

Kubernetes Monitoring

release date. 2026. 09. 11.

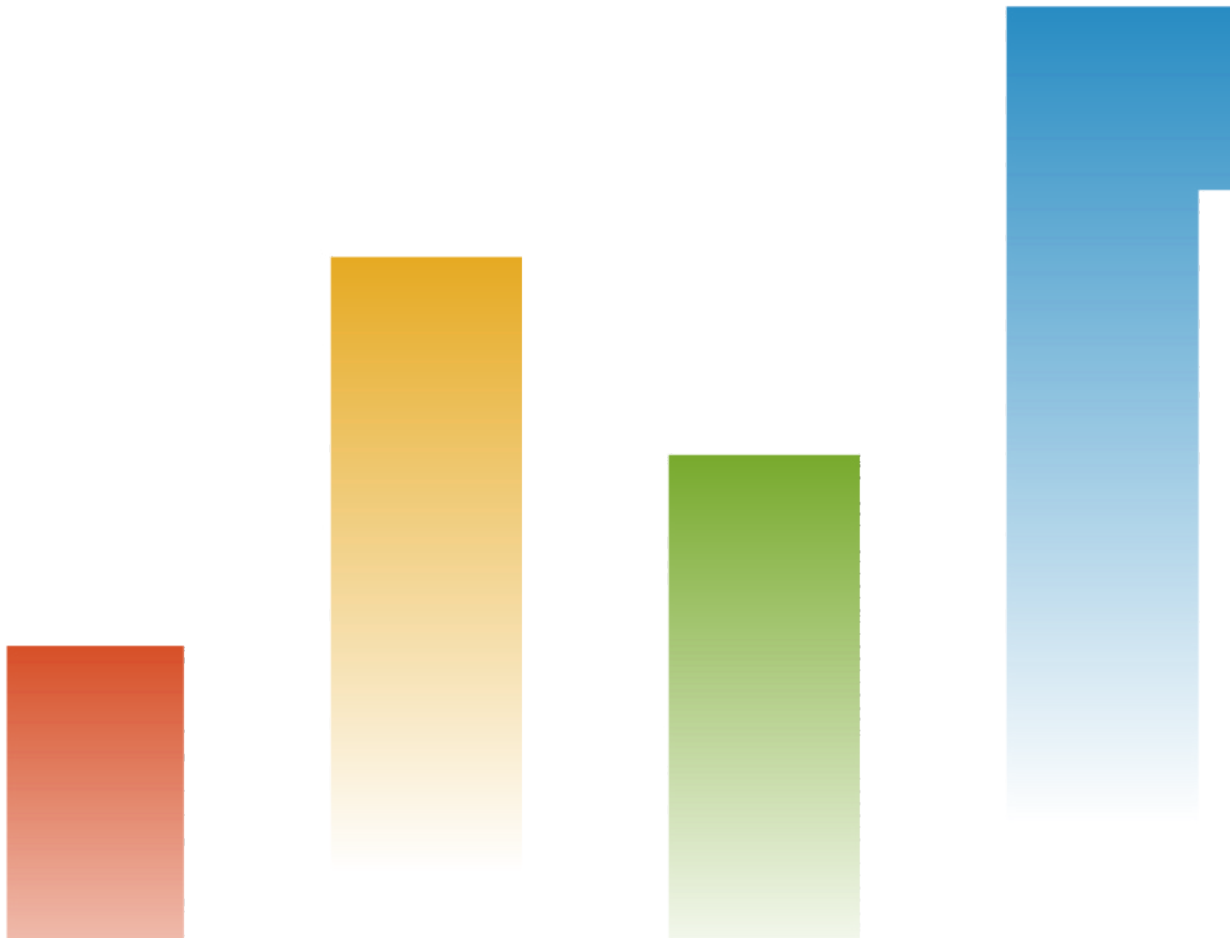


Table Of Contents

• Kubernetes Monitoring	7
◦ Getting started.....	10
◦ Supported environment	13
◦ Installation.....	19
■ WhaTap Operator	23
■ Install WhaTap Operator.....	26
■ APM Auto-Installation Configuration	39
■ OpenAgent Configuration	49
■ WhaTap CR configuration options.....	73
■ Installation of the master and node agents	81
■ Application Agent Installation	99
■ Installing Docker Java.....	102
■ Installing Docker Node.js	115
■ Installing Docker Python	119
■ Installing the docker PHP	125
■ Installing the docker Go	129
■ Installation troubleshooting	140
◦ Configuration	146
◦ Management	
■ Namespace management.....	161
■ Correlated project management.....	164
■ Kubernetes agent.....	167
■ Kubernetes Agent List	169
■ Agent Update.....	173

