SQL 패치 건수', '+ 트랜잭션', '+ 클라이언트 IP', '+ 도메인', '+ 유저 에이전트', '+ Referer', '+ 트랜잭션 ID', '+ 멀티 트랜잭션 ID', '+ WClientID', and '+ 로그인 ID'. There's also an '에러만 보기' (Show errors only) checkbox and a 'CSV' button. The main table area has a '컬럼 선택' (Column selection) button. The table header includes: '에이전트명 (oname)', '시작 시간', '종료 시간', '경과 시간', '트랜잭션', 'CPU 사용 ...', 'HTTP 호출 ...', 'SQL 건수', and 'SQL 패치 건...'. A central message box says: '상세 검색을 위해 필터를 이용해주세요. 여러개의 검색 조건에 의한 데이터 탐색 방식을 지원합니다. 검색 범위가 지나치게 큰 경우 데이터 추출에 제한이 있습니다.'

① 영역에서 조회 시간을 설정 후 **컨테이너** 옵션에서 조회 대상을 선택하세요. **🔍** 버튼을 선택하면 검색 결과가 테이블에 표시됩니다. ② 영역의 결과 목록에서 원하는 항목을 선택하면 **트랜잭션 정보** 창이 나타납니다. **트랜잭션 정보** 창을 활용한 트랜잭션 트레이스 상세 분석에 관한 자세한 내용은 [다음 문서](#)를 참조하세요.


- ① • 에러 상태의 트랜잭션만 검색하려면 **에러만 보기**를 클릭한 다음 **🔍** 버튼을 선택하세요.
- 검색 조건에 따라 응답 지연이 발생할 수 있기 때문에 검색 시간을 가급적 짧게 설정할 것을 권고합니다.

컬럼 선택

② 영역의 **☰ 컬럼 선택** 버튼 클릭 시 다음과 같이 컬럼 선택 창이 나타납니다. 원하는 컬럼을 적절하게 선택하세요.

전체 선택 전체 해제 초기화(기본값)							×
<input type="checkbox"/> 에이전트	<input type="checkbox"/> 클라이언트	<input type="checkbox"/> 에러	<input type="checkbox"/> HTTP 호출	<input type="checkbox"/> 리소스	<input type="checkbox"/> SQL	<input type="checkbox"/> 트랜잭션	
<input type="checkbox"/> 컨테이너 ID	<input type="checkbox"/> 도시	<input type="checkbox"/> 에러 클래스	<input checked="" type="checkbox"/> HTTP 호출 건수	<input checked="" type="checkbox"/> CPU 사용 시간	<input type="checkbox"/> DB 연결 시간	<input type="checkbox"/> 액티브 스택 여부	
<input type="checkbox"/> 에이전트 ID (oid)	<input type="checkbox"/> 운영 체제	<input type="checkbox"/> 에러 단계	<input type="checkbox"/> HTTP 호출 시간	<input type="checkbox"/> 메모리 할당량	<input checked="" type="checkbox"/> SQL 패치 건수	<input type="checkbox"/> 커스텀 TID	
<input type="checkbox"/> 에이전트 그룹 ID	<input type="checkbox"/> 국가	<input type="checkbox"/> 에러 메시지			<input type="checkbox"/> SQL 패치 시간	<input type="checkbox"/> 도메인	
<input type="checkbox"/> 에이전트 그룹 명	<input type="checkbox"/> 클라이언트 IP				<input type="checkbox"/> 트랜잭션 해시	<input checked="" type="checkbox"/> 경과 시간	
<input checked="" type="checkbox"/> 에이전트 명 (oname)	<input type="checkbox"/> 로그인 ID				<input checked="" type="checkbox"/> SQL 건수	<input checked="" type="checkbox"/> 종료 시간	
<input type="checkbox"/> 에이전트 서버 ID	<input type="checkbox"/> 유저 에이전트				<input type="checkbox"/> SQL 시간	<input type="checkbox"/> CallerTxId	
<input type="checkbox"/> 에이전트 서버 명	<input type="checkbox"/> UserAgentHash					<input type="checkbox"/> Call Depth	
<input type="checkbox"/> 프로젝트 코드	<input type="checkbox"/> 사용자 종류					<input type="checkbox"/> HTTP 메소드	
<input type="checkbox"/> Pod 명	<input type="checkbox"/> WClientID					<input type="checkbox"/> 멀티 트랜잭션 ID	
<input type="checkbox"/> Sub Type						<input checked="" type="checkbox"/> 원본 URL	
						<input type="checkbox"/> Referer	
						<input checked="" type="checkbox"/> 트랜잭션	
						<input checked="" type="checkbox"/> 시작 시간	
						<input type="checkbox"/> 트랜잭션 ID	

필터 적용

① 영역에서 필터를 활용해 상세 검색이 가능합니다. 다양한 검색 조건에 의한 데이터 탐색 방식을 지원합니다. 필터 항목 역시 여러 개를 적용할 수 있습니다. 원하는 필터를 지정한 다음  버튼을 선택하세요.

- ⓘ • 일부 필터 항목은 상품에 따라 지원하지 않을 수 있습니다.
- 다중 필터 적용 시 **멀티 트랜잭션 ID** 필터를 적용하여 검색할 경우 **트랜잭션 ID** 필터만 추가 적용할 수 있습니다.

경과 시간

CP Kubernetes Monitoring Demo(EKS) ▾ 트랜잭션 검색

시간 선택: < 2024/03/07 00:00 ~ 2024/03/07 23:59 1일 > | 컨테이너: 전체 선택 ▾

필터:
 1000 |
 + 에러 메시지 + HTTP 호출 건수 + SQL 건수 + SQL 패지 건수 + 트랜잭션 + 클라이언트 IP + 도메인 + 유저 에이전트 + Referer + 트랜잭션 ID + 멀티 트랜잭션 ID

+ WClientID + 로그인 ID 에러만 보기

🔍

📄 컬럼 선택

	에이전트 명 (oname)	시작 시간	종료 시간 ↓	경과 시간	트랜잭션	CPU 사용 ...	HTTP 호출 ...	SQL 건수	SQL 패지 건...
1	virtual4	2024/03/07 00:00:00.332	2024/03/07 00:00:01.403	1,071	/product/load/division/jeju	1	0	11	4,079
2	virtual2	2024/03/07 00:00:00.327	2024/03/07 00:00:01.408	1,081	/order/delete/dept/daejun	2	0	11	5,692
3	virtual6	2024/03/07 00:00:00.402	2024/03/07 00:00:01.506	1,104	/account/delete/employee/kwangju	2	0	11	6,957
4	virtual3	2024/03/07 00:00:00.328	2024/03/07 00:00:01.522	1,194	/order/write/dept/kwangju	1	0	11	4,811
5	virtual12	2024/03/07 00:00:00.596	2024/03/07 00:00:01.919	1,323	/edu/create/division/kwangju	2	0	11	4,416
6	virtual7	2024/03/07 00:00:00.848	2024/03/07 00:00:01.926	1,078	/product/save/dept/daejun	2	0	11	5,948
7	virtual1	2024/03/07 00:00:00.864	2024/03/07 00:00:01.972	1,108	/account/create/unit/daegu	3	0	11	5,916
8	virtual8	2024/03/07 00:00:00.849	2024/03/07 00:00:01.981	1,132	/edu/pickup/division/seoul	2	0	11	4,216
9	virtual5	2024/03/07 00:00:00.909	2024/03/07 00:00:02.001	1,092	/account/remove/division/jeju	2	0	11	6,801
10	virtual3	2024/03/07 00:00:01.326	2024/03/07 00:00:02.386	1,060	/product/kill/dept/jeju	1	0	11	4,483
11	virtual4	2024/03/07 00:00:01.329	2024/03/07 00:00:02.624	1,295	/account/write/division/kwangju	1	0	11	3,882
12	virtual12	2024/03/07 00:00:01.598	2024/03/07 00:00:02.657	1,059	/account/load/employee/pusan	2	0	11	5,209
13	virtual15	2024/03/07 00:00:01.500	2024/03/07 00:00:02.755	1,255	/product/load/unit/seoul	1	0	11	6,173
14	virtual6	2024/03/07 00:00:01.706	2024/03/07 00:00:02.766	1,060	/edu/delete/dept/daegu	2	0	11	6,128

이전 다음

트랜잭션 응답시간 기준으로 검색합니다. 기준 이상과 미만의 2개 값을 입력해 검색합니다.

에러 메시지

CP Kubernetes Monitoring Demo(EKS) ▾ 트랜잭션 검색

시간 선택: < 2024/03/07 00:00 ~ 2024/03/07 23:59 1일 > 컨테이너: 전체 선택 ▾

필터: + 경과 시간 - 에러 메시지 SLOW + HTTP 호출 건수 + SQL 건수 + SQL 패지 건수 + 트랜잭션 + 클라이언트 IP + 도메인 + 유저 에이전트 + Referer + 트랜잭션 ID + 멀티 트랜잭션 ID

+ WClientID + 로그인 ID 에러만 보기

🔍 CSV

📄 컬럼 선택

이름 (oname)	시작 시간	종료 시간 ↓	경과 시간	트랜잭션	CPU 사용 ...	HTTP 호출 ...	SQL 건수	SQL 패지 건...	에러 메시지
	2024/03/07 07:37:33.892	2024/03/07 07:40:06.050	152,158	/sales/pickup/dept/seoul	6	10	5	2,264	SLOW_SQL
	2024/03/07 08:13:24.385	2024/03/07 08:15:08.398	104,013	/account/remove/dept/pusan	6	12	3	1,681	SLOW_SQL
	2024/03/07 08:21:58.748	2024/03/07 08:23:47.365	108,617	/order/remove/unit/jeju	1	1	2	407	SLOW_SQL
	2024/03/07 09:18:04.833	2024/03/07 09:20:20.816	135,983	/product/write/unit/kwangju	6	10	8	738	SLOW_SQL
	2024/03/07 09:36:09.707	2024/03/07 09:37:42.759	93,052	/order/save/dept/kwangju	4	9	4	0	SLOW_SQL
	2024/03/07 09:45:27.406	2024/03/07 09:47:05.174	97,768	/account/kill/unit/seoul	4	7	4	0	SLOW_SQL
	2024/03/07 10:00:05.274	2024/03/07 10:02:09.393	124,119	/edu/delete/unit/seoul	8	15	8	1,468	SLOW_SQL
	2024/03/07 10:01:26.195	2024/03/07 10:03:16.080	109,885	/edu/read/division/kwangju	7	10	5	219	SLOW_SQL
	2024/03/07 13:39:27.633	2024/03/07 13:40:54.536	86,903	/product/kill/unit/pusan	1	1	2	0	SLOW_SQL
	2024/03/07 14:52:07.605	2024/03/07 14:54:10.770	123,165	/sales/remove/employee/kwangju	7	13	7	617	SLOW_SQL
	2024/03/07 15:17:47.278	2024/03/07 15:20:05.706	138,428	/account/read/division/daejeon	0	5	3	0	SLOW_SQL
	2024/03/07 15:40:13.738	2024/03/07 15:41:43.304	89,566	/product/remove/employee/pusan	0	14	6	499	SLOW_SQL
	2024/03/07 16:01:24.415	2024/03/07 16:03:53.280	148,965	/sales/delete/division/pusan	4	6	7	1,800	SLOW_SQL

이전 다음

트랜잭션 에러 메시지를 기반으로 검색합니다. 디폴트 상태에서 에러 관련 컬럼이 보이지 않을 수 있습니다. 컬럼을 선택해 확인하세요.

❗ 에러 상태의 트랜잭션은 빨간색으로 표시됩니다.

HTTP 호출 건수

Kubernetes Monitoring Demo(EKS) ▾ 트랜잭션 검색

시간 선택 < 2024/03/07 00:00 ~ 2024/03/07 23:59 1일 > 컨테이너 전체 선택 ▾

필터

+ 경과 시간 + 에러 메시지 - HTTP 호출 건수 5 + SQL 건수 + SQL 패지 건수 + 트랜잭션 + 클라이언트 IP + 도메인 + 유저 에이전트 + Referer + 트랜잭션 ID + 멀티 트랜잭션 ID

+ WClientID + 로그인 ID 예러만 보기

🔍 CSV

📄 컬럼 선택

	에이전트명 (oname)	시작 시간	종료 시간 ↓	경과 시간	트랜잭션	CPU 사용 시간	HTTP 호출 건수	SQL 건수	SQL 패지 건...
1	virtual6	2024/03/07 00:00:00.402	2024/03/07 00:00:09.379	8,977	/order/write/division/jeju	0	7	5	190
2	virtual1	2024/03/07 00:00:01.861	2024/03/07 00:00:12.004	10,143	/order/load/dept/seoul	0	7	6	751
3	virtual5	2024/03/07 00:00:01.909	2024/03/07 00:00:13.474	11,565	/order/write/division/daejun	0	9	6	288
4	virtual8	2024/03/07 00:00:01.849	2024/03/07 00:00:13.749	11,900	/edu/save/employee/seoul	0	10	4	1,062
5	virtual7	2024/03/07 00:00:01.847	2024/03/07 00:00:13.847	12,000	/sales/kill/dept/pusan	0	9	2	340
6	virtual4	2024/03/07 00:00:01.325	2024/03/07 00:00:14.208	12,883	/account/delete/employee/daejun	0	10	4	1,446
7	virtual12	2024/03/07 00:00:01.596	2024/03/07 00:00:15.162	13,566	/order/create/unit/daejun	0	9	4	732
8	virtual15	2024/03/07 00:00:01.498	2024/03/07 00:00:15.562	14,064	/sales/delete/division/jeju	0	11	6	1,088
9	virtual3	2024/03/07 00:00:01.326	2024/03/07 00:00:16.590	15,264	/account/delete/employee/kwangju	0	12	6	1,685
10	virtual5	2024/03/07 00:00:22.439	2024/03/07 00:00:30.425	7,986	/account/pickup/employee/kwangju	3	5	5	739
11	virtual8	2024/03/07 00:00:23.351	2024/03/07 00:00:32.931	9,580	/account/create/dept/seoul	5	8	4	1,064
12	virtual7	2024/03/07 00:00:22.920	2024/03/07 00:00:33.018	10,098	/product/read/division/daejun	5	8	5	408
13	virtual6	2024/03/07 00:00:31.938	2024/03/07 00:00:38.269	6,331	/edu/create/employee/seoul	4	5	5	670

이전 다음

트랜잭션 중 외부 HTTP 서비스를 호출한 건수가 입력값 이상에 해당하는 트랜잭션을 검색합니다. 예시 이미지의 경우 5건 이상을 조회합니다.

SQL 건수

CP Kubernetes Monitoring Demo(EKS) ▾ 트랜잭션 검색

시간 선택: < 2024/03/07 00:00 ~ 2024/03/07 23:59 1일 > 컨테이너: 전체 선택 ▾

필터: + 경과 시간 + 에러 메시지 + HTTP 호출 건수 + SQL 건수 1200 + SQL 패치 건수 + 트랜잭션 + 클라이언트 IP + 도메인 + 유저 에이전트 + Referer + 트랜잭션 ID + 멀티 트랜잭션 ID
+ WClientID + 로그인 ID 예러만 보기

🔍 CSV

📄 컬럼 선택

	에이전트 명 (oname)	시작 시간	종료 시간 ↓	경과 시간	트랜잭션	CPU 사용 ...	HTTP 호출 ...	SQL 건수	SQL 패치 건...
1	virtual6	2024/03/06 23:36:01.577	2024/03/07 00:04:09.315	1,687,738	/order/read/employee/daejun 📄 📄	0	10	1,504	736,255

트랜잭션별 SQL 수행 건수가 설정한 입력값 이상에 해당하는 트랜잭션을 검색합니다. 예외적으로 과도한 수의 SQL을 호출하는 트랜잭션을 검색할 때 이용합니다.

SQL 패치 건수

CP Kubernetes Monitoring Demo(EKS) ▾ 트랜잭션 검색

시간 선택: < 2024/03/11 00:00 ~ 2024/03/11 23:59 1일 > 컨테이너: 전체 선택 ▾

필터: + 경과 시간 + 에러 메시지 + HTTP 호출 건수 + SQL 건수 + SQL 패치 건수 500000 + 트랜잭션 + 클라이언트 IP + 도메인 + 유저 에이전트 + Referer + 트랜잭션 ID + 멀티 트랜잭션 ID
+ WClientID + 로그인 ID 예러만 보기

🔍 CSV

📄 컬럼 선택

	에이전트 명 (oname)	시작 시간	종료 시간 ↓	경과 시간	트랜잭션	CPU 사용 ...	HTTP 호출 ...	SQL 건수	SQL 패치 건...
1	virtual3	2024/03/11 00:08:03.718	2024/03/11 00:18:25.396	621,678	/product/read/unit/jeju 📄 📄	264	5	1,003	502,277
2	virtual1	2024/03/11 00:16:04.358	2024/03/11 00:24:39.449	515,091	/product/kill/unit/daejun 📄 📄	341	12	1,003	506,705
3	virtual2	2024/03/11 04:01:41.335	2024/03/11 04:16:29.103	887,768	/account/save/employee/seoul 📄 📄	239	2	1,002	509,630
4	virtual15	2024/03/11 04:42:09.848	2024/03/11 04:53:32.471	682,623	/account/delete/employee/jeju 📄 📄	0	7	1,002	500,402

SQL 수행 후 DB 패치 건수를 기준으로 입력값 이상에 해당하는 트랜잭션을 검색합니다. 과도한 데이터 패치를 수행하는 트랜잭션을 검색할 때 이용합니다.

트랜잭션

CP Kubernetes Monitoring Demo(EKS) ▾ 트랜잭션 검색

시간 선택 < 2024/03/11 00:00 ~ 2024/03/11 23:59 1일 > 컨테이너 전체 선택 ▾

필터

+ 경과 시간 + 예러 메시지 + HTTP 호출 건수 + SQL 건수 + SQL 패치 건수 **- 트랜잭션 product** + 클라이언트 IP + 도메인 + 유저 에이전트 + Referer + 트랜잭션 ID + 멀티 트랜잭션 ID

+ WClientID + 로그인 ID 예러만 보기

🔍 CSV

📄 컬럼 선택

	에이전트명 (oname)	시작 시간	종료 시간 ↓	경과 시간	트랜잭션	CPU 사용 ...	HTTP 호출 ...	SQL 건수	SQL 패치 건...
1	virtual6	2024/03/11 00:00:00.261	2024/03/11 00:00:01.465	1,204	/product/create/division/daegu M 🗑	3	0	11	6,100
2	virtual1	2024/03/11 00:00:01.335	2024/03/11 00:00:02.571	1,236	/product/remove/employee/seoul M 🗑	3	0	11	6,169
3	virtual8	2024/03/11 00:00:01.542	2024/03/11 00:00:02.787	1,245	/product/load/employee/seoul P 🗑	2	0	11	5,614
4	virtual15	2024/03/11 00:00:03.775	2024/03/11 00:00:04.835	1,060	/product/write/employee/kwangju M 🗑	1	0	11	5,322
5	virtual4	2024/03/11 00:00:03.968	2024/03/11 00:00:05.047	1,079	/product/load/dept/jeju M 🗑	2	0	11	5,343

트랜잭션 기준으로 검색합니다. 입력한 검색 문자열은 포함 여부(부분 일치)로 비교 검색합니다.

클라이언트 IP

CP Kubernetes Monitoring Demo(EKS) ▾ 트랜잭션 검색

시간 선택 < 2024/03/11 00:00 ~ 2024/03/11 23:59 1일 > 컨테이너 전체 선택 ▾

필터

+ 경과 시간 + 예러 메시지 + HTTP 호출 건수 + SQL 건수 + SQL 패치 건수 + 트랜잭션 **- 클라이언트 IP 87.132** + 도메인 + 유저 에이전트 + Referer + 트랜잭션 ID + 멀티 트랜잭션 ID

+ WClientID + 로그인 ID 예러만 보기

🔍 CSV

📄 컬럼 선택

	에이전트명 (oname)	시작 시간	종료 시간 ↓	경과 시간	트랜잭션	클라이언트 IP	CPU 사용 ...	HTTP 호출 ...	SQL
1	virtual5	2024/03/11 00:00:00.849	2024/03/11 00:00:01.959	1,110	/account/save/employee/seoul M 🗑	87.132.232.135	2	0	
2	virtual15	2024/03/11 00:00:04.835	2024/03/11 00:00:06.099	1,264	/account/remove/division/jeju M 🗑	87.132.232.135	2	0	
3	virtual7	2024/03/11 00:00:56.525	2024/03/11 00:00:57.585	1,060	/sales/read/employee/daejeon M 🗑	87.132.232.135	1	0	
4	virtual3	2024/03/11 00:01:16.340	2024/03/11 00:01:17.399	1,059	/product/remove/unit/jeju M 🗑	87.132.232.135	1	0	
5	virtual7	2024/03/11 00:03:33.109	2024/03/11 00:03:34.177	1,068	/product/write/dept/kwangju M 🗑	87.132.232.135	1	0	

클라이언트 IP 기반으로 트랜잭션을 검색합니다. 단 IP 문자열은 PREFIX 비교(전방 일치)로 검색합니다.