■ Agent deletion	174
◦ Dashboard	
■ Container Map	177
■ Container view, Pod view	191
■ Target grouping	195
■ Container map metrics and status	202
■ External project integration and configuration	206
■ Container Map Detailed Analysis	214
■ Target Information	218
■ Trace	222
■ Metrics	225
■ Log	231
■ Event	236
■ Call Information	244
■ Node Map	250
■ Performance Summary	264
■ Flex Board	267
■ Using Flex board	270
■ Managing the widgets	280
◦ OpenMetrics	
■ OpenMetrics Template List	285
■ OpenMetrics Explorer	287
◦ Cluster	
■ Cluster Summary	290
■ Namespace Status	296
◦ Node	

■ Node List	298
■ Node Timeline	310
■ Node Details	319
■ Node Disk List.....	321
○ GPU.....	326
■ GPU Dashboard.....	331
■ GPU Trends	336
■ GPU Service Grouping	338
○ Container	
■ Container List.....	344
■ Container Timeline.....	350
■ Container Volume	354
■ Container images	357
■ Out Of Memory Killed Container.....	359
○ Workload	
■ Pod List.....	362
■ Pod Timeline	368
■ Pending Pod Status.....	373
■ Container Application List	378
■ Deployment list	384
○ Service networking	
■ Service List	388
■ Ingress List.....	393
○ Storage	
■ PV/PVC List	397
○ Control Plane	

■ kube-apiserver Dashboard	405
■ kube-apiserver Metrics Search	414
■ etcd Dashboard	423
■ kube-scheduler Dashboard	430
◦ Analysis	
■ Object Manifest Information	438
■ Kubernetes event	446
■ Pod Startup Analysis	450
■ Pod Startup Analysis	455
■ Application Hitmap	460
■ Multi Service Hitmap	462
■ MSA Analysis	464
■ Application Stack	472
◦ Metrics	473
■ Kubernetes metrics	476
■ GPU metrics	527
■ Metrics Chart	530
■ Metrics Chart	553
■ Metric Anomaly Detection	556
■ Metrics Search	559
◦ Log	561
◦ Report	566
◦ Integrated Report	569
◦ Application	
■ Application Service Dashboard	578
■ HITMAP - Trace analysis	601

■ Daily Application Status.....	628
■ Transaction search	630
■ Instance Performance Monitoring	
■ Java	0
■ Node.js	0
■ Python	0
■ PHP	0
■ Go	0
◦ Alert	642
■ Event Configuration	0
■ Metrics	649
■ Composite Metrics.....	671
■ Hitmap Patterns.....	678
■ Anomaly Detection	684
■ Real-time logs	688
■ Composite log	701
■ Transaction	715
■ Event Configuration	727
■ Event Notification Settings.....	731
■ Event Notification Format.....	753
■ Event History and Monitoring	761
■ Real-time Notifications.....	769
■ System Events	772
◦ Laboratory	775

Kubernetes Monitoring

WhaTap Kubernetes monitoring is a unified solution that analyzes containers, pods, nodes, entire clusters, and the applications running inside containers from a single screen. It supports both multi-cloud and hybrid cloud environments, and lets infrastructure and application teams collaborate on the same data.

Kubernetes monitoring structure

The smallest deployment unit in Kubernetes is a Pod, not a container, and a Pod can contain one or more containers. Kubernetes monitoring therefore requires both a container view and a Pod view. WhaTap collects metrics for containers and Pods into separate categories and provides both views on the **Container Map**.

A category is a name that identifies the metrics collected from your project. Each category has its own Tags and Fields.

Key features

- **Unified monitoring** — Monitor Kubernetes systems across multi-cloud and hybrid cloud environments from a single screen.
- **MSA call relationship analysis** — Trace API call relationships of transactions in Pod-level distributed environments.
- **Root cause analysis** — Monitor hosts, containers, and applications together to quickly identify the root cause of incidents.
- **In-container application analysis** — Provides hitmaps, traces, and multi-transaction tracking for Java, Node.js, and Python applications.
- **Unified visibility through Container Map** — Provides Kubernetes resources, metrics, events, and logs on a single screen. Infrastructure and application teams can collaborate on the same view.
- **Resource usage trend analysis** — Provides usage trend dashboards for containers and nodes. You can drill down from the full container/node list within a WhaTap project to the details of each target.
- **Instant meta information lookup** — View meta information directly on the Container Map without accessing the master node. A separate master meta dashboard shows the meta information of all components at a

glance.

- **Focused log analysis** — Stores container log files and standard output in real time, and automates classification and indexing. You can focus on log analysis without accessing each container directly.

Try WhaTap Kubernetes monitoring through the tutorial video below.

Monitoring targets

WhaTap Kubernetes monitoring targets Kubernetes clusters and the applications running on top of them.

- **Kubernetes cluster**
 - CPU, memory, disk (volume), and network for containers
 - Cluster components such as namespaces, deployments, ReplicaSets, and pods
 - CPU, memory, disk, and network for nodes (hosts)
- **In-container applications** — Java, Node.js, Python, PHP, Go

The following monitoring procedure is divided into cluster monitoring and application monitoring.

Monitoring procedure

Cluster monitoring procedure

1. Create a **cluster project** in [WhaTap Monitoring Service](#). The first Kubernetes project you create automatically becomes the cluster project. For details, see [Getting started](#).
2. Install the master agent (`whatap-master-agent`) and node agent (`whatap-node-agent`) in your Kubernetes environment. Both agents run as pods inside your cluster. For details, see [Agent installation](#).
3. (Optional) If your cluster has many namespaces, you can split them into separate namespace projects. For details, see [Creating a namespace project](#).

✓ A Kubernetes namespace is a mechanism for isolating resource groups within a single cluster. You can

✓ list the namespaces in your cluster with the `kubectl get namespace` command.

Application monitoring procedure

To monitor applications running inside containers, install application agents separately. The supported languages are Java, Node.js, Python, PHP, and Go. For details, see [Application agent installation](#).

Getting Started

WhaTap monitoring starts with the following steps:

[Step 1. Sign up](#)

[Step 2. Select a product](#)

[Step 3. Create a project](#)

[Step 4. Check the access key](#)

After confirming the supported environment, proceed to the agent installation step.

Sign up

To use the WhaTap monitoring service, you must first [sign up](#). For more information about signing up, see the [Account Management > Sign Up](#) documentation.

Select a product

1. Log in to the [WhaTap Monitoring Service](#).
2. Select  **All Projects** > **+ Project**.
3. On the **Select product** screen, select the Go product to monitor and then click the Create button.

Create a project

Understanding the Kubernetes project structure

To start Kubernetes monitoring, first create a **cluster project**. If your cluster has many namespaces and a high number of containers and pods, you can additionally create **namespace projects** to split the monitoring scope.

Project List

Project name / PCODE 🔍 ☆ 📊 📈 ☰

☆ CP **k8s** ⚙️


Node	18	Container	77	Namespace	3
Node CPU core	516	Pod	34	Deployment	4
Node memory	4.2T	Application	0	ReplicaSet	5

1 **Kubernetes**

☆ ⚙️


Node	1	Container	1	Namespace	1
Node CPU core	32	Pod	1	Deployment	1
Node me...	264.7G	Application	0	ReplicaSet	1

2 **Namespace**

1 Cluster project

- The project that monitors all resources in a Kubernetes cluster.
- Master and node agents are installed in the **cluster project**.
- In the project list, the **cluster project** shows a **CP** tag next to its name and a **Kubernetes** tag at the bottom.
- For details on how to create one, see the [Getting started](#) document.

2 Namespace project

- A project that monitors resources and application containers at the Kubernetes namespace level.
- In the project list, the **namespace project** shows a **Namespace** tag at the bottom.
- For details on how to manage one, see the [Manage namespace projects](#) document.

Creating a Kubernetes project

After selecting the product to monitor, create a project before installing the agent.

1. On the **Create project** screen, enter or select the following items to create a project:

- **Project name** (required): Enter the project name.
- **Data server region** (required): Select the region where the data server is located. A region is a collection of data centers that provide cloud services. Your data is stored in the selected region.
- **Time zone** (required): Set the time zone that serves as the reference for generating alerts and reports.
- **Notification language setting** (required): Set the language for alert notification messages generated in the project (supports Korean and English).
- **Project groups**: You can group and manage multiple projects. Select a group if there is one to join.
- **Project description**: Enter an additional description for the project.