도메인

CP Kubernetes Monitoring Demo(EKS) ▾ 트랜잭션 검색

시간 선택 < 2024/03/11 00:00 ~ 2024/03/11 23:59 1일 > 컨테이너 전체 선택 ▾

필터

+ 경과 시간 + 에러 메시지 + HTTP 호출 건수 + SQL 건수 + SQL 패치 건수 + 트랜잭션 + 클라이언트 IP + 도메인 127.0.0.1:8181 + 유저 에이전트 + Referer + 트랜잭션 ID + 멀티 트랜잭션 ID

+ WClientID + 로그인 ID 에러만 보기

🔍 CSV

📄 컬럼 선택

	에이전트 명 (oname)	시작 시간	종료 시간 ↓	경과 시간	트랜잭션	도메인	CPU 사용 ...	HTTP 호출 ...	SQL 건수	S
1	virtual1	2024/03/11 00:00:00.333	2024/03/11 00:00:01.418	1,085	/edu/save/unit/daejun	127.0.0.1:8181		3	0	11
2	virtual4	2024/03/11 00:00:00.401	2024/03/11 00:00:01.462	1,061	/order/kill/division/seoul	127.0.0.1:8181		1	0	11
3	virtual6	2024/03/11 00:00:00.261	2024/03/11 00:00:01.465	1,204	/product/create/division/daegu	127.0.0.1:8181		3	0	11
4	virtual12	2024/03/11 00:00:00.253	2024/03/11 00:00:01.547	1,294	/account/create/dept/kwangju	127.0.0.1:8181		2	0	11
5	virtual8	2024/03/11 00:00:00.539	2024/03/11 00:00:01.727	1,188	/edu/kill/dept/daegu	127.0.0.1:8181		2	0	11

HTTP 서비스 도메인(host:port) 기반으로 트랜잭션을 검색합니다. 검색 조건은 도메인이 완전히 일치(전체 일치)해야 합니다.

유저 에이전트

CP Kubernetes Monitoring Demo(EKS) ▾ 트랜잭션 검색

시간 선택 < 2024/03/11 00:00 ~ 2024/03/11 23:59 1일 > 컨테이너 전체 선택 ▾

필터

+ 경과 시간 + 에러 메시지 + HTTP 호출 건수 + SQL 건수 + SQL 패치 건수 + 트랜잭션 + 클라이언트 IP + 도메인 + 유저 에이전트 Chrome + Referer + 트랜잭션 ID + 멀티 트랜잭션 ID

+ WClientID + 로그인 ID 에러만 보기

🔍 CSV

📄 컬럼 선택

이름 (oname)	시작 시간	종료 시간 ↓	경과 시간	트랜잭션	CPU 사용 ...	HTTP 호출 ...	SQL 건수	SQL 패치 건...	유저 에이전트	
	2024/03/11 00:00:00.806	2024/03/11 00:00:01.871	1,065	/account/kill/employee/kwangju		2	0	11	5,195	Mozilla/5.0 (Windows; U; Windows NT 6.1; en-US; AppleWebKit/532.1 (KHTML, like Gecko) Chrome/4.0.213.1 Safari/532.1
	2024/03/11 00:00:01.399	2024/03/11 00:00:02.502	1,103	/account/pickup/division/seoul		1	0	11	4,350	Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; AppleWebKit/532.0 (KHTML, like Gecko) Chrome/3.0.198.0 Safari/532.0

유저 에이전트 기반으로 트랜잭션을 검색(부분 일치)합니다.

Referer

Referer 문자열 기반으로 트랜잭션을 검색합니다.

① 기본 설정 상태의 에이전트는 Referer 정보를 수집하지 않습니다. Referer 설정이 필요한 경우 다음 문서를 참조해 관련 설정을 추가하세요.

- [Java](#)
- [Node.js](#)
- [Python](#)

트랜잭션 ID

CP Kubernetes Monitoring Demo(EKS) ▾ 트랜잭션 검색

시간 선택: < 2024/03/14 00:00 ~ 2024/03/14 23:59 [1일] > 컨테이너: 전체 선택 ▾

필터: + 경과 시간 + 에러 메시지 + HTTP 호출 건수 + SQL 건수 + SQL 패치 건수 + 트랜잭션 + 클라이언트 IP + 도메인 + 유저 에이전트 + Referer + 트랜잭션 ID -4884510994358729563 + 멀티 트랜잭션 ID + WClientID + 로그인 ID 에러만 보기

🔍 CSV

컬럼 선택

에이전트명 (oname)	시작 시간	종료 시간	경과 시간	트랜잭션	CPU 사용 ...	HTTP 호출 ...	SQL 건수	SQL 패치 건...	트랜잭션 ID
irtual7	2024/03/14 00:00:00.099	2024/03/14 00:00:01.163	1,064	/account/write/division/jeju	2	0	11	4,106	-488451099 4358729563

트랜잭션 ID 기반으로 트랜잭션을 검색(전체 일치)합니다. 트랜잭션 ID의 경우 검색 인덱스가 생성되어 있기 때문에 빠른 속도로 검색할 수 있습니다.

- ① • 트랜잭션 ID는 [트랜잭션 정보](#) 창의 [레코드 요약](#) 탭에서 확인할 수 있습니다. 자세한 내용은 [다음 문서](#)를 참조하세요.
- 최신 버전의 에이전트가 아닌 경우 인덱스 사용이 아닌 폴스캔이 발생할 수 있습니다.

멀티 트랜잭션 ID

CP Kubernetes Monitoring Demo(EKS) ▾ 트랜잭션 검색

시간 선택: < 2024/03/14 00:00 ~ 2024/03/14 23:59 1일 > 컨테이너: 전체 선택 ▾

필터: + 경과 시간 + 에러 메시지 + HTTP 호출 건수 + SQL 건수 + SQL 패치 건수 + 트랜잭션 + 클라이언트 IP + 도메인 + 유저 에이전트 + Referer + 트랜잭션 ID - 멀티 트랜잭션 ID -1652480132223001848

+ WClientID + 로그인 ID 예러만 보기

🔍 CSV

📄 필터 선택

에이전트명 (oname)	시작 시간	종료 시간 ↓	경과 시간	트랜잭션	CPU 사용 ...	HTTP 호출 ...	SQL 건수	SQL 패치 건...	멀티 트랜잭션 ID
irtual7	2024/03/14 00:00:00.099	2024/03/14 00:00:01.163	1,064	/account/write/division/jeju 📄	2	0	11	4,106	-1652480132223001848
irtual7	2024/03/14 01:39:22.389	2024/03/14 01:39:23.447	1,058	/order/create/dept/daegu 📄	1	0	11	4,150	-1652480132223001848
irtual7	2024/03/14 01:39:23.618	2024/03/14 01:39:24.803	1,185	/product/pickup/employee/pusan 📄	1	0	11	5,793	-1652480132223001848
irtual7	2024/03/14 01:39:24.812	2024/03/14 01:39:26.033	1,221	/order/pickup/unit/pusan 📄	1	0	11	4,237	-1652480132223001848
irtual7	2024/03/14 01:39:49.669	2024/03/14 01:39:50.791	1,122	/order/save/division/jeju 📄	1	0	11	5,991	-1652480132223001848
irtual7	2024/03/14 01:39:50.792	2024/03/14 01:39:51.858	1,066	/account/save/dept/pusan 📄	1	0	11	4,581	-1652480132223001848
irtual7	2024/03/14 00:00:00.096	2024/03/14 01:39:51.858	5,991,762	/product/read/unit/pusan 📄	0	6	1,004	515,581	-1652480132223001848

멀티 트랜잭션 ID 기반으로 트랜잭션을 검색(전체 일치)합니다.

- ① • 멀티 트랜잭션 ID는 [트랜잭션 정보](#) 창의 [레코드 요약](#) 탭에서 확인할 수 있습니다. 자세한 내용은 [다음 문서](#)를 참조하세요.
- 최신 버전의 에이전트가 아닌 경우 인덱스 사용이 아닌 풀스캔이 발생할 수 있습니다.
- [멀티 트랜잭션 ID](#) 필터를 적용하여 검색할 경우 [트랜잭션 ID](#) 필터만 추가 적용할 수 있습니다.

WClientID

CP Kubernetes Monitoring Demo(EKS) ▾ 트랜잭션 검색

시간 선택: < 2024/03/14 00:00 ~ 2024/03/14 23:59 1일 > 컨테이너: 전체 선택 ▾

필터: + 경과 시간 + 에러 메시지 + HTTP 호출 건수 + SQL 건수 + SQL 패치 건수 + 트랜잭션 + 클라이언트 IP + 도메인 + 유저 에이전트 + Referer + 트랜잭션 ID + 멀티 트랜잭션 ID

- WClientID [] + 로그인 ID 에러만 보기

🔍

📄 컬럼 선택

에이전트명 (oname)	시작 시간	종료 시간 ↓	경과 시간	트랜잭션	CPU 사용 ...	HTTP 호출 ...	SQL 건수	SQL 패치 건...	WClientID
rtual7	2024/03/14 00:00:00.099	2024/03/14 00:00:01.163	1,064	/account/write/division/jeju		2	0	11	4,106
rtual1	2024/03/14 00:00:32.933	2024/03/14 00:00:34.141	1,208	/account/delete/dept/seoul		2	0	11	6,045
rtual1	2024/03/14 00:02:59.129	2024/03/14 00:03:00.327	1,198	/account/delete/division/seoul		2	0	11	5,916
rtual9	2024/03/14 00:04:04.929	2024/03/14 00:04:05.987	1,058	/order/kill/unit/pusan		1	0	11	7,482
rtual12	2024/03/14 00:04:08.831	2024/03/14 00:04:09.889	1,058	/order/read/dept/pusan		1	0	11	4,419
rtual12	2024/03/14 00:04:12.027	2024/03/14 00:04:13.245	1,218	/sales/create/division/pusan		2	0	11	4,160
rtual12	2024/03/14 00:05:15.491	2024/03/14 00:05:16.747	1,256	/sales/save/division/jeju		2	0	11	5,934

클라이언트 ID(`wClientID`)는 와탭 에이전트가 사용자를 구분하기 위해 부여하는 ID입니다. 이 ID 값 기반으로 트랜잭션을 검색(전체 일치)합니다. 검색을 위한 별도 인덱스는 없습니다.

❗ 와탭은 클라이언트와 관련한 정보를 기본 저장합니다.

로그인 ID

로그인 아이디 기반으로 트랜잭션을 검색합니다.

❗ 기본 설정 상태의 에이전트는 로그인 ID 정보를 수집하지 않습니다. 로그인 ID 설정이 필요한 경우 [다음 문서](#)를 참조해 설정을 추가하세요.

검색 결과 다운로드하기

사용자가 설정한 필터 조건에 따라 검색된 결과를 CSV 파일로 다운로드할 수 있습니다.

1. 🔍 버튼을 클릭해 검색 결과를 조회하세요.
2. 버튼을 선택하세요.

3. 최대 CSV 라인 수를 입력하세요. 입력한 라인 수만큼 조회된 결과를 저장할 수 있습니다.
4. [다운로드](#) 버튼을 선택하면 CSV 파일로 다운로드합니다.

ⓘ CSV 파일 이름은 `profile_{pcode}_YYMMDD.csv` 형식입니다.

알림 설정하기

이벤트 조건을 설정하고 이메일, SMS, 메신저, App Push 등 다양한 경로로 알림을 수신할 수 있습니다.

- 에이전트와 수집 서버의 통신 두절 및 자원 사용률, 액티브 트랜잭션, 에러, 응답 지연, 응답 분포 이상에 대한 알림 설정이 가능합니다.
- 어려운 조건의 알림 설정은 태그 카운트 알림 기능을 통해 제공합니다.
- 머신 러닝 기반 AI 알림 설정을 통해 히트맵 패턴을 감지해 알림 기능을 제공합니다. AI 모듈이 애플리케이션 실행 분포 패턴을 판단해 알림을 보냅니다. 복잡한 조건 없이 이슈 상황을 빠르고 정확하게 인지할 수 있습니다.
- [이벤트 설정](#) 메뉴를 통해 임계치 기반 알림 조건 및 수신 설정을 할 수 있습니다.
- [이벤트 수신 설정](#) 메뉴에서는 개인별 알림 수신 채널 및 시간대를 설정할 수 있습니다.
- [이벤트 기록](#) 메뉴에서는 알림 발생 기록을 조회할 수 있습니다.

- ① • 다른 계정에 대한 알림을 설정하려면 [수정](#), [알림 설정](#), [멤버 관리](#) 권한 중 하나가 있어야 합니다.
 - 이벤트를 추가하거나 설정하려면 알림 설정 권한이 있어야 합니다. 권한 체계에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.
 - 경고 알림과 관련해 모니터링 플랫폼별 지원되는 이벤트 종류를 확인하려면 [다음 문서](#)를 참조하세요.

컨테이너 경고 알림

홈 화면 > 프로젝트 선택 > 경고 알림 > 이벤트 설정 > 컨테이너 탭 선택

컨테이너의 자원 사용량과 진행 중 트랜잭션, 에러 상황의 이벤트 조건을 통해 경고 알림을 보냅니다.

이벤트 설정

이벤트 비활성 경고 1분 | 이벤트 상태가 해결되면 추가 알림 | 에이전트 중복 알림 1분

메트릭	조건	지속	무음	상태	
서버 CPU	조건 (%) 70%	90%	30초	5분	On
서버 메모리	조건 (%) 70%	90%	30초	5분	Off
서버 디스크	조건 (%) 70%	90%	30초	5분	Off
액티브 트랜잭션	조건 (건수) 100		30초	5분	Off
트랜잭션 에러	조건 (건수) 10		30초	5분	Off
트랜잭션 응답시간	조건 (건수) 10	조건 (응답시간) 10초	5초	5분	Off

메트릭스

카테고리	대상	규칙	이벤트 상태가 해결되면 추가 알림	이벤트 수신 태그
test	ApplicationActiveTrxMeter 5분 *	active_httpc > 0	Off	전체 멤버 수신

복합 메트릭스

이벤트 설정 없음. '이벤트 추가' 버튼을 클릭하십시오.

기본 옵션

다음은 각 항목별 설정할 수 있는 기본 옵션입니다.

- 에이전트 비활성화 경고

설정된 시간 동안 에이전트로부터 데이터를 수신할 수 없을 경우 경고 알림을 보냅니다. 토글 버튼을 선택해 기능을 켜거나 끌 수 있습니다.

- 이벤트 상태가 해결되면 추가 알림

애플리케이션 이벤트 항목 중 발생한 이벤트 상태가 해결되면 추가 알림 송신 여부를 선택할 수 있습니다. 토글 버튼을 선택해 기능을 켜거나 끌 수 있습니다.

- 에이전트 중복 알림

모니터링 중인 에이전트 가운데 동일한 ID를 설정한 에이전트가 2개 이상일 때 지표를 중복으로 수집하면서 발생하는 경고 알림입니다. 에이전트 이름을 중복으로 설정하지 않았는지 에이전트 설정에서 관련 옵션을 확인하세요. 블루/그린과 같은 배포 상황에서 에이전트가 중복되어 경고 알림이 발생할 경우 이 기능을 이용하길 권장합니다. 같은 이름의 에이전트가 동시에 2개 이상 기동된 경우 선택한 시간 동안 알림을 받지 않습니다.

- 조건

① 조건은 경고(Warning) 레벨, ② 조건은 위험(Critical) 레벨입니다.

- ③ 지속

선택한 시간 동안 이벤트 조건이 지속될 경우 알림을 보냅니다.

- ④ 무음

알림을 보낸 후 선택한 시간 동안 같은 알림을 보내지 않습니다.

- ⑤ On / Off

토글 버튼을 선택해 발생하는 이벤트를 켜거나 끌 수 있습니다.

항목별 이벤트 설정

다음의 이벤트 항목을 설정할 수 있습니다.

- 서버 CPU, 서버 메모리, 서버 디스크

각 항목의 사용량이 설정한 조건(%)에 설정한 기준치를 도달하면 알림을 보냅니다.

- 액티브 트랜잭션

액티브 트랜잭션 발생 건수가 조건(건수)에 설정한 기준치를 넘는 상태를 지속하면 알림을 보냅니다. 발생 건수 기본값은 100개 기준입니다.

- 트랜잭션 에러

정상적으로 수행하지 않은 트랜잭션을 의미합니다. 에러 건수가 **조건(건수)**에 설정한 기준치를 넘은 상태를 지속하면 알림을 보냅니다. 수집 주기는 5초입니다.

- **트랜잭션 응답시간**

트랜잭션이 끝난 시간으로 확인합니다. 예를 들어 **10초**의 **응답시간** 기준을 넘는 트랜잭션의 **건수**가 **10**개를 초과하고, 이 상태가 **5초간 지속**되면 알림을 보냅니다.

이벤트 설정

홈 화면 > 프로젝트 선택 >  사이트맵 > 경고 알림 > 이벤트 설정 New











새로운 **이벤트 설정** 메뉴는 카테고리(Category)가 아닌 필드(Field) 중심의 사용자 경험을 강화한 메뉴입니다. 매트릭스 이벤트 설정과 같은 강력한 기능을 모두 필요치 않거나 단순한 알림 조건을 반복해서 사용하는 사용자에게 빠르게 경고 알림을 적용할 수 있는 이벤트 기능입니다. 생성한 프로젝트의 상품에 적합한 기본 이벤트 템플릿을 제공하여 빠르고 쉽게 원하는 경고 알림 이벤트를 설정할 수 있습니다.

카테고리가 아니라 필드를 먼저 선택해 사용자가 인지하는 지표와 이벤트 설정의 불일치를 해소합니다. 시뮬레이션이 주는 시각적 직관성을 결합하여 빠르고 정확하게 이벤트 설정을 완료할 수 있습니다.

- ① 신규 프로젝트는 별도의 설정 없이 이 기능을 이용할 수 있으나 이미 생성된 프로젝트에서는 **이벤트 템플릿 생성** 버튼을 선택하세요. 새로운 기능에 최적화된 이벤트 목록을 자동으로 생성합니다.
- 상품에 따라 제공하는 이벤트 템플릿은 다를 수 있습니다.
- 이 기능은 알림 설정 권한이 있는 멤버만 이용할 수 있습니다. 멤버 권한에 대한 자세한 설명은 [다음 문서](#)를 참조하세요.

기본 이벤트 템플릿

생성한 프로젝트의 상품에 따라 기본 이벤트 템플릿을 제공합니다. 이벤트 목록의 가장 왼쪽에 **활성화** 버튼을 선택해 원하는 이벤트를 활성화할 수 있습니다.

활성화	수정	이벤트 이름 ↑	지표 이름	지표	대상	연속 ①	일시 중지	해결된 알림	이벤트 수신
<input type="checkbox"/>		Active Transaction	Active Transaction (count)	● > 100	전체	6회	5분	사용	전체 수신
<input type="checkbox"/>		Container CPU Utilization Relative ...	Container CPU Utilization Relative to ...	● > 70 ● > 90	전체	6회	5분	사용	전체 수신
<input type="checkbox"/>		Container Memory Utilization Rela...	Container Memory Utilization Relative...	● > 70 ● > 90	전체	6회	5분	사용	전체 수신
<input type="checkbox"/>		Deployment Replicas Pending Pod...	Deployment Replicas Pending Pods C...	● > 0	전체	6회	5분	사용 안함	전체 수신
<input type="checkbox"/>		Node CPU	Node CPU (%)	● > 70 ● > 90	전체	6회	5분	사용 안함	전체 수신
<input type="checkbox"/>		Node Disk(/) Utilization	Node Disk(/) Utilization (%)	● > 70 ● > 90	전체	6회	5분	사용 안함	전체 수신
<input type="checkbox"/>		Node Memory	Node Memory (%)	● > 70 ● > 90	전체	6회	5분	사용 안함	전체 수신
<input type="checkbox"/>		Restart Count Of Container In Pod	Restart Count Of Container In Pod (co...	● > 5	전체	사용 안함	5분	사용 안함	전체 수신
<input type="checkbox"/>		Transaction Error	Transaction Error (count)	● > 10	전체	6회	5분	사용	전체 수신
<input type="checkbox"/>		Transaction Response Time	Transaction Response Time (ms)	● > 10,000	전체	사용 안함	5분	사용	전체 수신

제공하는 이벤트 템플릿은 다음과 같습니다. **지표** 항목에 설정된 기본값은 사용자가 원하는 값으로 수정할 수 있습니다.

- [Active Transaction](#)

카테고리: `app_counter`

액티브 트랜잭션 발생 건수(`active_tx_count`)가 100개를 초과하면 경고(**Warning**) 수준의 알림을 보냅니다.

- [Container CPU Utilization Relative to Container CPU Limit](#)

카테고리: `container`

컨테이너의 실제 CPU 사용률(`cpu_per_quota`)이 70%를 초과하면 경고(**Warning**) 수준의 알림을, 90%를 초과하면 위험(**Critical**) 수준의 알림을 보냅니다.

- [Container Memory Utilization Relative to Container Memory Limit](#)

카테고리: `container`

컨테이너의 실제 메모리 사용량(`mem_percent`)이 70%를 초과하면 경고(**Warning**) 수준의 알림을, 90%를 초과하면 위험(**Critical**) 수준의 알림을 보냅니다.

- [Deployment Replicas Pending Pods Count](#)

카테고리: `kube_pod_stat`

사용할 수 있는 Pod 개수(`available_pod`)가 배포 시 필요한 Pod 개수(`desired_pod`)보다 작으면 경고(**Warning**) 수준의 알림을 보냅니다. 사용할 수 있는 Pod 개수와 배포 시 필요한 Pod 개수의 차이를 임계값으로 입력하세요.

- [Node CPU](#)

카테고리: `server_base`

Node의 CPU 사용률(`cpu`)이 70%를 초과하면 경고(**Warning**) 수준의 알림을, 90%를 초과하면 위험(**Critical**) 수준의 알림을 보냅니다.

- [Node Disk\(/\) Utilization](#)

카테고리: `server_disk`

Node의 루트 경로에 해당하는 디스크 사용률(`usedPercent`)이 70%를 초과하면 경고(**Warning**) 수준의 알림을, 90%를 초과하면 위험(**Critical**) 수준의 알림을 보냅니다.