2. After completing all settings, click the **Creating a project** button.

ⓘ When adding a project with an organization selected, you must set the **Groups of organization** as required. For more information about groups, see [the following documentation](#).

Check the access key

The **access key** is a unique ID for activating the WhaTap service.

In the installation guide section, select the **Getting the access key** button. After automatically issuing the **access key**, proceed to the next step.

ⓘ The **access key** cannot be used duplicately in other projects, and a separate **access key** is required for **namespace projects** as well.

ⓘ After creating a project, you are automatically redirected to the **Agent installation** page. If you are not redirected to the **Agent installation** page, select **Management > Agent installation** from the left menu.

Supported Environment

Before starting Kubernetes monitoring, check the following support items. WhaTap Kubernetes can be installed on standard Kubernetes 1.16 or later and the corresponding environments.

Supported platforms

Platform	Support
Amazon EKS (Elastic Kubernetes Service)	✓
AKS (Azure Kubernetes Service)	✓
GKE (Google Kubernetes Engine)	✓
OKE (Oracle Container Engine for Kubernetes)	✓
Naver Cloud Kubernetes Service	✓
NHN Kubernetes Service (NKS)	✓
Red Hat OpenShift (3.x / 4.x)	✓
Cocktail	✓
Accordion	✓
PaaS-TA	✓
Kakao i Cloud	✓
Rancher	✓

! WhaTap can be installed in any Standard Kubernetes environment that is certified by the CNCF (Cloud

! Native Computing Foundation).

NVIDIA GPU

The WhaTap Kubernetes node agent collects GPU metrics through NVIDIA's Data Center GPU Manager (DCGM) Exporter. The collection is configured using a sidecar pattern.

- **Sidecar pattern**

DCGM Exporter runs as a secondary container in the same Pod as the main application container. This sidecar pattern lets DCGM Exporter collect GPU status information efficiently.

- **DCGM Exporter**

The `dcgm-exporter` container collects GPU status and performance metrics through DCGM.

- **Metric collection and transmission**

The `whatap-node-agent` container requests and collects GPU metrics from the HTTP endpoint of `dcgm-exporter`.

! The HTTP endpoint of `dcgm-exporter` typically uses port 9400.

- **MIG environment support**

WhaTap Kubernetes supports the **MIG (Multi-Instance GPU)** feature in NVIDIA GPU clusters, providing **fine-grained monitoring not only by physical GPU but also by MIG instance**.

MIG mode lets you partition a single physical GPU into multiple logical instances to isolate workloads and allocate resources. WhaTap supports accurate metric collection and visualization in such environments as well.

! GPU monitoring requires this feature to be enabled. For details, see [GPU monitoring installation](#).

For the list of collected GPU metrics, see [GPU metrics](#).

Browser

WhaTap Monitoring is available in web browsers and mobile Apps.

Browser	Recommended	Supported version
Google Chrome	O	111 or later
Mozilla FireFox	X	Latest version
Edge	X	Latest version
Safari	X	Latest version

Support guidance

- For browser compatibility and performance reasons, the latest version of Chrome is recommended.
- The user interface (UI) is implemented using HTML5 standard technologies, so Internet Explorer is not supported.

Limitations

WhaTap's web interface does not support mobile browsers. To access WhaTap on a mobile device, install the WhaTap app for Android or iOS. The WhaTap mobile app is optimized for mobile environments. For more information about the WhaTap mobile app, see [this document](#).

Service deployment models

You can use WhaTap Monitoring as a SaaS or on-premises (self-hosted) service. Whichever you choose, you use the same agents and the same monitoring features.

Deployment	Support	Description
SaaS	✓	Connects to collection servers operated by WhaTap. See the firewall section for regional addresses
On-premises (self-hosted)	✓	Collection servers you build and operate yourself (air-gapped environments supported). Contact your sales representative for the scope

ⓘ You can also deploy WhaTap on-premises in air-gapped environments without internet connectivity.

Feature differences by deployment model

Whichever you choose, the monitoring features are the same. The mobile app is the only exception and is available on the service.whatap.io SaaS service only.

ⓘ To use certain features on-premises, a separate agreement is required.