- [Node Memory](#)

카테고리: `server_base`

Node의 메모리 사용률(`memory_pused`)이 70%를 초과하면 경고(**Warning**) 수준의 알림을, 90%를 초과하면 위험(**Critical**) 수준의 알림을 보냅니다.

- [Restart Count Of Container In Pod](#)

카테고리: `kube_pod`

Pod의 재시작 횟수(`RestartCount`)가 5개를 초과하면 경고(**Warning**) 수준의 알림을 보냅니다.

- **Transaction Error Count**

카테고리: `app_counter`

트랜잭션 에러 건수(`tx_error`)가 10개를 초과하면 경고(**Warning**) 수준의 알림을 보냅니다.


- **Transaction Response Time**

카테고리: `app_counter`

트랜잭션의 평균 응답 시간(`resp_time`)이 10회 연속, 2,000ms(2초)를 초과한 상태로 지속되면 경고(**Warning**) 수준의 알림을 보냅니다.

ⓘ 기본 이벤트 템플릿에 적용된 필드에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

이벤트 수정하기

기본 이벤트 템플릿에 적용된 지표값을 수정해 경고 알림이 발생 기준을 변경할 수 있습니다. 그 외에도 다양한 옵션 설정을 통해 알림 발생 기준을 변경할 수 있습니다. 수정하려는 이벤트 항목에서  버튼을 선택하면 이벤트 수정 화면으로 이동합니다. 다음의 항목을 차례로 수정한 다음 [저장](#) 버튼을 선택하세요.

이벤트 조건 정의

이벤트 발생 조건 기준을 설정할 수 있습니다.

1 이벤트 조건 정의

< 2024/02/06 10:46 ~ 2024/02/06 11:46 60분 > 실행

①
조건 설정 후 실행 버튼을 누르면 이벤트를 시뮬레이션 할 수 있습니다.

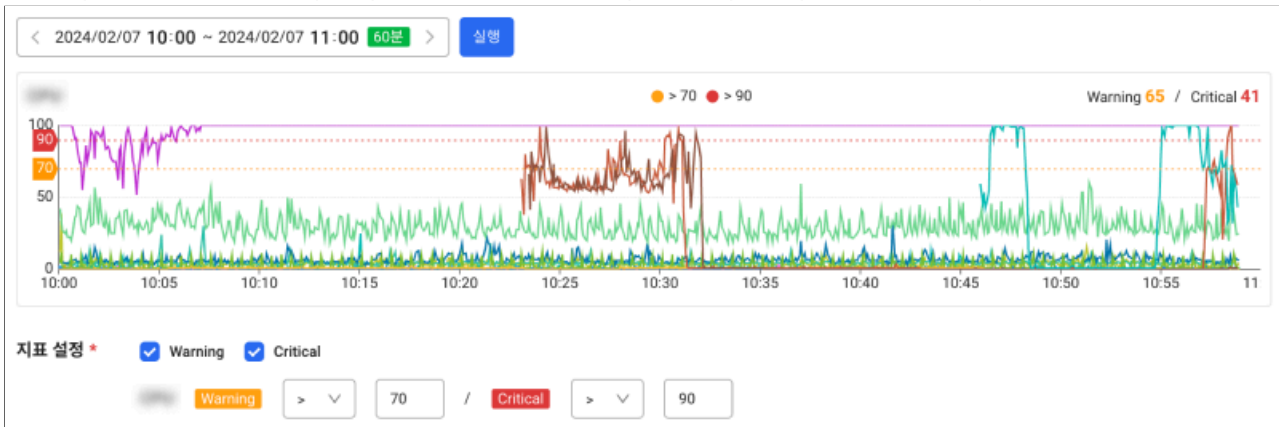
지표 설정 * Warning Critical
Warning > 70 / Critical > 90

연속 사용 안함 연속 1 회 발생 (1분에 최대 N회 발생)

일시 중지 1분
알림 수신 후 선택한 시간 동안 이벤트가 발생하지 않습니다.
단, "이벤트 상태가 해결되면 추가 알림" 기능을 활성화한 경우에는 RECOVERED 알림 수신 후 선택한 시간 동안 이벤트가 발생하지 않습니다.

해결된 알림
Critical과 Warning 레벨의 이벤트가 해결되면 RECOVERED 상태의 알림을 수신합니다.

- 실행: 이벤트 조건을 설정한 다음 버튼을 선택하면 설정한 이벤트를 시뮬레이션할 수 있습니다.



- 지표 설정: Warning 또는 Critical 알림에 대한 임계값을 설정할 수 있습니다. 연산자를 설정하고 임계값을 입력하세요.
 - 연산자 종류: > (보다 큼), >= (보다 크거나 같음), < (보다 작음), <= (보다 작거나 같음), == (같음)
 - 임계값으로 양수, 음수, 0을 입력할 수 있지만, 선택한 이벤트 템플릿에 따라 음수 입력 여부는 다를 수 있습니다.

- ⓘ ○ 제공되는 이벤트 템플릿에 따라 설정할 수 있는 알림 수준(Level)은 다를 수 있습니다.
- **Warning** 또는 **Critical** 수준 하나만 선택할 수 있지만, 두 항목을 모두 비활성화할 수는 없습니다.

- **연속**: 설정한 이벤트 조건을 충족하는 횟수에 따라 알림을 보낼 수 있습니다.
 - **사용 안함**: 이벤트가 발생할 때마다 알림을 보냅니다.
 - **연속**: 입력한 횟수만큼 이벤트가 발생하면 알림을 보냅니다.
- **일시 중지**: 알림 수신 후 선택한 시간 동안 이벤트가 발생하지 않습니다.
- **해결된 알림**: 이벤트가 해결되면 **RECOVERED** 상태의 알림을 보냅니다.

☑ 다음 조건에 따라 알림을 보내는 기준이 달라집니다.

연속	해결된 알림	동작
사용 안함	On	설정된 이벤트 조건을 충족하거나 해결되면 알림을 보냅니다.
	Off	설정된 조건을 충족할 때마다 알림을 보냅니다.
연속 N 회 발생	On	N 회 이상 조건을 충족하거나 N 회 이상 이벤트가 해결되면 알림을 보냅니다.
	Off	N 회 이상 설정한 이벤트 조건을 충족하면 알림을 보냅니다. 알림을 보낸 후 횟수를 초기화해 횟수를 다시 집계합니다.

이벤트 대상 선택

특정 에이전트에서만 발생하는 이벤트를 알림으로 보내도록 설정할 수 있습니다. **에이전트 (인스턴스)** 항목을 클릭한 다음 원하는 에이전트를 선택하세요.

▼ 2 이벤트 대상 선택

대상 선택 전체 대상이 포함됩니다.

에이전트 (인스턴스)

위 항목들을 설정하지 않으면 프로젝트에 포함된 전체 에이전트를 대상으로 이벤트가 실행되며 많은 알림이 발생할 수 있습니다.

이벤트 대상에 대한 자세한 설명은 가이드 문서를 참고하세요. [가이드 문서 >](#)

이벤트를 시뮬레이션하려면 화면 최상단에 '이벤트 조건 정의' 항목에서 [실행 버튼](#)을 선택하세요.

- ① • 제공되는 템플릿에 따라 선택할 수 있는 대상은 다를 수 있습니다.
- 선택한 대상에 대한 이벤트 발생 현황을 확인하려면 [이벤트 조건 정의](#)에서 [실행](#) 버튼을 선택하세요.
- 대상을 선택하지 않으면 프로젝트에 포함된 전체 에이전트를 대상으로 알림을 보냅니다. 전체 에이전트를 대상으로 이벤트가 실행되면 많은 알림이 발생할 수 있습니다.

기본 정보 및 수신 설정

알림으로 수신되는 이벤트의 제목과 메시지를 작성하고, 수신 대상을 선택할 수 있습니다.

▼ 3 기본 정보 및 수신 설정

이벤트 활성화

이벤트 이름 *

메시지 *

\$를 입력하면 변수를 입력할 수 있습니다.
ex) \$(active_tx_8) \$(time)에 예러가 발생했습니다.

수신 테스트

이벤트 수신 전체 수신 태그 선택 수신

- **이벤트 활성화:** 현재 이벤트를 활성화할 수 있습니다.
- **이벤트 이름:** 기본 제공된 템플릿의 이벤트 이름이 입력되어 있습니다. 사용자가 원하는 이벤트 이름으로 편집할 수 있습니다.
- **메시지:** 기본 제공된 템플릿의 메시지가 입력되어 있습니다. 사용자가 원하는 메시지로 편집할 수 있습니다. 버튼을 클릭하면 이전에 입력한 메시지 기록을 확인할 수 있습니다.

✓ 메시지 입력 창에 `#{Tag}` 또는 `#{Field}` 변수를 입력해 메시지를 작성할 수 있습니다.
 분석 > 매트릭스 조회 메뉴에서 **카테고리**를 선택한 다음 입력할 수 있는 `#{Tag}` 또는 `#{Field}` 변수를 확인하세요.
 현재 이벤트 템플릿의 **카테고리** 이름은 [다음 문서](#)의 **카테고리** 항목을 참조하세요.

- **수신 테스트**: 현재 이벤트로 발생하는 알림 수신 시 입력한 이벤트 이름과 메시지를 사전 점검할 수 있습니다. 필수 항목(**지표 설정**, **이벤트 이름**, **메시지**)을 모두 입력해야 테스트할 수 있습니다.

ⓘ 테스트 중에는 실제 매트릭스 값이나 변수에 대한 치환 기능이 작동하지 않으며, 수신자 태그가 설정된 사용자에게만 알림을 보낼 수 없습니다.

- **이벤트 수신**: 현재 이벤트로 발생하는 알림을 수신할 멤버를 선택할 수 있습니다.
 - **전체 수신**: 프로젝트에 소속된 멤버 전원에게 알림을 보냅니다.
 - **태그 선택 수신**: 선택한 태그를 가진 프로젝트 멤버와 3rd-party 플러그인에 알림을 보냅니다. **태그 추가** 또는 **+** 버튼을 클릭해 태그 목록에서 원하는 태그를 선택하세요.

ⓘ **경고 알림 > 이벤트 수신 설정** 메뉴에서 프로젝트 멤버와 3rd-party 플러그인에 태그를 설정할 수 있습니다. 자세한 내용은 [다음 문서](#)를 참조하세요.

이벤트 추가하기

기본 제공되는 이벤트 템플릿을 편집해 사용자가 원하는 알림을 추가할 수 있습니다.

1. 화면 오른쪽 위에 **이벤트 추가** 버튼을 선택하세요.
2. 템플릿 목록에서 추가하려는 이벤트 항목을 선택하세요.
3. **이벤트 대상 선택** 및 **이벤트 대상 선택**, **기본 정보 및 수신 설정** 섹션의 옵션을 차례로 설정하세요.
4. 모든 설정을 완료한 다음 화면 오른쪽 위에 **저장** 버튼을 선택하세요.


이벤트 목록에서 새로 추가한 이벤트를 확인할 수 있습니다.

ⓘ • 기본 제공되는 이벤트 템플릿과 구별하여 사용하려면 **이벤트 이름** 항목을 편집해 저장하세요.

- ① 기본 제공하는 이벤트 템플릿에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.
- 이벤트 조건 설정의 각 섹션에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

이벤트 삭제하기


기본 제공된 이벤트 또는 새로 추가한 이벤트를 삭제할 수 있습니다.

- 이벤트 목록에서 삭제하려는 이벤트 항목에서  버튼을 선택하세요.
- 이벤트 수정 화면으로 이동하면 화면 오른쪽 위에 **삭제** 버튼을 선택하세요.
- 삭제 확인 메시지가 나타나면 다시 한번 **삭제** 버튼을 선택하세요.

- ① 삭제한 이벤트는 복구할 수 없습니다.

JSON 형식으로 수정하기

이벤트 설정을 JSON 형식으로 수정할 수 있습니다.

- 화면 오른쪽 위에 **JSON**  버튼을 선택하세요.
- 편집 창이 나타나면 JSON 형식에 맞춰 내용을 수정하세요.
- 수정을 완료하면 화면 오른쪽 위에 **저장** 버튼을 선택하세요.

- ① 수정한 내용이 JSON 형식에 맞지 않으면 화면 아래에 에러 메시지가 표시되며, 저장할 수 없습니다. 표시되는 에러 메시지는 형식에 따라 다를 수 있습니다.

Expected ', or }' after property value in JSON at position 1964 (line 80 column 42)

JSON 데이터의 구조는 다음과 같습니다.

```
{
```

```

"metaId": "infra001",
"displayName": "CPU",
"stateful": true,
"selectCondition": {},
"warningEnabled": true,
"criticalEnabled": true,
"receiver": [],
"warningThreshold": "cpu > 70",
"criticalThreshold": "cpu > 90",
"repeatCount": 1,
"silentPeriod": 60000,
"enabled": false,
"message": "CPU = ${cpu}"
},

```

JSON 데이터의 필드는 이벤트 설정에서 다음 옵션 항목과 연결됩니다.



JSON 필드	옵션
metaId	사용자가 선택한 템플릿의 고유 식별자 값
displayName	이벤트 이름
stateful	해결된 알림
selectCondition	대상 선택
warningEnabled	Warning 이벤트 활성화 여부
criticalEnabled	Critical 이벤트 활성화 여부
receiver	이벤트 수신 > 태그 선택 수신 옵션의 수신 태그 키값 목록
warningThreshold	Warning 이벤트의 임계값 설정
criticalThreshold	Critical 이벤트의 임계값 설정

JSON 필드	옵션
repeatCount	연속 N 회 발생
silentPeriod	일시 중지
enabled	이벤트 활성화
message	메시지

JSON 파일로 공유하기


이벤트 설정을 JSON 파일로 저장해 다른 사용자와 설정을 공유하거나 다른 사용자의 설정을 가져올 수 있습니다.


내보내기

1. 화면 오른쪽 위에 [JSON](#)  버튼을 선택하세요.
2. JSON 편집 창이 나타나면 [내보내기](#)  버튼을 선택하세요.
3. JSON 파일이 다운로드되면 공유할 다른 사용자에게 전달하세요.

 JSON 파일 이름은 event-rules---.json 형식입니다.

가져오기

1. 화면 오른쪽 위에  버튼을 선택하세요.
2. [내보내기](#) 기능을 통해 다운로드한 JSON 파일을 선택하세요.
3. JSON 편집 창이 나타나면 [목록에 추가하기](#) 또는 [덮어쓰기](#) 버튼을 선택하세요.

 이 기능은 같은 종류의 상품 간에 이용할 것을 권장합니다. 다른 상품의 프로젝트로부터 이벤트 설정을 가져올 수는 있지만 정상 작동하지 않습니다.

이벤트 검색하기

이벤트 목록에서 이벤트 이름 또는 지표를 기준으로 검색할 수 있습니다. 검색 입력란에 문자열을 입력한 다음 🔍 버튼을 선택하세요.

메트릭스 경고 알림

홈 화면 > 프로젝트 선택 > 경고 알림 > 이벤트 설정 > 메트릭스 탭 선택

메트릭스 이벤트란?

메트릭스 이벤트는 기본 이벤트(애플리케이션 이벤트, 서버 이벤트 등)보다 구체적이고 복잡한 이벤트를 설정할 때 사용합니다. 프로젝트에서 실시간으로 수집 중인 메트릭스 데이터를 기반으로 이벤트를 설정할 수 있습니다. 사용에 따라 두 가지 설정 방법 중 하나를 선택해 이벤트를 설정할 수 있습니다.

- 메트릭스 이벤트
- 복합 메트릭스 이벤트

ⓘ 메트릭스에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

메트릭스 이벤트

[경고 알림](#) > [이벤트 설정](#) 메뉴에서 화면 위에 [메트릭스](#)를 선택하세요. 화면 오른쪽 위에 [이벤트 추가](#)를 선택하세요. [메트릭스 이벤트](#) 창이 나타납니다.

× 메트릭스 이벤트

이벤트명 *

이벤트 활성화

템플릿 사용 안 함 ▼

카테고리 * 카테고리를 선택해 주세요. ▼

레벨 * Critical Warning Info 이벤트 상태가 해결되면 추가 알림 ⓘ

메시지 * 카테고리를 먼저 선택해 주세요. 🔊

ex. \${active_tx_8} \${time}에 에러가 발생했습니다.

수신 테스트 🔔 수신 테스트

수신 테스트를 이용하려면 필수 항목(이벤트 명, 카테고리, 레벨, 메시지)에 값을 입력하거나 선택하세요.

발생 조건 & 대상 설정

이벤트 발생조건과 대상 입력시 특수문자(~!@#%*&*()_+=[])가 포함되거나 숫자로 시작하는 필드명을 입력하면 오류가 발생할 수 있습니다.
직접 입력 옵션을 선택한 다음 예시와 같이 중괄호({})로 묶어서 입력하세요.

이벤트 발생 조건 * 선택 입력 직접 입력

필드를 선택해 주세요.

>

값

+ 추가

이벤트 대상 필터링 선택 입력 직접 입력

태그를 선택해 주세요.

>

값

+ 추가

🔗 가이드바로가기

▶ 이벤트 수신 설정

▶ 알림 규칙 테스트

저장

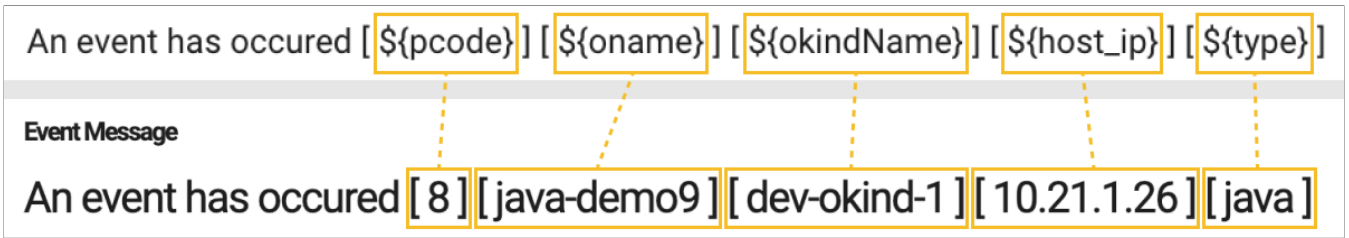
ⓘ 메트릭스에 이벤트 템플릿에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

기본 정보 입력

- **이벤트명**: 추가하려는 이벤트 이름을 입력하세요.
- **이벤트 활성화**: 이벤트를 활성화 여부를 선택하세요.
- **템플릿**: 만들어진 템플릿을 선택해 빠르고 쉽게 이벤트를 설정할 수 있습니다. 템플릿을 사용하지 않을 경우 **사용 안 함**을 선택하세요.
- **카테고리**: 메트릭스 데이터를 구분하는 단위입니다. 메트릭스 이벤트 설정 시 필수 선택 값입니다.

The screenshot shows a form with fields for '카테고리 *', '레벨 *', '메시지 *', and '수신 테스트'. The '카테고리 *' dropdown is open, displaying a search bar and a list of categories. The categories listed are 'AlertCount' with various time intervals (1시간, 5분) and their corresponding keys (alert_count, alert_count(h1), alert_count(m5)). The '직접 입력하기' (Direct Input) option is highlighted.

- **카테고리** 선택 옵션에는 ① 이름과 ② 데이터 수집 간격, ③ 키 정보를 표시합니다. 이벤트 설정 시 해당 카테고리의 키 값을 사용합니다.
 - **카테고리**는 최근 3시간 범위 내 프로젝트에서 수집 중인 메트릭스 데이터를 조회해 목록에 표시합니다. **카테고리** 선택 옵션에 수집 간격이 표시되지 않는 경우 **직접 입력하기** 옵션을 선택해 카테고리 키를 입력할 수 있습니다.
- **레벨**
 - 이벤트 발생 시 경고 수준을 나타냅니다. **Critical**, **Warning**, Info 수준으로 나눕니다. **Critical**, **Warning** 레벨 설정 시 **이벤트 상태가 해결되면 추가 알림** 선택 옵션이 활성화됩니다.
 - **이벤트 상태가 해결되면 추가 알림**: 이벤트 항목 중 발생한 이벤트 상태가 해결되면 추가 알림 송신 여부를 선택할 수 있습니다. 토글 버튼을 선택해 기능을 켜거나 끌 수 있습니다.
- **메시지**
 - 이벤트 발생 시 출력하는 알림 메시지를 입력합니다. `{Tag}` 또는 `{Field}` 입력으로 메시지에 변수를 적용할 수 있습니다. 변수에 입력할 키는 선택한 메트릭스 데이터 **카테고리**에 포함된 값이어야 합니다. **메트릭스 조회** 메뉴에서 입력할 수 있는 태그 또는 필드키를 확인할 수 있습니다.



ⓘ 버튼을 클릭하면 이전에 입력한 메시지 기록을 확인할 수 있습니다.

수신 테스트

필수 항목인 이벤트명, 카테고리, 레벨, 메시지 정보를 기준으로 알림을 발생시켜 메시지를 점검하는 기능입니다.

ⓘ 수신 테스트를 이용하려면 필수 항목(이벤트명, 카테고리, 레벨, 메시지)에 값을 입력하거나 선택해야 합니다.

이벤트 발생 조건

이벤트 발생 조건 *

선택 입력 직접 입력

4 cpu > 70 && mem > 80

5 6

cpu	Cpu (pct) ▾	>	70	
&& ▾	mem	Memory (pct) ▾	>	80

+ 추가

4 필드, 5 연산자 선택, 6 임계값을 입력해 이벤트 발생 조건을 설정하세요.