Firewall

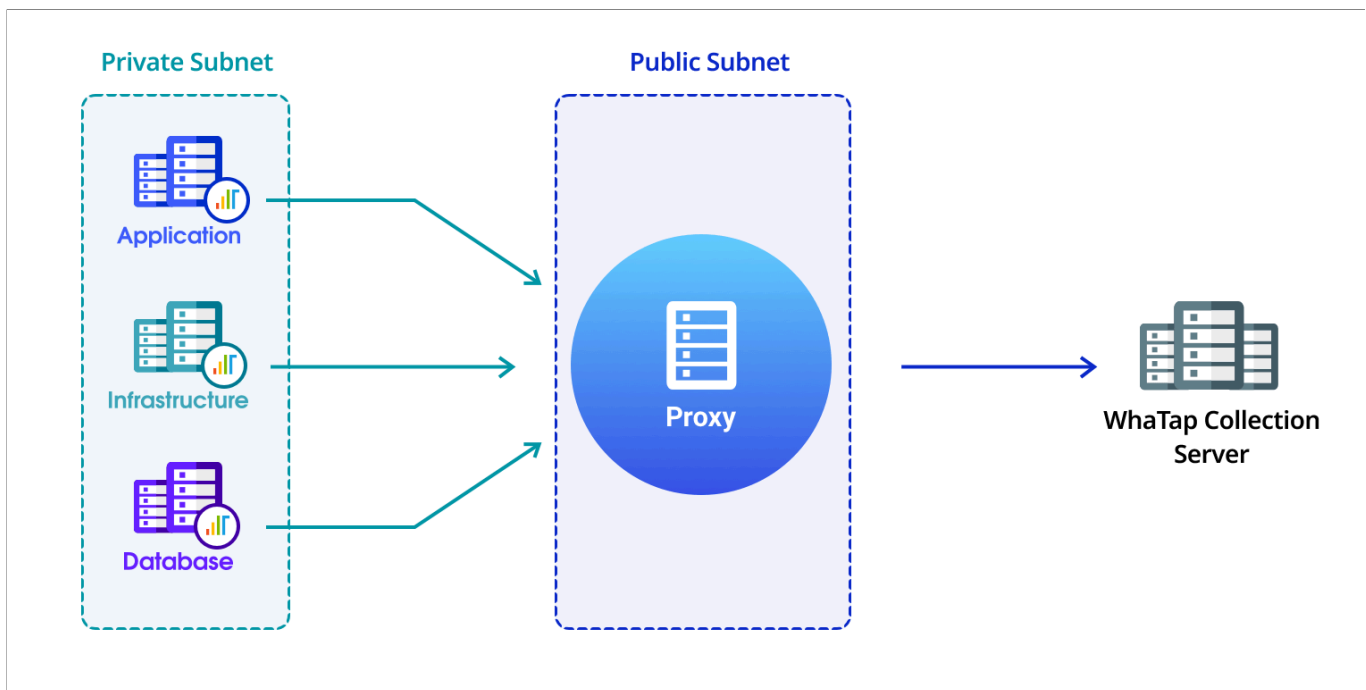
The WhaTap agent must be able to access the collection server through **TCP port 6600**. Allow access to the collection server address that is closest to the monitored target.

Source: WhaTap agent

Cloud	Destination	Destination IP	Port
AWS	WhaTap collection server (Seoul)	13.124.11.223 / 13.209.172.35	TCP 6600
	WhaTap collection server (Tokyo)	52.68.36.166 / 52.193.60.176	TCP 6600
	WhaTap collection server (Singapore)	18.138.0.93 / 18.139.67.236	TCP 6600
	WhaTap collection server (Jakarta)	108.136.91.69 / 108.137.158.44	TCP 6600

Cloud	Destination	Destination IP	Port
	WhaTap collection server (Thailand)	43.209.109.219 / 43.209.48.86	TCP 6600
	WhaTap collection server (Mumbai)	13.127.125.69 / 13.235.15.118	TCP 6600
	WhaTap collection server (California)	52.8.223.130 / 52.8.239.99	TCP 6600
	WhaTap collection server (Virginia)	107.23.220.101 / 54.236.221.105	TCP 6600
	WhaTap collection server (Frankfurt)	3.125.142.162 / 3.127.76.140	TCP 6600
Azure	WhaTap collection server (Seoul)	52.231.66.38 / 20.194.5.115	TCP 6600
	WhaTap collection server (Tokyo)	52.246.169.54 / 20.210.27.232	TCP 6600
Kakao	WhaTap collection server (Seoul)	61.109.237.237 / 61.109.238.166	TCP 6600

If the agent cannot directly access the collection server, use the provided proxy module to access it.



Mobile app

The WhaTap mobile app supports both Android and iOS environments. You can install the WhaTap app by following the link or scanning the QR code. For detailed information about the mobile app, see [the following](#).

iOS	Android
 iOS 12 or later	 Android 5.0 or later

Agent Installation

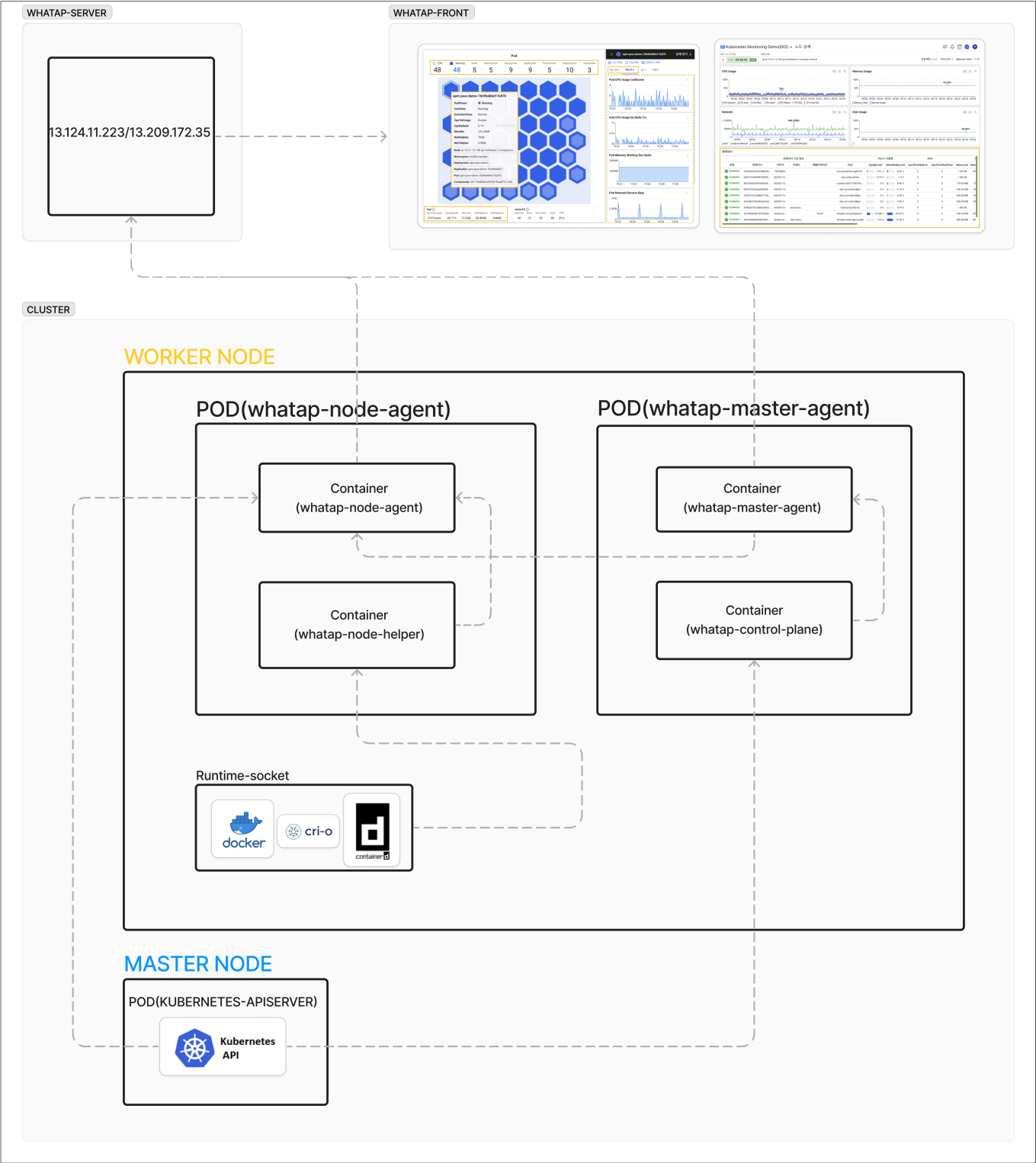
This page introduces the WhaTap Kubernetes monitoring agent architecture and the two installation methods. Review the prerequisites and architecture first, then choose the installation method that fits your environment.