이벤트 대상 필터링

이벤트 대상 필터링

선택 입력 직접 입력

7 8 9

태그를 선택해 주세요. ▾	>	값
----------------	---	---

+ 추가

7 태그, 8 연산자 선택, 9 필터링값을 입력해 대상을 필터링합니다. 입력값이 없을 경우 전체 에이전트를 대상으로 경고 알림을 보냅니다.

- ⓘ 이벤트 발생 조건과 이벤트 대상 필터링에서 사용할 수 있는 기본 문법과 연산자 목록은 [다음 문서](#)를 참조하세요.
- 이벤트 발생 조건과 이벤트 대상 필터링 옵션은 선택 입력 또는 직접 입력 옵션을 선택할 수 있습니다.

- ① 이벤트 설정 내용이 저장된 이후에는 해당 옵션값은 **직접 입력** 옵션으로 관리합니다. 이후 **선택 입력** 옵션으로 전환하면 옵션값이 초기화될 수 있습니다.
- 이벤트 발생 조건과 대상 입력 시 특수 문자(`~!@#%&^&*()_+=[]`)를 포함하거나 숫자로 시작하는 필드명을 입력하면 오류가 발생할 수 있습니다. 이런 경우 **직접 입력** 옵션을 선택한 다음 예시와 같이 중괄호(`{ }`)로 묶어서 입력하세요.

```

    ${4xxErrorType} == '401'
```

이벤트 수신 설정

▼ 이벤트 수신 설정

발생 횟수 최근 동안 회 발생

선택 시간 동안 설정한 이벤트가 입력 횟수만큼 발생할 때 알림을 수신합니다.
 선택 시간이 "사용 안 함"인 경우에는 지정한 횟수만큼 연속적으로 발생할 때 알림을 수신합니다.
 "이벤트 상태가 해결되면 추가 알림"을 사용하는 경우, 선택 시간은 "사용 안 함"을 권장합니다.
 선택한 카테고리의 수집 주기는 5초입니다.

이벤트 발생 일시 중지

알림 수신 후 선택한 시간 동안 이벤트가 발생하지 않습니다.
 단, "이벤트 상태가 해결되면 추가 알림" 기능을 활성화한 경우에는 RECOVERED 알림 수신 후 선택한 시간 동안 이벤트가 발생하지 않습니다.

이벤트 수신 태그 전체 멤버 수신

이벤트 설정 시 이벤트 수신 태그를 선택하여 해당 태그를 가진 프로젝트 멤버와 3rd-party 플러그인에 알림을 전송할 수 있습니다.
 이벤트 수신 설정 메뉴에서 프로젝트 멤버와 3rd-party 플러그인에 각각 태그를 지정할 수 있습니다

[🔗 프로젝트 이벤트 수신설정 메뉴](#)

이벤트 설정 시 태그를 선택하지 않은 경우 프로젝트 이벤트 수신 설정 메뉴의 나머지 수신 조건(활성화 여부 등)에 따라 알림이 발생합니다.

- 발생 횟수:** 선택한 시간 동안 **이벤트 발생 조건**에서 설정한 이벤트가 입력 횟수만큼 발생하면 경고 알림을 보냅니다.

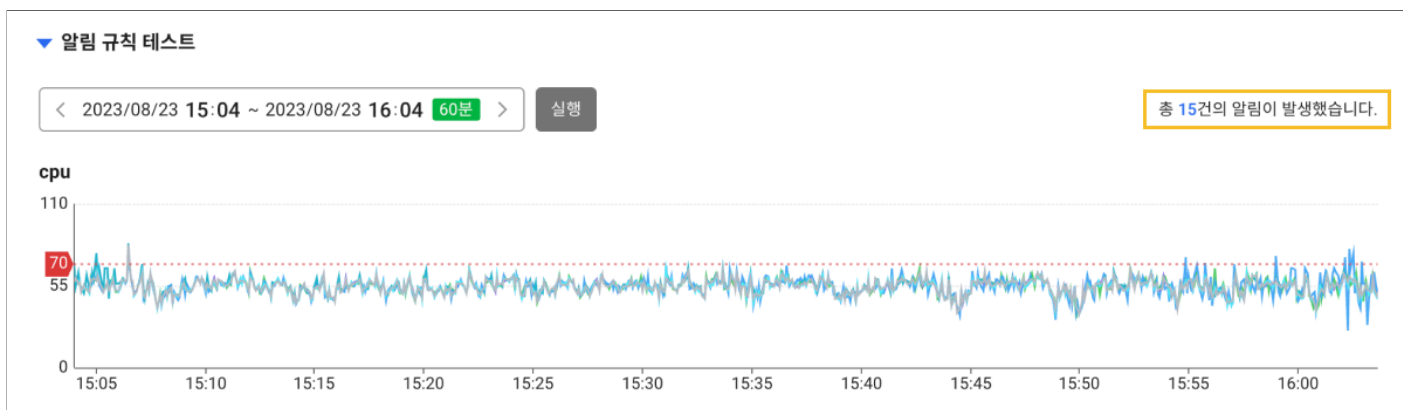
- ①
 - 선택 시간을 **사용 안 함**으로 설정하면 입력한 횟수만큼 연속 발생할 때 알림을 보냅니다.
 - 이벤트 상태가 해결되면 추가 알림** 옵션을 활성화한 경우 선택 시간은 **사용 안 함**으로 선택할 것을 권장합니다.
 - 카테고리** 옵션에서 선택한 항목의 수집 주기는 5초입니다.

- 이벤트 발생 일시 중지:** 과도한 경고 알림 발생을 방지할 수 있는 옵션입니다. 첫번째 경고 알림 이후 선택한 시간 동안 경고 알림을 보내지 않습니다. 또한 **이벤트 기록** 메뉴에 기록되지 않습니다.

- **관련 카테고리:** 관련 카테고리를 5개까지 설정하고 알림 조회 시 참조합니다.
- **이벤트 수신 태그:** 이벤트 수신 태그를 선택하면 해당 태그를 가진 프로젝트 멤버와 3rd-party 플러그인에 알림을 전송할 수 있습니다. 이벤트 수신 태그를 선택하지 않으면 프로젝트 전체 멤버에게 경고 알림을 보냅니다.

ⓘ 경고 알림 > 이벤트 수신 설정 메뉴에서 프로젝트 멤버와 3rd-party 플러그인에 태그를 설정할 수 있습니다.

알림 규칙 테스트



선택한 시간 동안 설정한 이벤트 조건을 실행해 몇 번의 경고 알림이 발생했는지 확인할 수 있습니다. 실행 버튼을 선택하면 알림 발생 건수 정보를 알 수 있으며, 이벤트 발생 조건에서 선택한 필드와 임계치를 차트상에 표시합니다.

복합 메트릭스 이벤트

복합 메트릭스 이벤트를 이용하려면 다음의 개념에 대한 이해가 필요합니다.

- [메트릭스란?](#)
- [MXQL](#)

복합 메트릭스 이벤트는 메트릭스 데이터에 보다 복잡한 규칙을 활용해 이벤트를 생성하고 경고 알림을 보낼 수 있습니다. 복합 메트릭스는 다음과 같은 상황에서 효과적으로 사용할 수 있습니다.

- 여러 에이전트에서 수신된 데이터에 대해 종합적인 이벤트 판정을 해야할 때
- 과거 데이터와 현재 데이터를 비교해 이벤트 판정을 해야할 때

메트릭스 이벤트는 에이전트로부터 메트릭스를 수신할 때마다 이벤트 판정을 합니다. 반면, 복합 메트릭스 이벤트는 각 에이전트에서 수집한 메트릭스들을 데이터베이스에 저장합니다. 그리고 다시 조회해서 이벤트 판정을 합니다. 이와 같은 특성 때문에 여러

에이전트의 데이터를 종합적으로 활용하거나 과거의 데이터를 활용할 수 있습니다. 하지만 **MXQL**이라는 와탭 고유의 데이터 조회 언어를 사용해야한다는 진입장벽이 존재합니다. 따라서 사용자들이 기초적인 **MXQL**만 이해하더라도 효과적으로 이벤트를 설정할 수 있도록 이벤트 템플릿을 제공합니다. MXQL 기초 사용자는 이벤트 대상 필터링과 이벤트 조건에 대한 쿼리만 수정해서 이벤트를 적용할 수 있습니다.

1. 경고 알림 > 이벤트 설정 메뉴에서 화면 위에 **메트릭스**를 선택하세요.
2. **복합 메트릭스** 섹션에서 오른쪽에 **이벤트 추가**를 선택하세요.
3. **복합 메트릭스** 창이 나타나면 **차트로 생성하기**를 선택하세요.

이벤트 설정 창이 나타납니다.

이벤트 설정

알림

이벤트 활성화

레벨 위험 경고 정보 이벤트 상태가 해결되면 추가 알림

제목

메시지

이벤트 발생 조건

데이터 조회 범위 1분

조건 [HttpCallCount]
이벤트 데이터 조회에 포함된 필드만 사용 가능합니다

[> 추가정보](#)

이벤트 규칙 테스트

2023/08/22 실행

이벤트 데이터 조회

위젯 텍스트

< 2023/08/23 16:41 ~ 2023/08/23 16:51 10분 >

카테고리 [선택된 카테고리]

필터 + 필터 추가

그룹화 [0id (pk) X] 타임 유닛 5 초 ↓

필드 [HttpCallCount X]

name	sum	avg	max	recently
633280970	10285	85.70833333333333	122	81
1387800924	10363	86.35833333333333	121	101
-1128904592	10824	90.2	117	100
-1143239575	10038	83.65	124	107
-857948929	10388	86.56666666666666	115	102
-877561626	10278	85.65	127	102

저장

ⓘ 복합 메트릭스 이벤트를 설정하려면 **이벤트 설정** 권한이 있어야 합니다.

이벤트 데이터 조회

복합 메트릭스 이벤트는 메트릭스 데이터 질의 언어인 **MXQL**을 기반으로 이벤트 조건을 생성합니다. [차트로 생성하기](#) 기능은 **MXQL**의 자동완성을 위한 콤보박스 기능을 제공합니다. 이벤트 데이터를 조회하여 차트를 구성한 다음 이벤트 발행 조건을 직접 입력하기 위한 템플릿입니다. [위젯](#) 또는 [텍스트](#) 옵션을 선택해 이벤트를 설정하세요.

[위젯](#) | [텍스트](#)

시계열 차트를 구성하는 옵션을 통해 이벤트 설정 시 사용할 **MXQL**을 자동완성할 수 있습니다.

이벤트 데이터 조회

위젯 텍스트

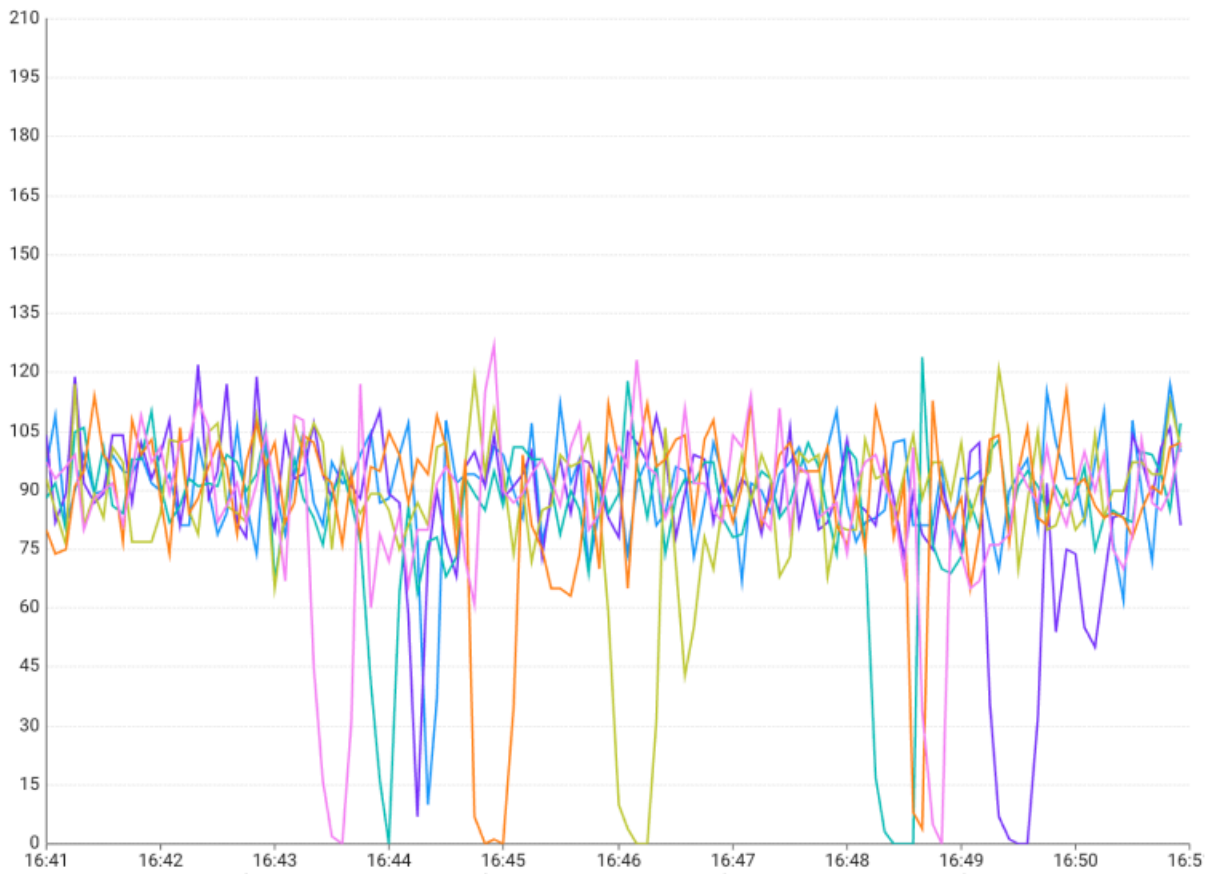
< 2023/08/23 16:41 ~ 2023/08/23 16:51 10분 >

카테고리

필터 + 필터 추가

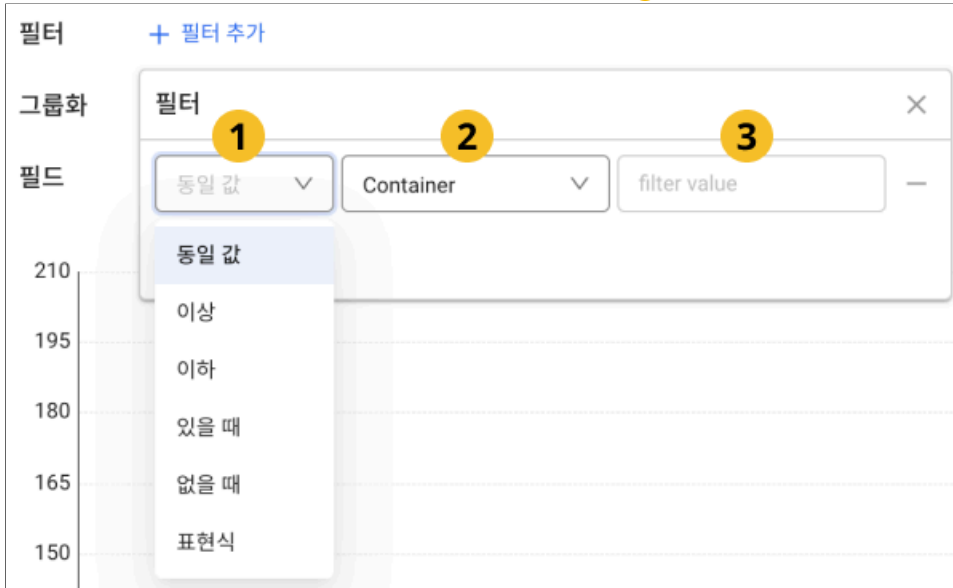
그룹화 타임 유닛 5 초

필드



name	sum	avg	max	recently
633280970	10285	85.70833333333333	122	81
1387800924	10363	86.35833333333333	121	101
-1128904592	10824	90.2	117	100

- **필터**: 이벤트 조건 대상을 선택합니다. ① 연산식, ② 태그, ③ 필터링값을 입력해 필터링 조건을 생성합니다.



- **그룹화**: 그룹화된 메트릭스 데이터를 선택합니다. 다중 선택할 수 있습니다.
- **타임 유닛**: 그룹화된 데이터를 나눌 시간 기준을 설정합니다. 초, 분, 시간 단위로 선택하고 설정할 수 있습니다.
- **필드**: 이벤트 발행 조건에 사용할 필드를 선택합니다. 다중 선택할 수 있습니다.

MXQL을 평문 그대로 수정할 수 있는 편집창이 나타납니다.

이벤트 데이터 조회

위젯 텍스트

```

INJECT timepast
HEADER { "httpc_count$":"COUNT" }
OIDSET { oid:$oid, okind:$okind, onode:$onode }
CATEGORY {"app_counter":6h, "app_counter{m5}":3d, "app_counter{h1}":unlimit }
TAGLOAD
INJECT default
UPDATE {key: httpc_count, value: avg}
FILTER {key: container, exist: true}
GROUP {pk:[oid], timeunit: 5000}
UPDATE {key: httpc_count, value: avg}
CREATE {key: _id_, expr:"oid"}
CREATE {key: _name_, expr:"oname"}
SELECT [_name_, _id_, time, oid, httpc_count]
    
```

조회

< 2023/08/23 16:51 ~ 2023/08/23 17:01 10분 >



httpc_count	_name_	_id_	time	oid
89	demo-8105	-857948929	2023/08/23 16:51:00	-857948929
82	demo-8100	-1128904592	2023/08/23 16:51:00	-1128904592
98	demo-8101	-877561626	2023/08/23 16:51:00	-877561626
81	demo-8102	1387800924	2023/08/23 16:51:00	1387800924
90	demo-8104	-1143239575	2023/08/23 16:51:00	-1143239575
107	demo-8103	633280970	2023/08/23 16:51:00	633280970
107	demo-8105	-857948929	2023/08/23 16:51:05	-857948929
85	demo-8100	-1128904592	2023/08/23 16:51:05	-1128904592
97	demo-8101	-877561626	2023/08/23 16:51:05	-877561626
94	demo-8102	1387800924	2023/08/23 16:51:05	1387800924
86	demo-8104	-1143239575	2023/08/23 16:51:05	-1143239575
85	demo-8103	633280970	2023/08/23 16:51:05	633280970
88	demo-8105	-857948929	2023/08/23 16:51:10	-857948929
6	demo-8100	-1128904592	2023/08/23 16:51:10	-1128904592
79	demo-8101	-877561626	2023/08/23 16:51:10	-877561626
84	demo-8102	1387800924	2023/08/23 16:51:10	1387800924
77	demo-8104	-1143239575	2023/08/23 16:51:10	-1143239575
70	demo-8103	633280970	2023/08/23 16:51:10	633280970
86	demo-8105	-857948929	2023/08/23 16:51:15	-857948929
5	demo-8100	-1128904592	2023/08/23 16:51:15	-1128904592

이전

페이지 1 of 4

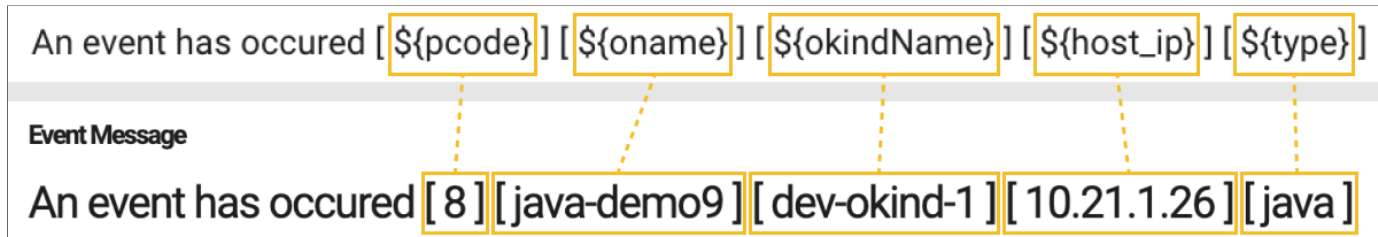
다음

저장

알림

경고 알림 설정의 기본 정보를 입력합니다.

- **이벤트 활성화:** 토글 버튼을 클릭해 이벤트를 활성화할 여부를 선택할 수 있습니다.
- **레벨:** 위험(Critical), 경고(Warning), 정보 수준 중 하나의 레벨을 선택하세요. **이벤트 상태가 해결되면 추가 알림:** 이벤트 항목 중 발생한 이벤트 상태가 해결되면 추가 알림 송신 여부를 선택할 수 있습니다. 토글 버튼을 선택해 기능을 켜거나 끌 수 있습니다.
- **제목:** 경고 알림의 제목을 입력하세요.
- **메시지:** 이벤트 발생 시 출력하는 알림 메시지를 입력합니다. `${Tag}` 또는 `${Field}` 입력으로 메시지에 변수를 적용할 수 있습니다. 변수에 입력할 키는 선택한 메트릭스 데이터 **카테고리**에 포함된 값이어야 합니다. **메트릭스 조회** 메뉴에서 입력할 수 있는 태그 또는 필드키를 확인할 수 있습니다.



알림 정책

경고 알림을 보낼 조건을 입력합니다.

- **데이터 조회 범위:** 이벤트 조건에 사용할 **MXQL**의 실시간 데이터 조회 범위를 설정합니다. 이벤트 데이터 조회에 포함된 필드만 사용할 수 있습니다.
복합 메트릭스 이벤트는 DB에 저장된 메트릭스를 조회해서 활용합니다. 따라서 데이터를 조회할 시간 범위를 먼저 지정해야 합니다. 데이터 조회 시간을 5분으로 선택하면 최근 5분동안 수집된 데이터를 조회해서 이벤트 발생 조건을 확인합니다. 최근 데이터에 대해서 이벤트를 설정할 때에는 짧게, 넓은 시간에 대해서 통계적으로 접근하고 싶을 때에는 길게 설정할 수 있습니다. 실제 사용 예시는 [다음 문서](#)를 참조하세요.
- **조건:** MXQL에 반영한 필드와 연산 규칙, 임계치를 입력합니다.

부가 정보

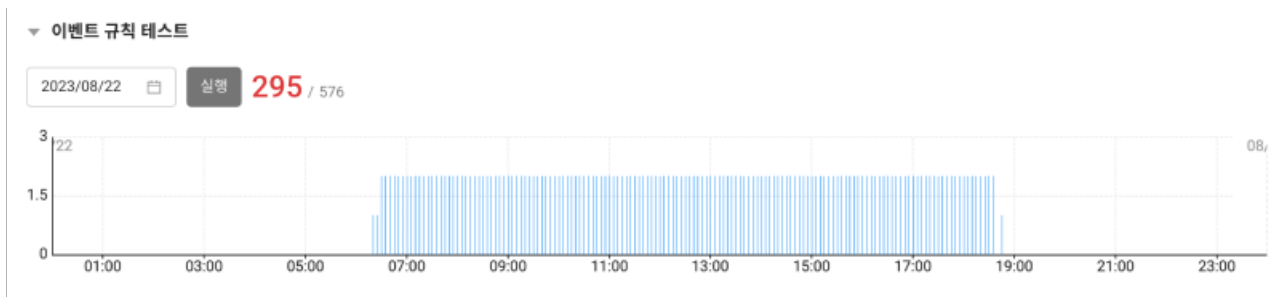
경고 알림 수신과 관련한 부가적인 옵션을 설정합니다.

- **인터벌:** 선택한 시간 간격으로 알림 조건을 확인합니다.
- **무음:** 과도한 경고 알림 발생을 방지할 수 있는 옵션입니다. 첫번째 경고 알림 이후 선택한 시간 동안 경고 알림을 보내지 않습니다. 또한 **이벤트 기록** 메뉴에 기록되지 않습니다.

- **이벤트 수신 태그:** 이벤트 수신 태그를 선택하면 해당 태그를 가진 프로젝트 멤버와 3rd-party 플러그인에 알림을 전송할 수 있습니다. 이벤트 수신 태그를 선택하지 않으면 프로젝트 전체 멤버에게 경고 알림을 보냅니다.

ⓘ 경고 알림 > 이벤트 수신 설정 메뉴에서 프로젝트 멤버와 3rd-party 플러그인에 태그를 설정할 수 있습니다.

이벤트 규칙 테스트



선택한 시간 동안 설정한 이벤트 조건을 실행해 몇 번의 경고 알림이 발생했는지 확인할 수 있습니다. 실행 버튼을 선택하면 알림 발생 건수 정보를 알 수 있으며, 이벤트 발생 조건에서 선택한 필드와 임계치를 차트상에 표시합니다.

이벤트 설정에 포함된 대부분의 내용들이 **MXQL**을 사용해서 지정됩니다. **MXQL**이 적절하게 작성되었는지 시뮬레이션할 수 있는 기능을 제공합니다. 시뮬레이션 기능은 과거의 24시간 데이터를 조회해서 이벤트 판정을 한 다음 몇 건의 메트릭스가 조회되었고 그 중 몇 건에서 이벤트 판정이 성공했는지 알려줍니다.

메트릭스 이벤트 수정 및 삭제

1. 경고 알림 > 이벤트 설정 메뉴로 이동한 다음 **메트릭스** 탭을 선택하세요.
2. 이벤트 목록에서 수정 또는 삭제하려는 항목의 가장 오른쪽에 버튼을 선택하세요.
3. 메트릭스 또는 복합 메트릭스 이벤트 설정 창이 나타나면 각 옵션을 수정한 다음 **저장** 버튼을 선택하세요.
 선택한 이벤트를 삭제하려면 이벤트 설정 창의 오른쪽 위에 **삭제** 버튼을 선택하세요.

발생 조건, 대상 선택 가이드

메트릭스 경고 알림의 이벤트 발생 조건과 이벤트 대상 선택은 동일한 문법을 사용합니다. 단, 이벤트 발생 조건은 태그(Tag)의 Key를 변수로 사용하고, 이벤트 대상 선택은 필드(Field)의 Key를 변수로 사용합니다.

기본 문법

- 문자열을 그냥 입력하면 변수, 작은 따옴표(') 또는 큰 따옴표("")로 감싸면 text로 인식합니다.

```
oid == "oid"
```

- oid : 변수
- == : 함수
- "oid" : text

```
// oname가 ott-1235일 경우
```

```
// 정상적인 경우
```

```
onname = 'ott-1235' 또는 onname = "ott-1235"
```

```
// 비정상적인 경우, 알림이 동작하지 않습니다.
```

```
onname = ott-1235
```

- 숫자를 그냥 입력하면 number, 작은 따옴표(') 또는 큰 따옴표("")로 감싸면 text로 인식합니다.

```
oid == 123
```

- oid : 변수
- == : 함수
- 123 : number

```
// oid가 123일 경우
```

```
// 정상적인 경우
```

```
oid = 123
```

```
// 비정상적인 경우, 알림이 동작하지 않습니다.
```

```
id == '123' 또는 oid == "123"
```

사용 가능한 연산자 목록

연산자	사용법	설명
==	operand1 == operand2	operand1과 operand2의 값이 동일한지 확인합니다.
!=	operand1 != operand2	operand1과 operand2의 값이 다른지 확인합니다.
>	operand1 > operand2	operand1의 값이 operand2의 값보다 큰지 확인합니다.
>=	operand1 >= operand2	operand1의 값이 operand2의 값보다 크거나 같은지 확인합니다.
<	operand1 < operand2	operand1의 값이 operand2의 값보다 작은지 확인합니다.
<=	operand1 <= operand2	operand1의 값이 operand2의 값보다 작거나 같은지 확인합니다.
like	operand1 like operand2	operand1에 operand2가 포함되어 있는지를 패턴으로 검색합니다.
&&	expression1 && expression2	expression1과 expression2이 모두 true인지 확인합니다.
and	expression1 and expression2	expression1과 expression2이 모두 true인지 확인합니다. &&와 동일한 역할을 수행하는 연산자입니다.
	expression1 expression2	expression1 또는 expression2이 true인지 확인합니다.
or	expression1 or expression2	expression1 또는 expression2이 true인지 확인합니다. 와 동일한 역할을 수행하는 연산자입니다.

like 사용법

와일드카드(*****)를 통해 포함 문자열을 편리하게 검색할 수 있습니다.

- 특정 키워드로 시작하는 문자열 검색

Key like "Value*"

- 특정 키워드로 끝나는 문자열 검색

```
Key like "*Value"
```

- 특정 키워드가 포함된 문자열 검색

```
Key like "*Value*"
```

- 키워드 중간에 와일드카드(`*`)를 사용할 수 없습니다.

```
// 지원하지 않는 문법
Key like "Va*lue"
```

- `like` 연산자에서 와일드카드(`*`)를 생략하는 경우 `equals(==)`로 동작합니다.

```
// 아래의 두 문장은 완전히 같은 결과를 가집니다.
Key like "Value"
Key == "Value"
```

사용 가능한 함수 목록

함수	사용법	설명
startsWith	<code>startsWith(param1, param2)</code>	param1을 Key로 하는 Value가 param2로 시작하면 <code>true</code> , 반대의 경우 <code>false</code>

함수	사용법	설명
endsWith	endsWith(param1, param2)	param1을 Key로 하는 Value가 param2로 끝나면 true, 반대의 경우 false
isNull	isNull(param1)	param1이 null이면 true, 반대의 경우 false
isNotNull	isNotNull(param1)	param1이 null이 아니면 true, 반대의 경우 false
isEmpty	isEmpty(param1)	param1이 null 또는 EmptyString("") 이면 true, 반대의 경우 false
isNotEmpty	isNotEmpty(param1)	param1이 null도 아니고 EmptyString("") 도 아니면 true, 반대의 경우 false

startsWith

```
startsWith(Key, "Value")
```

endsWith

```
endsWith(Key, "Value")
```

isNull

```
isNull(Key)
```

isNotNull

```
isNotNull(Key)
```

isEmpty

```
isEmpty(Key)
```

isNotEmpty

```
isNotEmpty(Key)
```

메트릭스 이벤트 템플릿

❗ 쿠버네티스 이벤트의 `reason` 필드 정보는 [쿠버네티스 공식 문서](#)를 참조하세요.

BackOff

Kubernetes 이벤트의 `reason` 필드 정보 중 `BackOff`가 0회 이상 찍힌 경우 해당 알림이 발생합니다. [메시지](#) 예시는 다음과 같습니다.

```
쿠버네티스 이벤트 메시지(Kube Event  ${message} )
```

Evicted

Kubernetes 이벤트의 `reason` 필드 정보 중 `Evicted`가 0회 이상 찍힌 경우 해당 알림이 발생합니다. [메시지](#) 예시는 다음과 같습니다.

```
쿠버네티스 이벤트 메시지(Kube Event  ${message} )
```

FailedCreatePodSandBox

Kubernetes 이벤트의 `reason` 필드 정보 중 `FailedCreatePodSandBox`가 0회 이상 찍힌 경우 해당 알림이 발생합니다. [메시지](#) 예시는 다음과 같습니다.

```
쿠버네티스 이벤트 메시지(Kube Event  ${message} )
```

FailedMount

Kubernetes 이벤트의 `reason` 필드 정보 중 `FailedMount` 가 0회 이상 찍힌 경우 해당 알림이 발생합니다. [메시지](#) 예시는 다음과 같습니다.

```
쿠버네티스 이벤트 메시지(Kube Event message)
```

FailedScheduling

Kubernetes 이벤트의 `reason` 필드 정보 중 `FailedScheduling` 이 0회 이상 찍힌 경우 해당 알림이 발생합니다. [메시지](#) 예시는 다음과 같습니다.

```
쿠버네티스 이벤트 메시지(Kube Event message)
```

FailedSync

Kubernetes 이벤트의 `reason` 필드 정보 중 `FailedSync` 가 0회 이상 찍힌 경우 해당 알림이 발생합니다. [메시지](#) 예시는 다음과 같습니다.

```
쿠버네티스 이벤트 메시지(Kube Event message)
```

NodeNotReady

Kubernetes 이벤트의 `reason` 필드 정보 중 `NodeNotReady` 가 0회 이상 찍힌 경우 해당 알림이 발생합니다. [메시지](#) 예시는 다음과 같습니다.

```
쿠버네티스 이벤트 메시지(Kube Event message)
```

Unhealthy

Kubernetes 이벤트의 `reason` 필드 정보 중 `Unhealthy` 가 0회 이상 찍힌 경우 해당 알림이 발생합니다. [메시지](#) 예시는 다음과 같습니다.

```
쿠버네티스 이벤트 메시지(Kube Event message)
```

컨테이너 CPU 할당량 기준 사용률 알림

컨테이너의 CPU Limit 기준 CPU 전체 사용량(`cpu_per_quota`)이 70% 이상인 경우 해당 알림이 발생합니다. [메시지](#) 예시는 다음과 같습니다.

`${okindName}`의 `${oname}` 컨테이너의 CPU 사용률이 `${cpu_per_quota}` % \geq 70%로 높습니다.

컨테이너 메모리 Fail Count

컨테이너 메모리 Limit 도달 건수가 1회 이상인 경우 해당 알림이 발생합니다. [메시지](#) 예시는 다음과 같습니다.

`${okindName}`의 `${oname}` 컨테이너가 limit을 초과하여 `${mem_failcnt}`가 증가했습니다.

컨테이너 메모리 사용률

컨테이너 메모리 Limit 기준 사용률(`${container.mem_percent}`)이 90% 이상이 경우 해당 알림이 발생합니다. [메시지](#) 예시는 다음과 같습니다.

`${okindName}`의 `${oname}` 컨테이너 메모리 사용률이 `${container.mem_percent}` % \geq 90% 입니다.

컨테이너 DEAD 상태 알림

컨테이너 상태 코드가 100 일 경우 해당 알림이 발생합니다. 상태 코드 100은 DEAD를 의미합니다. [메시지](#) 예시는 다음과 같습니다.

`${oname}` 컨테이너가 DEAD 상태 입니다.

클러스터 CPU Request 알림

노드 할당 가능한 CPU량을 Limit CPU 총합으로 나눈 값에 100을 곱한 값이 80% 이상이면 해당 알림이 발생합니다. [메시지](#) 예시는 다음과 같습니다.

클러스터 CPU 할당량 대비 CPU Request(최소 요구자원)가 80% 이상입니다.

클러스터 Memory Request 알림

노드 할당 가능한 메모리량을 Limit Memory 총합으로 나눈 값에 100을 곱한 값이 80% 이상이면 해당 알림이 발생합니다. [메시지](#) 예시는 다음과 같습니다.

클러스터 Memory 할당량 대비 Memory Request(최소 요구자원)가 80% 이상입니다.

클러스터 CPU Request 알림

노드 할당 가능한 CPU량을 Limit CPU 총합으로 나눈 값에 100을 곱한 값이 60% 이상이면 해당 알림이 발생합니다. [메시지](#) 예시는 다음과 같습니다.

클러스터 CPU 할당량 대비 CPU Request(최소 요구자원)가 60% 이상입니다.

클러스터 Memory Request 알림

노드 할당 가능한 메모리량을 Limit Memory 총합으로 나눈 값에 100을 곱한 값이 60% 이상이면 해당 알림이 발생합니다. [메시지 예시](#)는 다음과 같습니다.

클러스터 Memory 할당량 대비 Memory Request(최소 요구자원)가 60% 이상입니다.

클러스터 POD수 알림

노드에 할당 가능한 Pod가 존재하지 않을 경우 해당 알림이 발생합니다. [메시지 예시](#)는 다음과 같습니다.

마스터에 할당 가능한 POD 개수는 0입니다.

노드 CPU 사용률 알림

노드의 CPU 사용률(`{cpu}`)이 70% 이상일 경우 해당 알림이 발생합니다. [메시지 예시](#)는 다음과 같습니다.

`{oname}`의 CPU 사용률이 `{cpu}` % >= 70% 입니다.

노드 메모리 사용률 알림

노드의 메모리 사용률(`{memory_pused}`)이 90% 이상일 경우 해당 알림이 발생합니다. [메시지 예시](#)는 다음과 같습니다.

`{oname}`의 메모리 사용률이 `{memory_pused}` % >= 90% 입니다.

할당 불가능 노드 알림

노드에 할당 가능한 Pod 수가 0개 이하일 경우 알림이 발생합니다.

APDEX

트랜잭션이 존재하면서 APDEX 점수가 0.7점보다 낮을 경우 해당 알림이 발생합니다. [메시지 예시](#)는 다음과 같습니다.

노드에 할당 가능한 Pod 수가 0개 이하일 경우 해당 알림이 발생합니다. [메시지 예시](#)는 다음과 같습니다.

APDEX가 0.7 보다 작습니다 (`{oname}`)

알림 메시지 사용자 정의

홈 화면 > 프로젝트 선택 > 경고 알림 > 이벤트 설정 > 메트릭스 탭 선택

프로젝트 멤버에게 메일 또는 서드 파티 플러그인을 통해 전달하는 경고 알림 메시지를 수정할 수 있는 사용자 정의 기능을 제공합니다. 화면 오른쪽 위에 **알림 메시지 설정** 버튼을 선택하세요. 수신할 메시지에서 불필요하다고 생각되는 항목을 감추거나 사용자가 원하는 특별한 내용을 추가할 수 있습니다.

이벤트 설정

애플리케이션 **메트릭스** 히트맵 패턴 이상치 탐지 로그

메트릭스 JSON 알림 메시지 설정 + 이벤트 추가

이벤트 이름	카테고리	대상	규칙	이벤트 상태가 해결되면 추가 알림	이벤트 수신 태그	설정
메트릭스 값이 임계값을 초과할 때	app_proc_counter	*	gc_oldgen_count >= 1	Off	이벤트 수신 태그 전체 멤버 수신	<input checked="" type="checkbox"/>
신규 이벤트를 생성하는 멤버가 갑자기 증가하거나 불필요하게 이벤트가 생성되는 경우	app_counter 5분	endsWith(okindN..)	(write_time > 3000 read_time > 30...	Off	이벤트 수신 태그 A팀 메시지	<input type="checkbox"/>

복합 메트릭스 JSON 알림 메시지 설정 + 이벤트 추가

이벤트 이름	조건	이벤트 상태가 해결되면 추가 알림	이벤트 수신 태그	설정
TPS has changed by more than 50% compared to the previous week	one_week_diff_abs > current_tps * 0.3	On	이벤트 수신 태그 전체 멤버 수신	<input type="checkbox"/>

- ① 이 기능은 **메트릭스**, **복합 메트릭스** 경고 알림에만 적용할 수 있습니다.
- 이 기능은 프로젝트 선택 > **실험실** > **알림 메시지** 설정 메뉴 경로로도 진입할 수 있습니다.
- 이 기능은 **알림 설정** 권한을 가진 멤버가 설정할 수 있습니다. 자세한 내용은 [다음 문서](#)를 참조하세요.

기본 화면 안내

알림 메시지

알림 메시지
수신할 메시지의 상세 항목을 선택하고, 항목 별 내용을 편집할 수 있습니다.

다른 프로젝트에 복사 전체 초기화 전체 저장

전체 펼치기 발생 이벤트 발생 메시지 해결 이벤트 해결 메시지

전체 발생 해결 검색

기본 설정 초기화 미리보기 저장 전체 활성화

1	메트릭스 발생	상태	ex) 상태:OCCURED/SOLVED	3	4	5
		알림 레벨	ex) 알림 레벨:Warning		수정	ON
		프로젝트 번호	ex) 프로젝트 번호:101		수정	ON
		프로젝트 이름	ex) 프로젝트 이름:Example Project		수정	ON
		애플리케이션 이름	ex) 애플리케이션 이름:Whatap-Front-01		수정	ON
		메시지	ex) 메시지:Message		수정	ON
		이벤트 발생 시각	ex) 이벤트 발생 시각:2022-09-08 14:23:17 +0900		수정	ON
		이벤트 발생 조건	ex) 이벤트 발생 조건:tps > 20		수정	ON
		이벤트 대상 필터링	ex) 이벤트 대상 필터링:oid == 800488350		수정	ON
		해결된 이벤트 알림 기능	ex) 해결된 이벤트 알림 기능:ON/OFF		수정	ON
		알림 종류	ex) 알림 종류:METRICS		수정	ON
		알림 라벨	ex) 알림 라벨:oid(800488350)		수정	ON
		지표	ex) 지표:tps		수정	ON
		현재값	ex) 현재값:25		수정	ON
		임계값	ex) 임계값:20		수정	ON
		현재값(복수)	ex) 현재값(복수):tps=23.39, tx_error=7		수정	ON
		반복 횟수	ex) 반복 횟수:5		수정	ON
		반복 시간	ex) 반복 시간:5 minutes		수정	ON
		무음 시간	ex) 무음 시간:10 minutes		수정	ON

메트릭스 해결 기본 설정 초기화 미리보기 저장 전체 활성화

복합 메트릭스 발생 기본 설정 초기화 미리보기 저장 전체 활성화


복합 메트릭스 해결 기본 설정 초기화 미리보기 저장 전체 활성화

- 알림 메시지의 종류는 **발생** 과 **해결** 로 분류할 수 있습니다. **발생** 은 이벤트가 발생했을 때 전달되는 알림이며, **해결** 은 이벤트가 해소되었을 때 전달되는 알림입니다.
- 1 영역은 수신할 메시지의 상세 항목입니다. 2 영역은 수신할 메시지의 상세 항목에 대한 내용입니다. 3 **미리보기** 버튼을

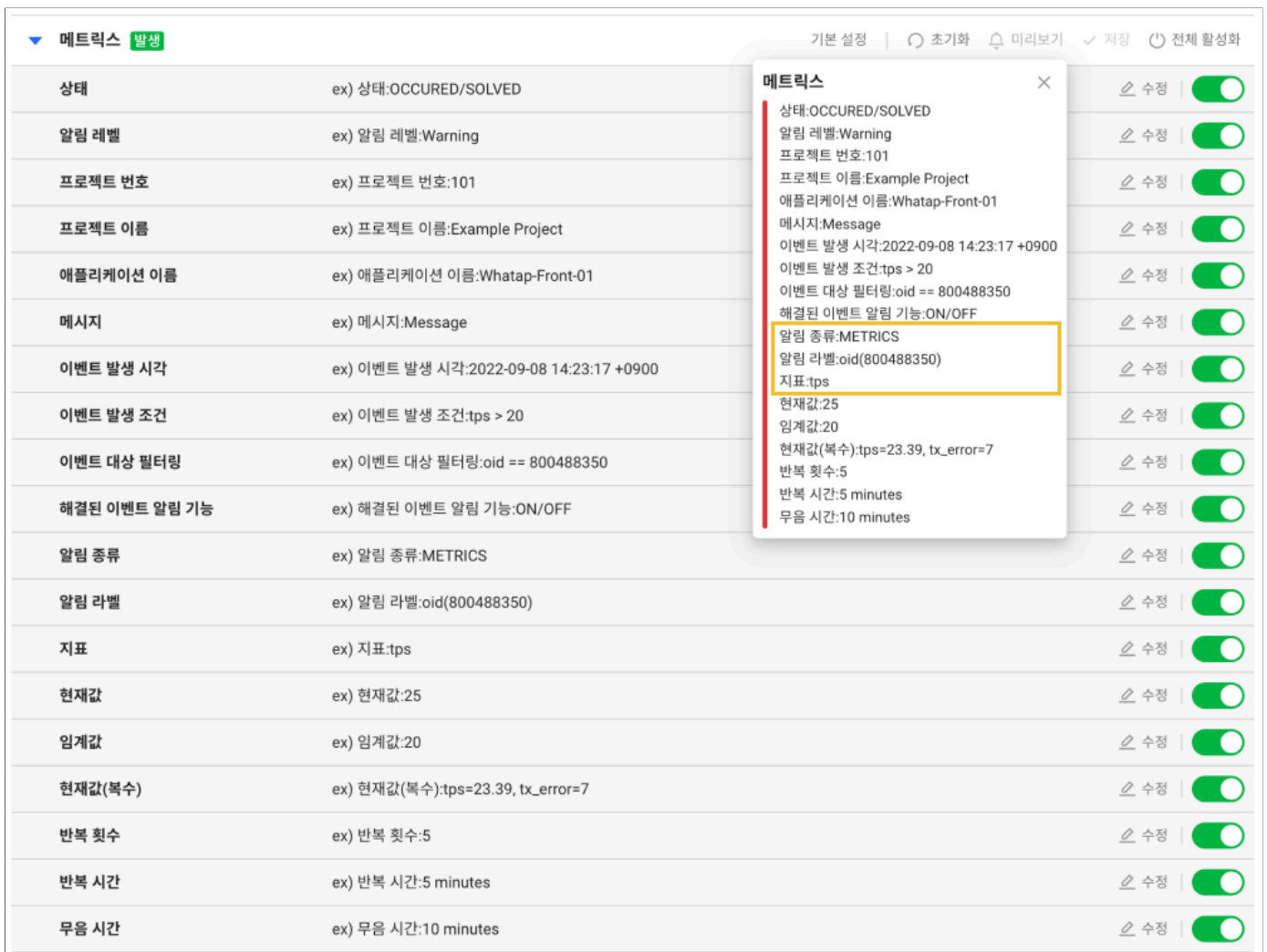
선택하면 수신할 메시지의 내용을 미리 확인할 수 있습니다.