ⓘ Before installing the agent, complete the following steps in the [Getting started](#) document.

- Sign up for WhaTap
- Create a **cluster project**
- Issue an access key

Kubernetes agent architecture

The WhaTap Kubernetes agent consists of master agent pods and node agent pods.



Node agent pod

The `whatap-node-agent` pod is deployed as a DaemonSet, with one pod running per node. The pod contains the following two containers.

- **whatap-node-agent**

Communicates with kube-apiserver to collect metrics for the workloads on the node. These metrics power categories such as `kube_pod` , `container` , and `kube_pod_volume` .

- **whatap-node-helper**

Mounts the runtime socket and root volume to collect resource usage of the node and its containers.

Master agent pod

The `whatap-master-agent` pod is deployed as a Deployment, with one pod running per cluster. The pod includes the following containers.

- **whatap-master-agent** (required)

Collects cluster-wide metrics. It also delivers and synchronizes shared data, such as namespace project information, to each node agent.

- **whatap-control-plane** (optional)

Collects metrics from Kubernetes control plane nodes. Currently it collects apiserver state and request count metrics. Enabling control plane monitoring requires an additional feature configuration. For details, see the [Control plane dashboard](#) document.

ⓘ In a future release, multiple master agent pods will be able to run simultaneously through a high-availability feature.

Choosing an installation method

WhaTap Kubernetes monitoring supports the following two installation methods. Choose the one that fits your operating environment and management style.

Install with WhaTap Operator

[WhaTap Operator](#) is a Kubernetes operator that manages master and node agents, application auto-instrumentation, and OpenAgent through a single custom resource (CR). It continuously watches the CR state and reconciles changes automatically.

See the following pages for installation and configuration.

- [WhaTap Operator installation](#) — Deploys the operator with a Helm chart and creates a `WhatapAgent` CR.
- [Application agent auto-injection](#) — Auto-injects APM agents into in-container applications via CR settings.
- [OpenMetrics collection with Open Agent](#) — Collects OpenMetrics data from external exporters.
- [WhatapAgent CR configuration options](#) — Reference of all available CR settings.

Install master and node agents directly

Deploy master and node agents using YAML or Helm chart by following the **Installation guide** page in the console. You can choose container runtime, Kubernetes version, and environment options (GKE, OpenShift, etc.) step by step.

- [Master and node agent installation](#)

Additional application agent installation

Whichever installation method you choose, you need to install application agents separately to monitor Java, Node.js, Python, PHP, or Go applications running inside containers.

- [Application agent installation](#)

Installation video

WhaTap Operator

WhaTap Operator is an open-source Kubernetes operator that enables deployment and configuration of WhaTap agents in Kubernetes environments.

Using the operator, you can integrate and manage all features from K8s and GPU (MIG) monitoring to auto-instrumentation and **OpenMetrics** collection through a single custom resource (CRD). In this process, the operator automatically validates the resources to be deployed, minimizing the possibility of errors in complex configurations and supporting stable operations.

Key Features

- **Deployment through Custom Resources:** Deploy and manage WhaTap agents and related components using Kubernetes custom resources.
- **Simplified Deployment Configuration:** Easily configure essential elements such as WhaTap agent version, resource requests/limits, and monitoring target registration in native Kubernetes resources.
- **Automatic APM Instrumentation:** Automatically inject APM agents into specific pods using Kubernetes standard label selectors.
- **OpenMetrics Collection through Open Agent:** Collect and utilize **OpenMetrics** data through Open Agent installation.
- **Integrated Monitoring Management:** Configure and manage both application performance monitoring (APM) and Kubernetes infrastructure monitoring at once through a single CR, enhancing operational efficiency.

Goals of WhaTap Operator

WhaTap Operator focuses on reducing the complexity of monitoring configuration and simplifying management in Kubernetes environments.

- **Reduced Effort in Agent Installation and Configuration:** Significantly reduces the installation and management burden of WhaTap monitoring agents.
- **Automation through Kubernetes Native Resources:** Automatically manage settings for WhaTap monitoring

targets using Kubernetes CRD.

```
apiVersion: monitoring.whatap.com/v2alpha1
kind: WhatapAgent
metadata:
  name: whatap
spec:
  features:
    apm:
      instrumentation:
        targets:
          - name: "java-app"
            enabled: true
            language: "java"
            whatapApmVersions:
              java: "2.2.58"
          # Selector configuration...
```

- **Configuration Abstraction and Validation:** Simplifies complex monitoring configurations and supports standard Kubernetes selectors (matchLabels, matchExpressions). Additionally, configuration validation helps minimize errors and support stable operations.