- 수신한 메시지의 내용을 수정하려면 **4**  수정 버튼을 선택하세요.
- 수신한 메시지의 상세 항목을 감추려면 **5**  토글 버튼을 선택해 비활성화하세요.

수신 메시지 상세 항목 감추기

수신 메시지의 내용 중 불필요하다고 생각되는 항목을 감출 수 있습니다. 각 상세 항목의 가장 오른쪽의 토글 버튼을 선택해 해당 항목을 수신 메시지에서 감추거나 보이도록 설정할 수 있습니다.  **미리보기** 버튼을 선택해 수신할 메시지의 내용을 미리 확인할 수 있습니다.

- 상세 항목 활성화 예시



항목	예시	설정
상태	ex) 상태:OCCURED/SOLVED	수정 ON
알림 레벨	ex) 알림 레벨:Warning	수정 ON
프로젝트 번호	ex) 프로젝트 번호:101	수정 ON
프로젝트 이름	ex) 프로젝트 이름:Example Project	수정 ON
애플리케이션 이름	ex) 애플리케이션 이름:Whatap-Front-01	수정 ON
메시지	ex) 메시지:Message	수정 ON
이벤트 발생 시간	ex) 이벤트 발생 시간:2022-09-08 14:23:17 +0900	수정 ON
이벤트 발생 조건	ex) 이벤트 발생 조건:tps > 20	수정 ON
이벤트 대상 필터링	ex) 이벤트 대상 필터링:oid == 800488350	수정 ON
해결된 이벤트 알림 기능	ex) 해결된 이벤트 알림 기능:ON/OFF	수정 ON
알림 종류	ex) 알림 종류:METRICS	수정 ON
알림 라벨	ex) 알림 라벨:oid(800488350)	수정 ON
지표	ex) 지표:tps	수정 ON
현재값	ex) 현재값:25	수정 ON
임계값	ex) 임계값:20	수정 ON
현재값(복수)	ex) 현재값(복수):tps=23.39, tx_error=7	수정 ON
반복 횟수	ex) 반복 횟수:5	수정 ON
반복 시간	ex) 반복 시간:5 minutes	수정 ON
무음 시간	ex) 무음 시간:10 minutes	수정 ON

메트릭스 [X]

상태:OCCURED/SOLVED

알림 레벨:Warning

프로젝트 번호:101

프로젝트 이름:Example Project

애플리케이션 이름:Whatap-Front-01

메시지:Message

이벤트 발생 시간:2022-09-08 14:23:17 +0900

이벤트 발생 조건:tps > 20

이벤트 대상 필터링:oid == 800488350

해결된 이벤트 알림 기능:ON/OFF

알림 종류:METRICS

알림 라벨:oid(800488350)

지표:tps

현재값:25

임계값:20

현재값(복수):tps=23.39, tx_error=7

반복 횟수:5

반복 시간:5 minutes

무음 시간:10 minutes

• 상세 항목 비활성화 예시

The screenshot shows a configuration page for '메트릭스 발생' (Metrics Occurrence). At the top right, there are buttons for '기본 설정' (Basic Settings), '초기화' (Reset), '미리보기' (Preview), '저장' (Save), and '전체 활성화' (Enable All). The main table lists various configuration items with their current values and a '수정' (Edit) button. A modal window titled '메트릭스' is open, displaying a list of these items with their values and a '수정' (Edit) button. The toggle switches for '알림 종류' (Alert Type), '알림 라벨' (Alert Label), and '지표' (Indicator) are currently turned off (greyed out), while all other items have their toggle switches turned on (green).


항목	현재 값	수정	활성화
상태	ex) 상태:OCCURED/SOLVED	수정	ON
알림 레벨	ex) 알림 레벨:Warning	수정	ON
프로젝트 번호	ex) 프로젝트 번호:101	수정	ON
프로젝트 이름	ex) 프로젝트 이름:Example Project	수정	ON
애플리케이션 이름	ex) 애플리케이션 이름:Whatap-Front-01	수정	ON
메시지	ex) 메시지:Message	수정	ON
이벤트 발생 시각	ex) 이벤트 발생 시각:2022-09-08 14:23:17 +0900	수정	ON
이벤트 발생 조건	ex) 이벤트 발생 조건:tps > 20	수정	ON
이벤트 대상 필터링	ex) 이벤트 대상 필터링:oid == 800488350	수정	ON
해결된 이벤트 알림 기능	ex) 해결된 이벤트 알림 기능:ON/OFF	수정	ON
알림 종류	ex) 알림 종류:METRICS	수정	OFF
알림 라벨	ex) 알림 라벨:oid(800488350)	수정	OFF
지표	ex) 지표:tps	수정	OFF
현재값	ex) 현재값:25	수정	ON
임계값	ex) 임계값:20	수정	ON
현재값(복수)	ex) 현재값(복수):tps=23.39, tx_error=7	수정	ON
반복 횟수	ex) 반복 횟수:5	수정	ON
반복 시간	ex) 반복 시간:5 minutes	수정	ON
무음 시간	ex) 무음 시간:10 minutes	수정	ON

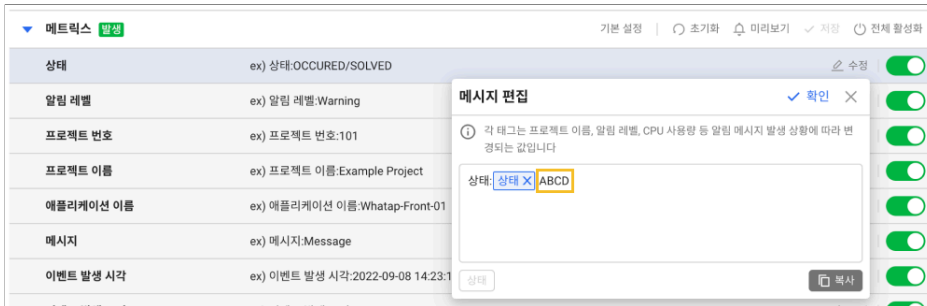
변경 사항을 반영하려면 ✓ **저장** 버튼을 선택하세요. 다른 카테고리의 변경 사항을 한번에 모두 반영하려면 화면 오른쪽 위에 **전체 저장** 버튼을 선택하세요.


- ① • 감춘 상세 항목을 수신 메시지에 다시 추가하려면 해당 항목의 토글 버튼을 선택해 활성화하세요.
- 모든 상세 항목을 수신 메시지에 추가하려면 () **전체 활성화** 버튼을 선택하세요.

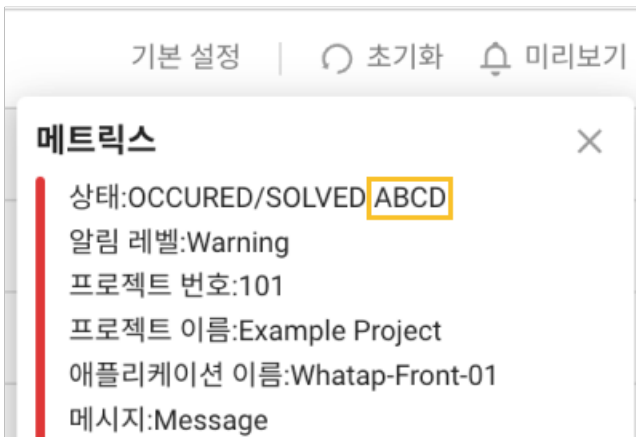
상세 항목 내용 수정하기


상세 항목의 내용에 사용자가 원하는 특별한 내용을 추가할 수 있습니다.

1. 각 상세 항목 가장 오른쪽에  수정 버튼을 선택하세요.
2. **메시지 편집** 창이 나타나면 추가할 내용을 입력하세요.

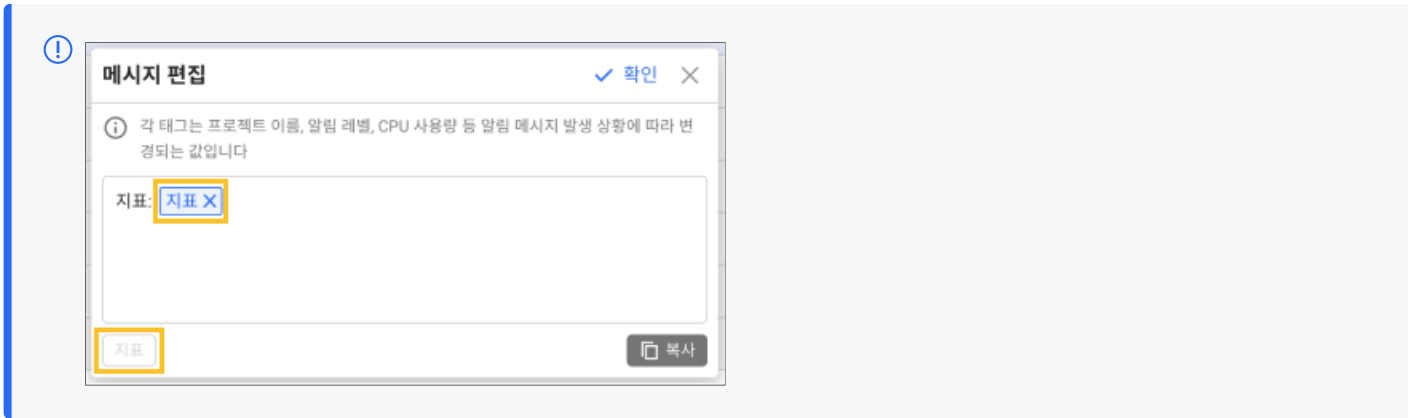


3. **확인** 버튼을 선택하세요.
4.  **미리보기** 버튼을 선택해 수신할 메시지의 내용을 미리 확인하세요.



변경 사항을 반영하려면  **저장** 버튼을 선택하세요. 다른 카테고리의 변경 사항을 한번에 모두 반영하려면 화면 오른쪽 위에 **전체 저장** 버튼을 선택하세요.

- ⓘ **메시지 편집** 창에서 텍스트 블록은 알림 발생 상황에 따라 변경되어 출력되는 변수입니다. 텍스트 블록을 드래그해서 위치를 변경할 수 있습니다. 텍스트 블록을 삭제했다면 창 아래에 텍스트 블록 버튼을 선택해 다시 추가할 수 있습니다.



- ⚠ 사용자에 의해 내용을 수정하기 전의 메시지는 사용자의 언어 상태에 따라 번역되어 전송되나, 내용 수정 이후에는 언어 상태와 관계 없이 사용자 정의된 포맷으로만 알림을 전송합니다.

다른 프로젝트에 설정 복사하기

사용자가 설정한 내용을 다른 프로젝트에 복사할 수 있습니다.

1. 화면 오른쪽 위에 [다른 프로젝트에 복사](#) 버튼을 선택하세요.
2. [프로젝트 적용](#) 창이 나타나면 설정 내용을 복사할 프로젝트를 선택하세요. 모든 프로젝트를 선택하려면 [전체 선택](#)을 선택하세요.
3. [적용](#) 버튼을 선택하세요.

- ⚠ 현재 프로젝트와 동일한 제품 타입만 선택할 수 있습니다.

초기화하기

사용자가 설정한 모든 변경 사항을 초기화하려면 [초기화](#) 버튼을 선택하세요. 모든 카테고리의 변경 사항을 초기화하려면 화면 오른쪽 위에 [전체 초기화](#) 버튼을 선택하세요. 확인 메시지 창이 나타나면 [확인](#) 버튼을 선택하세요.

- ⚠ 초기화된 설정은 복구할 수 없습니다.

AI 기반의 히트맵 패턴 경고 알림

홈 화면 > 프로젝트 선택 > 경고 알림 > 이벤트 설정 > 히트맵 패턴

애플리케이션 실행 분포 패턴을 머신 러닝을 통해 자동 인식하여 경고 알림을 보냅니다. 복잡한 조건 설정 없이도 이슈 상황을 빠르고 정확하게 인지할 수 있습니다. 수많은 히트맵 패턴들이 학습되어 있으며 실시간으로 수집되는 데이터에서 학습된 패턴과 유사한 패턴이 발생하면 경고 알림을 받아 볼 수 있습니다.

❗ 히트맵과 히트맵 패턴에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

기본 옵션

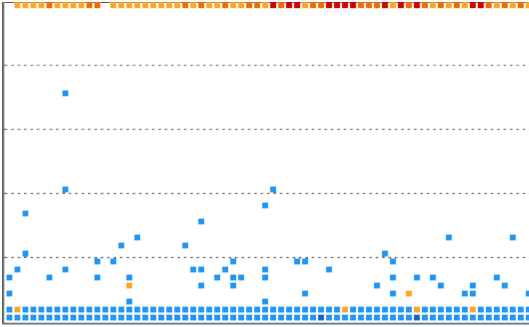
- **탐지 제외 시간**: 설정한 시간 미만의 패턴은 무시합니다.
- **무음**: 과도한 경고 알림 발생을 방지할 수 있는 옵션입니다. 첫번째 경고 알림 이후 선택한 시간 동안 경고 알림을 보내지 않습니다. 또한 **이벤트 기록** 메뉴에 기록되지 않습니다.

- **저장**: 설정한 내용을 저장하고 반영합니다.

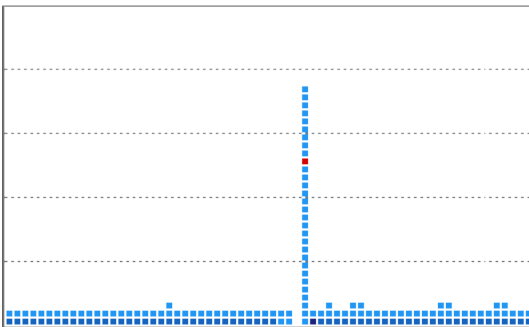
히트맵 패턴

다음의 히트맵 패턴이 발생하면 경고 알림을 보내도록 설정할 수 있습니다. 각 항목의 토글 버튼을 선택해 경고 알림을 켜거나 끌 수 있습니다.

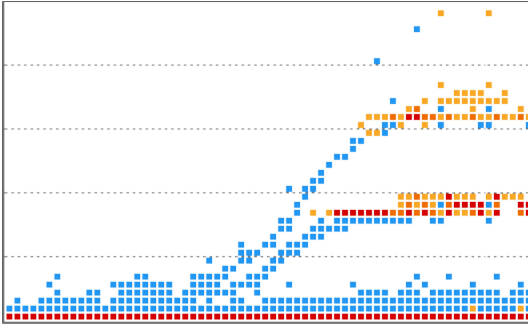
- **히트맵 가로라인 패턴**: 여러 트랜잭션이 일정한 시간 내에 종료되는 패턴입니다. 자원을 획득하거나 외부 HTTPS Call을 할 때 타임아웃 또는 지연이 발생합니다.



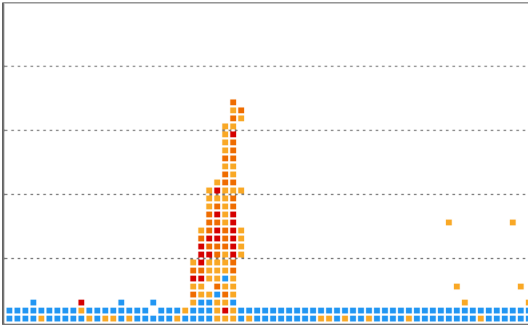
- **히트맵 세로라인 패턴**: 호출 시점이 다르지만 동일 시점에 트랜잭션이 종료되는 패턴입니다. 트랜잭션이 사용하는 공통의 자원에 일시적인 병목이 발생합니다.



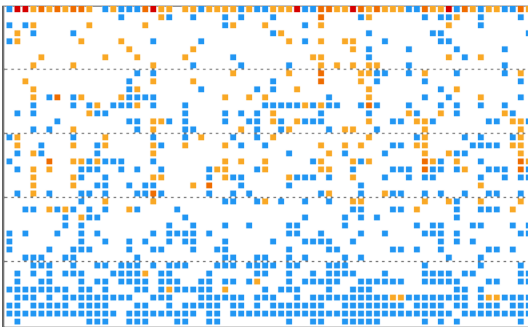
- **히트맵 플라잉 패턴**: 특정 리소스나 로그와 같은 공통 자원 부족현상으로 간격을 두고 파도 치는 것과 같은 현상의 패턴이 발생합니다.



- **히트맵 과부하 패턴:** 전체 또는 일부 응답에 일시적인 문제가 발생하면 일시적으로 트랜잭션이 밀집하는 패턴이 발생합니다.



- **히트맵 폭주 패턴:** 과도한 트랜잭션의 요청이나 부하가 발생하면 응답시간이 전체적으로 증가하는 패턴이 발생합니다.



이상치 탐지 경고 알림

홈 화면 > 프로젝트 선택 > 경고 알림 > 이벤트 설정 > 이상치 탐지 탭 선택

이상치 탐지(Anomaly Detection)란 데이터 안에서 anomaly, outlier, abnormal과 같이 예상치 못한 패턴을 찾는 일련의 활동입니다. 사용자는 매트릭스의 데이터를 기반으로 상승 패턴과 하락 패턴의 움직임이 평소와 같지 않을 경우 경고 알림을 보내도록 설정할 수 있습니다.

이벤트 설정

🏠
🔔
📄
💬
👤

애플리케이션
메트릭스
히트맵 패턴
이상치 탐지
로그

이상치 탐지
+ 이벤트 추가

	카테고리	필드	필터	상승/하락 패턴	
	app_counter(m5)	tx_error	없음	보통 / 보통	✎ 🗑️ 🏸
	java_memory(m5)	heap_used	없음	상승/하락 패턴 민감 / 민감	✎ 🗑️ 🏸
	app_host_resource(m5)	swap	없음	상승/하락 패턴 보통 / 보통	✎ 🗑️ 🏸
	app_counter	tps	없음	상승/하락 패턴 민감 / 민감	✎ 🗑️ 🟢
	app_counter(m5)	tx_count	<div style="display: flex; gap: 5px;"> <div style="border: 1px solid #ccc; padding: 2px;">demo-8100</div> <div style="border: 1px solid #ccc; padding: 2px;">demo-8101</div> </div> <div style="display: flex; gap: 5px; margin-top: 2px;"> <div style="border: 1px solid #ccc; padding: 2px;">demo-8102</div> <div style="border: 1px solid #ccc; padding: 2px;">demo-8103</div> </div>	비활성 / 보통	✎ 🗑️ 🏸

+ 이벤트 추가 버튼을 선택해 이상치 탐지 이벤트 설정을 진행할 수 있습니다. 모든 설정을 완료한 다음 저장 버튼을 선택하세요. 이상치 탐지 창의 제목 오른쪽에 토글 버튼을 선택해 활성화 여부를 선택할 수 있습니다.

이상치 탐지 이벤트 생성

이벤트 설정

애플리케이션 메트릭스 히트맵 패턴 **이상치 탐지** 로그

이상치 탐지

이벤트 이름	카테고리	필드
...	app_counter(m5)	tx_error
...	java_memory(m5)	heap_used
...	app_host_resource(m5)	swap
...	app_counter	tps
...	app_counter(m5)	tx_count

✕ 이상치 탐지
켜짐

▼ 이벤트 기본 정보

레벨 정보 경고 위험

* 메시지
알림 발생 시 알림 내용에 포함됩니다.

▼ 메트릭스

* 카테고리

* 필드

필터
이벤트 대상을 필터링합니다.

오브젝트 병합 없음 합산 평균

▼ 이상치 탐지

상승 패턴 둔감 보통 민감

하락 패턴 둔감 보통 민감

저장

이벤트 기본 정보

이상치 탐지 이벤트의 기본 정보를 입력하세요.

- **레벨**: 정보, 경고, 위험 수준 중 하나의 레벨을 선택하세요.
- **메시지**: 이벤트 발생 시 출력하는 알림 메시지를 입력합니다. `${태그 또는 필드키}` 입력으로 메시지에 변수를 적용할 수 있습니다. 변수에 입력할 키는 선택한 메트릭스 데이터 **카테고리**에 포함된 값이어야 합니다.

메트릭스

이벤트 발생 대상을 선택하세요.

- **카테고리**: 메트릭스 데이터를 구분하는 단위입니다. 메트릭스 이벤트 설정 시 필수 선택 값입니다.
- **필드**: 이벤트 발행 조건에 사용할 필드를 선택합니다. 다중 선택할 수 있습니다.
- **필터**: 이벤트 조건 대상을 선택합니다. 다중 선택할 수 있습니다.
- **오브젝트 병합**: 오브젝트 병합 방법을 선택할 수 있습니다.

이상치 탐지

상승 패턴 또는 하락 패턴의 민감도를 선택할 수 있습니다. 각 항목의 토글 버튼을 선택해 기능을 켜거나 끌 수 있습니다.

이상치 탐지 이벤트 수정

1. 경고 알림 > 이벤트 설정 메뉴에서 이상치 탐지 탭을 선택하세요.
2. 이상치 탐지 목록에서 수정하려는 항목의 오른쪽에 ✎ 버튼을 선택하세요.
3. 이상치 탐지 설정 창이 나타납니다. 수정이 필요한 항목을 수정한 다음 저장 버튼을 선택하세요.

이상치 탐지 이벤트 삭제

경고 알림 > 이벤트 설정 메뉴에서 이상치 탐지 탭을 선택하세요. 목록에서 삭제하려는 항목의 오른쪽에 🗑 버튼을 선택하세요. 확인 팝업 메시지가 나타나면 삭제 버튼을 선택하세요.

로그 경고 알림

- ❗ 이 기능은 로그 모니터링을 활성화해야 이용할 수 있습니다. [로그 모니터링 활성화](#)에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.
- 로그 모니터링의 주요 메뉴에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

홈 화면 > 프로젝트 선택 > 경고 알림 > 이벤트 설정 > 로그 탭

수집한 로그 데이터를 조건에 맞춰 필터링해 경고 알림을 설정할 수 있습니다. + 이벤트 추가 버튼을 선택해 로그 이벤트 경고 알림을 설정하세요. 모든 설정을 완료한 다음 저장 버튼을 선택하세요.

이벤트 설정

실시간 로그 이벤트
JSON
+ 이벤트 추가

이벤트 이름	카테고리	검색 키	검색 값	이벤트 발생 일시 중지	이벤트 대상 필터링	이벤트 수신 태그	설정
실시간 로그 이벤트	AppLog	onodeName	node-0	20분	onode == '334634079' && okindName != 'demo-okind-1'	test-tag	🔍 <input type="checkbox"/>
이벤트 테스트	AppLog	oid	-1128904592	사용 안 함	사용 안 함	전체 멤버 수신	🔍 <input checked="" type="checkbox"/>
test	AppLog	oname	demo-8100	사용 안 함	oname && 'demo-8100'	전체 멤버 수신	🔍 <input checked="" type="checkbox"/>

복합 로그 이벤트
JSON
+ 이벤트 추가

이벤트 이름	템플릿	카테고리	규칙	이벤트 상태가 해결되면 추가 알림	이벤트 발생 일시 중지	인터벌	이벤트 수신 태그	설정
이벤트 이름_테스트	2xx 상태코드 건수 <code>count</code>	AppLog	include_minus_exclude_count > 10	Off	1분	5분	전체 멤버 수신	🔍 <input checked="" type="checkbox"/>
retreter	사용 안 함	#WhatapEvent	rows > 10	Off	1분	5분	전체 멤버 수신	🔍 <input checked="" type="checkbox"/>
test	사용 안 함	*	rows > 10	Off	1분	5분	전체 멤버 수신	🔍 <input checked="" type="checkbox"/>

추가할 수 있는 로그 이벤트 다음과 같습니다.

- 실시간 로그 이벤트** : 실시간으로 수집한 로그에서 검색 값이 등장하면 경고 알림을 보냅니다.
- 복합 로그 이벤트** : 최근에 수집한 로그 중 일정 조건을 만족하는 로그가 일정 개수 이상 수집한 경우에 경고 알림을 보냅니다.

- ① 이 이벤트를 추가하거나 설정하려면 **알림 설정** 권한이 있어야 합니다. 사용자별 권한에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.
- 경고 알림과 관련해 모니터링 플랫폼별 지원되는 이벤트 종류를 확인하려면 [다음 문서](#)를 참조하세요.

이벤트 추가 공통 옵션

다음은 이벤트 추가 시 공통으로 설정할 수 있는 옵션입니다.

- 이벤트 이름** : 추가하려는 이벤트 이름을 입력하세요.
- 이벤트 활성화** : 토글 버튼을 클릭해 경고 알림 활성화 여부를 선택할 수 있습니다.
- 레벨** : **위험**, **경고**, 정보 중 하나의 레벨을 선택하세요.
- 메시지** : 이벤트 발생 시 출력하는 알림 메시지를 입력합니다. `#{태그 또는 필드키}` 입력으로 메시지에 변수를 적용할 수 있습니다. 변수에 입력할 키는 선택한 매트릭스 데이터 **카테고리**에 포함된 값이어야 합니다.
- 카테고리** : 로그 구분 명칭(로그 폴더명)을 목록에서 선택하거나 직접 입력할 수 있습니다.
- 이벤트 발생 일시 중지** : 과도한 경고 알림 발생을 방지할 수 있는 옵션입니다. 첫 번째 경고 알림 이후 선택한 시간 동안 경고 알림을 보내지 않습니다. 또한 **이벤트 기록** 메뉴에 기록되지 않습니다.
- 이벤트 수신 태그** : 이벤트 수신 태그를 선택하면 해당 태그를 가진 프로젝트 멤버와 3rd-party 플러그인에 알림을 전송할 수 있습니다. 이벤트 수신 태그를 선택하지 않으면 프로젝트 전체 멤버에게 경고 알림을 보냅니다.

태그를 추가하지 않으면 전체 멤버에게 경고 알림을 보냅니다. **+ 태그 추가**를 클릭한 다음 **태그 목록**에서 경고 알림 수신 대상을 선택하세요. **+ 새 태그 생성**을 선택해 태그를 추가할 수도 있습니다.

- ① **경고 알림 > 이벤트 수신 설정** 메뉴에서 프로젝트 멤버와 3rd-party 플러그인에 태그를 설정할 수 있습니다. **이벤트 수신 설정** 메뉴에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

실시간 로그 이벤트 추가

이벤트 추가

이벤트 이름 * 제목

이벤트 활성화

레벨 * 위험 경고 정보

메시지 * 메시지

카테고리 ⓘ * 카테고리(를) 선택해주세요

검색 키 * 카테고리(를) 먼저 선택해 주세요.

검색 값 * 카테고리(를) 먼저 선택해 주세요.

입력된 단어가 일치하는 경우 알림이 발생합니다.

이벤트 대상 필터링 ⓘ

선택 입력 직접 입력

검색 키 ▼ 연산자 ▼ 검색 값 ▼

+ 추가

입력값이 없을 경우, 실시간으로 수정

ex. oid == '12345678' && level == 'ERROR'

이벤트 발생 일시 중지 * 사용 안 함 ▼

알림 수신 후 선택한 시간 동안 이벤트가 발생하지 않습니다.

단, "이벤트 상태가 해결되면 추가 알림" 기능을 활성화한 경우에는 RECOVERED 알림 수신 후 선택한 시간 동안 이벤트가 발생하지 않습니다.

이벤트 수신 태그 ⓘ 전체 멤버 수신 + 태그 추가

[프로젝트 이벤트 수신설정 메뉴 바로가기](#)

저장

- **검색 키** : 로그 데이터 내에서 특정 값에 접근하기 위한 식별자를 의미합니다. 목록에서 선택하거나 직접 입력할 수 있습니다.

예시, HTTP 응답 상태 코드를 나타내는 값에 접근하고자 할 경우 **검색 키** `status`

- **검색 값** : **검색 키**에 해당하는 실제 데이터를 의미합니다. 로그에서 입력한 단어를 포함할 경우 경고 알림을 보냅니다. 목록에서 선택하거나 직접 입력할 수 있습니다.

예시, **검색 키** `status` **검색 값** `200`을 설정한 경우 HTTP 응답 상태 코드 200을 포함하는 로그 데이터 수집 시 경고 알림 발생

- **이벤트 대상 필터링** : **선택 입력** 옵션을 통해 **검색 키**와 **연산자**, **검색 값**을 선택해 대상을 필터링하거나 **직접 입력** 옵션을 선택할 수 있습니다. 입력값이 없을 경우 실시간으로 수집하는 데이터 전체에 대한 알림 발생 여부를 판단합니다.

예시, `AppLog` 카테고리의 로그 중 `level` 이 `ERROR`인 로그를 찾습니다. 일치하는 로그 중에서 `oid`가 `12345678`인 경우 경고 알림을 보냅니다.

복합 로그 이벤트 추가

×
이벤트 추가

***이벤트 이름**

이벤트 활성화

레벨 위험 경고 정보 이벤트 상태가 해결되면 추가 알림 ⓘ

사용 안 함

- 2xx 상태코드 건수 count
- 3xx 상태코드 건수 count
- 4xx 상태코드 건수 count
- 5xx 상태코드 건수 count
- 정상 상태코드(2xx,3xx) 건수 count
- 에러 상태코드(4xx,5xx) 건수 count
- 에러 수신 건수 count

템플릿 사용 안 함

***메시지**

***카테고리 ⓘ**

데이터 조회 범위 최근에 1 분 ▼
선택 시간동안 수집된 로그를 조회합니다.

로그 검색 조건 ⓘ 선택 입력 직접 입력

검색 키 ▼ 검색 값 ▼ 제외 —

+ 추가
이벤트 발행 조건을 입력하기 위해서 카테고리를 먼저 선택해 주세요.

***이벤트 발행 조건** 조건에 맞는 로그 > ▼ 10

인터벌 ⓘ 5 분 ▼

무음 ⓘ 1 분 ▼

이벤트 수신 태그 ⓘ 전체 멤버 수신 + 태그 추가

🔗 프로젝트 이벤트 수신설정 메뉴 바로가기

저장

- **템플릿** : 복합 로그 템플릿을 제공합니다.
- **로그 검색 조건**
 - **검색 키**에서 이벤트 발생 조건 대상을 선택할 수 있습니다. 선택한 **검색 키**에 해당하는 검색 값을 선택할 수 있습니다.
 - **검색 키**에서 동일한 항목을 추가할 경우 'OR' 조건으로, 다른 항목을 추가할 경우 'AND' 조건으로 동작합니다.
 - **제외** 체크 박스를 선택해 선택한 검색 값을 이벤트 발행 조건에서 제외할 수 있습니다.
 - **+ 추가**를 선택해 여러개의 이벤트 발행 조건을 추가 또는 제외 설정할 수 있습니다.
- **데이터 조회 범위** : 선택한 시간동안 수집한 로그를 조회합니다. 데이터 조회 시간을 5분으로 선택하면 최근 5분 동안 수집한 데이터를 조회해서 이벤트 발생 조건을 확인합니다.
- **이벤트 발행 조건** : 이벤트가 입력한 횟수와 선택한 연산자의 조건과 같이 발생할 때 경고 알림을 보냅니다.

예시, AppLog 카테고리의 로그 중 조건 입력에 해당하는 로그를 필터링 합니다. 조건 입력에서 제외를 체크한 경우 해당 조건으로 찾은 로그를 제외하겠다는 의미입니다. 따라서 level 이 ERROR 인 로그는 제외합니다. 최근 10분 동안 수집한 로그 중 이벤트가 5 보다 작을 경우 경고 알림을 보냅니다.

로그 이벤트 설정 수정하기

1. 경고 알림 > 이벤트 설정 메뉴로 이동하세요.
2. 로그 탭을 선택하세요.
3. 로그 이벤트 목록 중 수정하려는 이벤트 항목에서 오른쪽에 ✎ 버튼을 선택하세요.
4. 이벤트 설정 창이 나타나면 옵션을 수정한 다음 저장 버튼을 선택하세요.

선택한 로그 이벤트를 삭제하려면 이벤트 설정 창에서 오른쪽 위에 삭제 버튼을 선택하세요.




로그 이벤트 끄기

1. 경고 알림 > 이벤트 설정 메뉴로 이동하세요.
2. 로그 탭을 선택하세요.
3. 로그 이벤트 목록 중 경고 알림을 끄려는 이벤트 항목의 가장 오른쪽에 토글 버튼을 선택하세요.

다시 토글 버튼을 선택하면 해당 경고 알림이 활성화됩니다.

로그 이벤트 내보내기/불러오기

로그 이벤트의 설정 내용을 json 파일로 저장한 다음 불러와 재사용할 수 있습니다.

1. 경고 알림 > 이벤트 설정 메뉴로 이동하세요.
2. 로그 탭을 선택하세요.
3. 로그 이벤트 목록 위에 JSON  버튼을 선택하세요. JSON 내보내기 창이 나타납니다.
4. 내보내기 할 항목을 수정 또는 편집하세요.
5. 오른쪽 위에  내보내기 버튼을 선택하세요. 브라우저에서 json 파일을 다운로드합니다.
6. 로그 이벤트 목록 위에  버튼을 선택하세요.
7. 파일 선택 창이 나타나면 앞서 다운로드 받은 json 파일을 선택하세요.
8. JSON 가져오기 창이 나타나면 내용을 수정한 다음 + 목록에 추가하기 버튼을 선택하세요.

❗ 이벤트에 id가 존재합니다. id를 제거한 뒤 다시 시도하세요.

- 메시지가 나타나면 JSON 가져오기 창에서 id 항목을 삭제한 다음 + 목록에 추가하기 버튼을 선택하세요.
- 기존의 이벤트 항목에 덮어쓰기를 하려면 id 항목을 삭제한 다음 덮어쓰기 버튼을 선택하세요.

경고 알림 수신 설정

홈 화면 > 프로젝트 선택 > 경고 알림 > 이벤트 수신 설정

프로젝트 멤버들의 경고 알림 수신과 관련한 다양한 기능을 설정할 수 있습니다.

이벤트 수신 설정



▼ 사용자별 이벤트 수신 설정 (64)

이벤트 알림의 일괄 수신설정 및 접근 설정을 위한 모바일 기기 관리는 [계정 정보 메뉴](#)에서 가능합니다. [계정 정보 >](#)

이름	이메일 알림	SMS 알림	WhatsApp 알림	모바일 알림	반복 알림 (에스컬레이션)	이벤트 수신 태그
JH	<input checked="" type="checkbox"/> 수신 레벨: 위임	<input type="checkbox"/> 수신 레벨: 위임	등록된 번호가 없습니다.	<input type="checkbox"/> 모바일 기기 6대 수신 레벨: 전체	0 저장	<input type="checkbox"/> 수신 태그 미설정 알림 받기 testtag 테스트 태그 +
JT	<input checked="" type="checkbox"/> 수신 레벨: 전체	<input type="checkbox"/> 수신 레벨: 전체	등록된 번호가 없습니다.	등록된 기기가 없습니다.	0 저장	<input checked="" type="checkbox"/> 수신 태그 미설정 알림 받기 + 태그 추가
JU	<input checked="" type="checkbox"/> 수신 레벨: 전체	등록된 번호가 없습니다.	등록된 번호가 없습니다.	등록된 기기가 없습니다.	0 저장	<input checked="" type="checkbox"/> 수신 태그 미설정 알림 받기 + 태그 추가
KJ	<input type="checkbox"/> 수신 레벨: 전체	<input type="checkbox"/> 수신 레벨: 전체	등록된 번호가 없습니다.	등록된 기기가 없습니다.	0 저장	<input checked="" type="checkbox"/> 수신 태그 미설정 알림 받기 + 태그 추가
KY	<input checked="" type="checkbox"/> 수신 레벨: 전체	<input type="checkbox"/> 수신 레벨: 전체	등록된 번호가 없습니다.	등록된 기기가 없습니다.	0 저장	<input checked="" type="checkbox"/> 수신 태그 미설정 알림 받기 + 태그 추가
	<input checked="" type="checkbox"/> 수신 레벨: 전체	<input type="checkbox"/> 수신 레벨: 전체	등록된 번호가 없습니다.	등록된 기기가 없습니다.	0 저장	<input checked="" type="checkbox"/> 수신 태그 미설정 알림 받기

3rd 파티 플러그인

Slack, Telegram, Teams, Jandi, Webhook 등을 이용하여 알림 메시지를 받으실 수 있습니다.

플러그인 이름	인증 키	인증 값	반복 알림 (에스컬레이션)	이벤트 수신 태그	
SLACK			0 저장	<input checked="" type="checkbox"/> 수신 태그 미설정 알림 받기 + 태그 추가	
SLACK			0 저장	<input checked="" type="checkbox"/> 수신 태그 미설정 알림 받기 + 태그 추가	
SLACK			0 저장	<input checked="" type="checkbox"/> 수신 태그 미설정 알림 받기 + 태그 추가	
SLACK			0 저장	<input checked="" type="checkbox"/> 수신 태그 미설정 알림 받기 + 태그 추가	
TELEGRAM			0 저장	<input checked="" type="checkbox"/> 수신 태그 미설정 알림 받기 PREV_TEST +	
WEBHOOK_JSON			0 저장	<input checked="" type="checkbox"/> 수신 태그 미설정 알림 받기 + 태그 추가	

[+ 앱 추가하기](#)

대량 알림 발생 방지

알림이 대량으로 발생하면 지정된 시간 동안 알림이 일시적으로 중지됩니다.
대량 알림 차단 기능을 해제하려면 (이메일 주소 옆의) '중단 해제' 버튼을 눌러주세요.

활성화

탐지 시간: 5분

탐지 횟수: 10

정지 시간: 3시간

[저장](#)

수신 수단 선택하기

이메일 알림 이외에 SMS, 모바일 알림을 선택할 수 있습니다. 원하는 알림 수신 수단의 체크 박스를 체크하면 경고 알림을 받을 수 있습니다. 알림 수신 수단의 체크 박스를 해제하면 경고 알림을 보내지 않습니다.

- ① • 이메일 알림은 회원 가입 시 입력한 이메일 주소로 알림을 보냅니다.
- 프로젝트 최고 관리자를 제외한 모든 사용자는 자신의 수신 설정만 변경할 수 있습니다.

SMS 알림 수신 추가하기

SMS 알림 수신이 필요한 경우 [계정 관리](#)에서 사용자 전화번호를 설정하세요.

사용자 전화번호

전화번호는 경고 알림 문자에 사용됩니다. 전화번호를 변경하려면 SMS 인증이 필요합니다.

전화번호
 인증 완료

일반 휴대전화는 **한국 휴대전화 번호만 등록** 가능합니다.

1. 화면 오른쪽 위에 프로필 아이콘을 선택하세요.
2. 팝업 메뉴가 나타나면 [계정 관리](#) 버튼을 선택하세요.
3. [사용자 전화번호](#) 섹션에서 [일반 휴대전화](#) 버튼을 선택하세요.
4. [전화번호](#)에 인증번호를 수신할 전화번호를 입력하세요.
5. [인증번호 전송](#) 버튼을 선택하세요.
6. 사용자의 휴대전화로 전송된 인증 번호를 입력하세요.
7. [인증하기](#) 버튼을 선택하세요.

- ① • 등록된 전화번호를 변경하려면 [번호 변경하기](#) 버튼을 선택한 다음 위의 과정을 반복하세요.
- SMS를 알림으로 수신할 수 있는 전화번호는 **한국 휴대전화 번호만** 등록할 수 있습니다.

WhatsApp 알림 수신 추가하기

WhatsApp을 통해 알림을 수신할 수 있습니다.

사용자 전화번호

전화번호는 경고 알림 문자에 사용됩니다. 전화번호를 변경하려면 SMS 인증이 필요합니다.

일반 휴대전화 **WhatsApp**

전화번호

1. 화면 오른쪽 위에 프로필 아이콘을 선택하세요.
2. 팝업 메뉴가 나타나면 **계정 관리** 버튼을 선택하세요.
3. **사용자 전화번호** 섹션에서 **WhatsApp** 버튼을 선택하세요.
4. **전화번호**에 인증번호를 수신할 전화번호를 입력하세요.
5. **인증번호 전송** 버튼을 선택하세요.
6. WhatsApp 애플리케이션으로 전송된 인증번호 6자리를 입력하세요.
7. **인증하기** 버튼을 선택하세요.

ⓘ 등록된 전화번호를 변경하려면 **번호 변경하기** 버튼을 선택한 다음 위의 과정을 반복하세요.

수신 레벨 선택하기

경고 알림 레벨에 따라 알림 수신 여부를 선택할 수 있습니다. **사용자별 이벤트 수신 설정** 섹션의 사용자 목록에서 **수신 레벨**을 **전체** 또는 **위험**을 선택하세요.

- **전체**: 모든 경고 알림을 수신할 수 있습니다.
- **위험**: 위험 레벨의 경고 알림만 수신할 수 있습니다.

요일 및 시간별 알람 설정하기

요일별, 시간별 알림 수신 여부를 선택할 수 있습니다. **사용자별 이벤트 수신 설정** 섹션의 사용자 목록에서 가장 왼쪽에 **▼** 버튼을 선택하세요. 경고 알림 수신을 원하는 요일을 선택하거나 시간을 입력하세요. 알림 수신 수단별로 설정할 수 있습니다.

▼ 사용자별 이벤트 수신 설정 (64)

이벤트 알림의 일괄 수신설정 및 접근 설정을 위한 모바일 기기 관리는 [계정 정보 메뉴에서](#) 가능합니다. [계정 정보 >](#)

이름	이메일 알림	SMS 알림	WhatsApp 알림	모바일 알림	반복 알림 (에스컬레이션)	이벤트 수신 태그
HS	<input checked="" type="checkbox"/> 수신 레벨: 전체	<input type="checkbox"/> 0***** 등록된 번호가 없습니다.	<input type="checkbox"/>	<input type="checkbox"/> 모바일 기기 2대 수신 레벨: 전체	0 저장	<input checked="" type="checkbox"/> 수신 태그 미설정 알림 받기 + 태그 추가
알림 수신 언어	요일 <input checked="" type="checkbox"/> 월 <input checked="" type="checkbox"/> 화 <input checked="" type="checkbox"/> 수 <input checked="" type="checkbox"/> 목 <input checked="" type="checkbox"/> 금 <input checked="" type="checkbox"/> 토 <input checked="" type="checkbox"/> 일 시간 00:00 ~ 00:00	요일 <input checked="" type="checkbox"/> 월 <input checked="" type="checkbox"/> 화 <input checked="" type="checkbox"/> 수 <input checked="" type="checkbox"/> 목 <input checked="" type="checkbox"/> 금 <input checked="" type="checkbox"/> 토 <input checked="" type="checkbox"/> 일 시간 00:00 ~ 00:00	요일 <input checked="" type="checkbox"/> 월 <input checked="" type="checkbox"/> 화 <input checked="" type="checkbox"/> 수 <input checked="" type="checkbox"/> 목 <input checked="" type="checkbox"/> 금 <input checked="" type="checkbox"/> 토 <input checked="" type="checkbox"/> 일 시간 00:00 ~ 00:00	모바일 기기 테스트	테스트	

경고 알림 반복 설정하기

경고 알림 발생 시간으로부터 알림 발생 상황이 해소되지 않을 경우 최초 알림 발생 시각으로부터의 알림 반복 간격을 설정할 수 있습니다. 예를 들어, 경고 알림 발생 시간으로부터 0분(즉시), 1시간 후, 1일 후에 경고 알림을 반복하려면 '0,1H,1D'를 [반복 알림 \(에스컬레이션\)](#) 컬럼 항목에 입력하세요.

반복 알림 (에스컬레이션) ⓘ

0,1H,1D 저장

- ⓘ • 이 기능은 **Critical** (또는 **위험**) 등급의 모든 이벤트를 대상으로 합니다. 이벤트 추가 시 설정한 **레벨** 항목을 참조하세요.
- **M**: 분, **H**: 시간, **D**: 일, 단위를 생략하면 분 단위로 시간을 설정합니다.
- **저장** 버튼을 선택하지 않으면 설정을 완료할 수 없습니다.
- 숫자 또는 숫자+단위(**M**, **H**, **D**)로 입력하세요. 입력이 올바르지 않으면 메시지가 표시됩니다.

팀별, 사용자별 알림 설정하기

프로젝트의 멤버 중 특정 멤버 또는 팀을 대상으로 알림 수신 여부를 설정합니다. [메트릭스](#) 및 [이상치 탐지](#), [로그](#) 이벤트 설정의 [이벤트 수신 태그](#) 항목에서 태그를 추가하세요. 이벤트별로 경고 알림을 수신하는 멤버 또는 팀을 선택할 수 있습니다.

반대로 이벤트 수신 태그를 설정하지 않으면 전체 멤버에게 경고 알림을 보낼 수 있습니다.

> 이벤트 수신 태그 사용 예시

이벤트 수신 태그에 대한 사용 예시를 통해 팀별 또는 사용자별로 경고 알림을 전송하는 설정 방법에 대해 알아봅니다. 다음과 같이 팀 별로 서로 다른 경고 알림을 전송하도록 설정합니다.

알림 종류	서버팀 수신 여부	데브옵스팀 수신 여부
메트릭스 경고 알림	O	X
이상치 탐지 경고 알림	O	O
로그 실시간 경고 알림	X	O

1. 이벤트 수신 설정하기 (경고 알림 > 이벤트 수신 설정)

▼ 사용자별 이벤트 수신 설정 (64)

이벤트 알림의 일괄 수신설정 및 접근 설정을 위한 모바일 기기 관리는 [계정 정보 메뉴에서](#) 가능합니다. [계정 정보 >](#)

이름	이메일 알림	SMS 알림	WhatsApp 알림	모바일 알림	반복 알림 (에스컬레이션)	이벤트 수신 태그
DE	<input checked="" type="checkbox"/> 수신 레벨: 전체	등록된 번호가 없습니다.	등록된 번호가 없습니다.	등록된 기기가 없습니다.	0 저장	<input checked="" type="checkbox"/> 수신 태그 미설정 알림 받기 Server +
DE	<input checked="" type="checkbox"/> 수신 레벨: 전체	등록된 번호가 없습니다.	등록된 번호가 없습니다.	등록된 기기가 없습니다.	0 저장	<input checked="" type="checkbox"/> 수신 태그 미설정 알림 받기 DevOps +

프로젝트에 속한 멤버들 중 서버팀 소속은 [서버팀](#), 데브옵스팀 소속은 [데브옵스팀](#) 으로 이벤트 수신 태그를 설정하세요.

2. 메트릭스 경고 알림 설정하기 (경고 알림 > 이벤트 설정 > 메트릭스 > 이벤트 추가 > 이벤트 수신 설정)

이벤트 수신 태그

Server +

이벤트 설정 시 이벤트 수신 태그를 선택하여 해당 태그를 가진 프로젝트 멤버와 3rd-party 플러그인에 알림을 전송할 수 있습니다.
이벤트 수신 설정 메뉴에서 프로젝트 멤버와 3rd-party 플러그인에 각각 태그를 지정할 수 있습니다

[프로젝트 이벤트 수신설정 메뉴](#)

이벤트 설정 시 태그를 선택하지 않은 경우 프로젝트 이벤트 수신 설정 메뉴의 나머지 수신 조건(활성화 여부 등)에 따라 알림이 발생합니다.

[태그 추가](#) 또는 + 버튼을 클릭하세요. 태그 목록에서 원하는 태그를 선택하거나 새 태그를 생성하세요. 메트릭스 이벤트에 대한 알림을 `서버팀`으로 설정한 경우입니다.

3. 로그 실시간 경고 알림 설정 (경고 알림 > 이벤트 설정 > 로그 > 이벤트 추가 > 이벤트 수신 태그)

이벤트 수신 태그 ⓘ

DevOps +

[프로젝트 이벤트 수신설정 메뉴 바로가기](#)

[태그 추가](#) 또는 + 버튼을 클릭하세요. 태그 목록에서 원하는 태그를 선택하거나 새 태그를 생성하세요. 로그 실시간 이벤트에 대한 알림을 `데브옵스팀`으로 설정한 경우입니다.

4. 이상치 탐지 이벤트는 전체 멤버에게 경고 알림을 전송하므로 이벤트 수신 태그를 설정하지 않습니다.

이벤트 수신 태그 추가하기

1. 사용자별 이벤트 수신 설정 섹션의 사용자 목록에서 [태그 추가](#) 또는 + 버튼을 선택하세요.
2. 이벤트 수신 태그 팝업 창이 나타나면 태그 입력란에 태그 이름을 입력한 다음 엔터를 입력하거나 [새 태그 생성](#)을 선택하세요.
3. 태그 색상을 선택하세요.
4. [태그 생성](#) 버튼을 선택하세요.

[태그 목록](#)에서 생성한 태그를 확인할 수 있습니다. 생성한 태그를 적용하려면 해당 태그를 선택하세요.

이벤트 수신 태그 해제하기

1. 사용자별 이벤트 수신 설정 섹션의 사용자 목록에서 + 버튼을 선택하세요.
2. 이벤트 수신 태그 팝업 창이 나타나면 적용된 태그 옆에 × 버튼을 선택하세요.
3. 이벤트 수신 태그 팝업 창을 닫으세요.

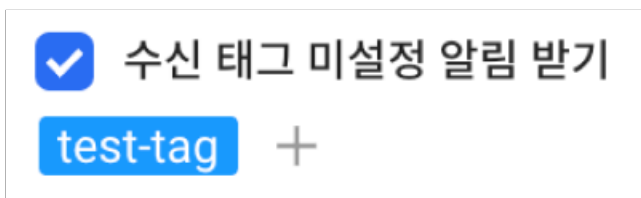
이벤트 수신 태그를 해제합니다.

이벤트 수신 태그 수정 및 삭제하기

1. 사용자별 이벤트 수신 설정 섹션의 사용자 목록에서 **태그 추가** 또는 **+** 버튼을 선택하세요.
2. 이벤트 수신 태그 팝업 창이 나타나면 **태그 목록**에서 수정 또는 삭제할 항목의 **✎** 버튼을 선택하세요.
3. 태그 이름을 수정하거나 색상을 변경한 다음 **적용** 버튼을 선택하세요.
태그를 삭제하려면 **🗑️ 태그 삭제** 버튼을 선택하세요.

❗ 이벤트에 적용 중인 이벤트 수신 태그 항목은 삭제할 수 없습니다.

수신 태그 미설정 알림



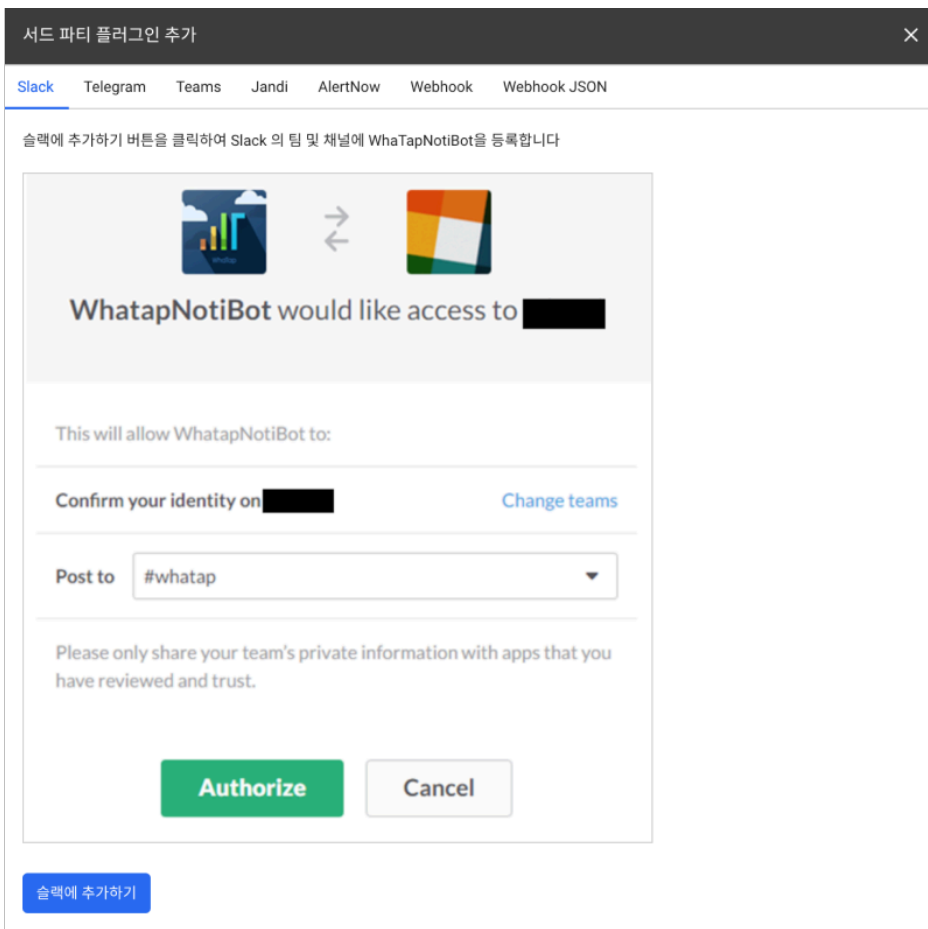
이벤트 수신 태그가 설정되지 않은 경고 알림을 받으려면 **수신 태그 미설정 알림 받기** 옵션을 선택하세요. 이벤트 수신 태그가 설정된 경고 알림만 받고 싶다면 선택을 해제하세요.

❗ 모든 경고 알림을 받지 않으려면 해당 옵션을 해제하고 선택한 이벤트 수신 태그가 없어야 합니다.

3rd 파티 플러그인 알림 추가하기

Slack, Telegram, Teams, Jandi, Webhook 등의 외부 애플리케이션을 통해 경고 알림을 받을 수 있습니다.

1. **경고 알림 > 이벤트 수신 설정** 메뉴에서 **3rd 파티 플러그인** 섹션의 **추가하기** 버튼을 선택하세요.
2. 원하는 서비스를 선택하세요.



3. 선택한 서비스의 화면 안내에 따라 설정을 진행하세요.
4. 모든 과정을 완료했다면 추가 버튼을 선택하세요.

ⓘ 와탭랩스의 지원 범위에 포함하지 않는 사내 메신저는 표준 Webhook, webhook json을 통해 연동할 수 있습니다.

대량 알림 발생 방지

알림이 대량으로 발생하면 설정한 시간 동안 경고 알림을 일시적으로 중단합니다. [경고 알림](#) > [이벤트 수신 설정](#) 메뉴에서 [대량 알림 발생 방지](#) 섹션으로 이동하세요.

대량 알림 발생 방지

알림이 대량으로 발생하면 지정한 시간 동안 알림이 일시적으로 중지됩니다.
대량 알림 차단 기능을 해제하려면 (이메일 주소 옆의) '중단 해제' 버튼을 눌러주세요.

활성화



탐지 시간

5분



탐지 횟수

10

정지 시간

3시간



저장

- **활성화** 토글 버튼을 선택해 기능을 켜거나 끌 수 있습니다.
- **탐지 시간** 동안 **탐지 횟수** 이상의 이벤트가 발생하면 **정지 시간** 동안 경고 알림을 중지합니다.

예를 들어, 5분 동안 20회의 이벤트가 발생하면 5분 동안 경고 알림을 중지합니다. 설정한 **정지 시간** 시간이 지나면 대량 알림 발생 방지 기능은 해제됩니다.

! 문자 메시지 알림이 하루 200건 이상 발생하면 일시 중지하며 다음 메시지를 표시합니다. 문자 알림 중단 기능을 해제하려면 **문자알림 중단 해제** 버튼을 선택하세요.

다량의 문자메시지가 전송되었습니다. (200건 / 일)

- !** 프로젝트에서 발생한 많은 양의 문자 메시지로 인해 문자 메시지 알림이 중지됩니다.
- 시작 시간 : 2024-02-02 18:40:01

문자알림 중단 해제

경고 알림 사용자 설정하기

계정 관리 메뉴에서 사용자 개인의 알림 수신 레벨, 수신 수단, 요일 및 시간 등을 설정할 수 있습니다.

1. 화면의 오른쪽 위에 프로필 아이콘을 선택하세요.
2. 팝업 메뉴가 나타나면 **계정 관리** 버튼을 선택하세요.

3. 화면을 아래로 스크롤해 [알림 수신 설정](#) 섹션으로 이동하세요.
4. 수신 레벨, 수신 수단, 요일 및 시간을 설정한 다음 [저장](#) 버튼을 선택하세요.


경고 알림 언어 설정

프로젝트에서 발생하는 경고 알림 메시지의 언어를 변경할 수 있습니다.



1. 홈 화면에서 경고 알림 메시지의 언어를 변경할 프로젝트를 선택하세요.
2. 화면 왼쪽 메뉴에서 [관리](#) > [프로젝트 관리](#)를 선택하세요.
3. [알림 언어 관리](#) 섹션에서 원하는 언어를 선택하세요.
4. 화면 오른쪽 아래에 [저장](#) 버튼을 선택하세요.



이벤트 기록

홈 화면 > 프로젝트 선택 > 경고 알림 > 이벤트 기록

경고 알림이 발생한 이력을 확인할 수 있습니다. 최근 1년 이내의 이력까지 조회할 수 있습니다. 각 항목을 설정한 다음  버튼을 선택하세요.

GP Kubernetes Monitoring Demo(New) ▾ 이벤트 기록

시간 선택: < 2023/08/30 16:40 ~ 2023/08/30 16:45 5분 > | 필터: 제목 | 전체 선택 |  | 

 | 

	제목	이벤트 발생 시각	이벤트 해소 시각	애플리케이션	메시지
1	HITMAP_FLOOD_PATTERN	2024/01/22 09:08:00	2024/01/22 09:09:06		Flood pattern was detected in the project hitmap.
2	HITMAP_FLOOD_PATTERN	2024/01/22 08:57:00	2024/01/22 08:58:04		Flood pattern was detected in the project hitmap.
3	HITMAP_FLOOD_PATTERN	2024/01/22 08:46:00	2024/01/22 08:47:03		Flood pattern was detected in the project hitmap.
4	HITMAP_FLOOD_PATTERN	2024/01/22 08:35:00	2024/01/22 08:36:03		Flood pattern was detected in the project hitmap.
5	HITMAP_FLOOD_PATTERN	2024/01/22 08:24:00	2024/01/22 08:25:00		Flood pattern was detected in the project hitmap.
6	HITMAP_FLOOD_PATTERN	2024/01/22 08:13:00	2024/01/22 08:13:59		Flood pattern was detected in the project hitmap.
7	HITMAP_FLOOD_PATTERN	2024/01/22 08:02:00	2024/01/22 08:02:57		Flood pattern was detected in the project hitmap.
8	HITMAP_FLOOD_PATTERN	2024/01/22 07:51:00	2024/01/22 07:51:56		Flood pattern was detected in the project hitmap.
9	HITMAP_FLOOD_PATTERN	2024/01/22 07:40:00	2024/01/22 07:40:55		Flood pattern was detected in the project hitmap.
10	HITMAP_FLOOD_PATTERN	2024/01/22 07:29:00	2024/01/22 07:29:54		Flood pattern was detected in the project hitmap.
11	HITMAP_FLOOD_PATTERN	2024/01/22 07:18:00	2024/01/22 07:18:53		Flood pattern was detected in the project hitmap.
12	HITMAP_FLOOD_PATTERN	2024/01/22 07:07:00	2024/01/22 07:07:52		Flood pattern was detected in the project hitmap.
13	HITMAP_FLOOD_PATTERN	2024/01/22 06:56:00	2024/01/22 06:56:51		Flood pattern was detected in the project hitmap.
14	HITMAP_FLOOD_PATTERN	2024/01/22 06:45:00	2024/01/22 06:45:50		Flood pattern was detected in the project hitmap.
15	HITMAP_FLOOD_PATTERN	2024/01/22 06:34:00	2024/01/22 06:34:49		Flood pattern was detected in the project hitmap.
16	HITMAP_HORIZONTAL_PATTERN	2024/01/22 06:28:00	2024/01/22 06:28:48		Horizontal pattern was detected in the project hitmap.
17	HITMAP_FLOOD_PATTERN	2024/01/22 06:18:00	2024/01/22 06:18:46		Flood pattern was detected in the project hitmap.

이전 1 다음

⚠ 선택한 프로젝트에 따라 화면 이미지는 다를 수 있습니다.

• 시간 선택

- 오른쪽에 위치한 녹색 버튼을 선택해 조회 시간을 선택할 수 있습니다.
- < 또는 > 버튼을 선택해 선택한 조회 시간 만큼 간격을 이동할 수 있습니다.
- 세부 시간을 선택하려면 날짜 또는 시간 영역을 선택하세요. 세부 시간을 설정한 다음 **적용** 버튼을 선택하세요.

- **필터:** 제목 또는 메시지 내용을 기준으로 이벤트 기록을 필터링할 수 있습니다.

- **컨테이너**: 프로젝트에 포함된 에이전트를 선택할 수 있습니다.
- **CSV**: 조회한 이벤트 기록 결과를 csv 파일로 저장할 수 있습니다. **CSV** 버튼을 클릭하면 **최대 CSV 라인 수**를 입력한 다음 **다운로드** 버튼을 선택하세요.
- **컬럼 선택**: 조회한 이벤트 기록 결과의 열 항목을 추가할 수 있습니다.
- **이벤트 설정**: **경고 알림 > 이벤트 설정** 메뉴로 이동합니다.
- **제목 / 메시지**: **이벤트 설정** 메뉴에서 추가한 이벤트의 **이벤트명**과 **메시지** 항목의 내용입니다.

- ① ○ 이벤트 추가에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.
- 알림 메시지 사용자 정의에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

- **이벤트 발생 시각**: 이벤트가 발생한 시각입니다.
 - 이벤트가 해소되지 않고 진행 중일 경우 **진행 중** 태그가 표시됩니다.
 - 정비 중인 경우 이벤트가 발생하면 **정비 중 발생** 태그가 표시됩니다.

- ① **정비 계획**에 대한 자세한 내용은 [다음 문서](#)를 참조하세요.

- **이벤트 해소 시각**: 설정한 이벤트가 해결된 시각입니다. 만약 해당 컬럼이 보이지 않는다면 **컬럼 선택** 버튼을 클릭한 다음 **이벤트 해소 시각**을 선택하세요.
- **애플리케이션**: **이벤트 설정** 메뉴에서 이벤트 추가 시 **이벤트 대상 필터링** 항목을 설정하면 표시됩니다.