✓ Why use WhaTap Operator instead of Helm charts or manual DaemonSet installation?

While you can install WhaTap agents through Helm charts or DaemonSets, WhaTap Operator provides the following advantages.

1. Automatic State Reconciliation

While Helm charts or DaemonSets require manual intervention for state changes, WhaTap Operator is included in the Kubernetes **reconciliation loop** to continuously monitor and automatically adjust the CR's state. For example, if a resource created by the CR is accidentally deleted or changed, the operator detects it and automatically recovers to the state defined in the CR.

2. Minimized Configuration Errors

Manual installation methods have a high possibility of errors. WhaTap Operator performs thorough validation of agent configurations to minimize such errors.



3. Integrated Management

You can integrate and manage WhaTap monitoring component configurations within the K8s cluster in a single file, improving operational efficiency.

4. Kubernetes Standards Compliance

The operator is treated as a first-class resource in the Kubernetes API and fully supports Kubernetes standard label selectors. This allows flexible selection of monitoring targets using familiar Kubernetes patterns.

Install WhaTap Operator

This guide explains how to install the WhaTap Operator in a Kubernetes environment and configure the `WhatapAgent` CR to start monitoring with WhaTap.

Prerequisites

- Kubernetes cluster (v1.19+)
- Helm 3.2 or later
- WhaTap account and license key

- ❗ • The WhaTap agent is installed in the **whatap-monitoring** namespace. This namespace cannot be changed.
- The collection server address and port must be reachable so that data can be sent (TCP Outbound). You can find the address and port on the **Management > Agent installation** screen of the WhaTap monitoring service.

Installation

Follow the steps below to install the WhaTap Operator and create the `WhatapAgent` CR.

- ✅ On the **Management > Agent installation > Kubernetes** screen of the WhaTap monitoring service, selecting **Operator** as the installation method generates the commands below along with a `WhatapAgent` CR file ([whatap_operator.yaml](#)). Choose the image version, container runtime, environment (GKE, OpenShift), and optional features (GPU monitoring, APM auto-installation, OpenMetrics collection) on the screen, and download the file with those settings applied. Use this document when you write the file yourself or edit a downloaded one.

1. Install the WhaTap Operator

- If you are installing the agent for the first time, run the following commands:

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

```
kubectl create ns whatap-monitoring
export WHATAP_HOST=<collector IP>
export WHATAP_LICENSE=<license key>
export WHATAP_PORT=<collector port>
kubectl create secret generic whatap-credentials --namespace whatap-monitoring --from-literal
WHATAP_LICENSE=$WHATAP_LICENSE --from-literal WHATAP_HOST=$WHATAP_HOST --from-literal
WHATAP_PORT=$WHATAP_PORT
helm install whatap-operator whatap/whatap-operator --namespace whatap-monitoring
```

✓ Advanced configuration

Image registry and pull secret

If you use a private registry or need to adjust image tags or the pull policy, set the following values in [values.yaml](#).

```
image:
  repository: public.ecr.aws/whatap/whatap-operator
  tag: latest
  pullPolicy: Always

# For private registry
imagePullSecret:
  name: <your-registry-secret-name>
```

Scheduling and priority options

You can control scheduling constraints and priority for the operator Pod. The template renders only when a value is specified, which preserves compatibility with existing users.

```
priorityClassName: "" # Example: system-cluster-critical
```



```
nodeName: ""           # When you need fixed scheduling on a specific node
nodeSelector: {}       # Example: { kubernetes.io/os: linux }
affinity: {}           # Standard K8s Affinity schema
tolerations: []        # Standard K8s Toleration list
```

- **Example**

```
priorityClassName: "high-priority"
nodeSelector:
  nodepool: system

affinity:
  nodeAffinity:
    requiredDuringSchedulingIgnoredDuringExecution:
      nodeSelectorTerms:
        - matchExpressions:
            - key: kubernetes.io/os
              operator: In
              values: ["linux"]

tolerations:
  - key: "node-role.kubernetes.io/control-plane"
    effect: NoSchedule
```

If you use a custom configuration instead of the default [values.yaml](#), install the operator with the following commands.

- If the namespace already exists

```
helm install whatap-operator whatap/whatap-operator -n whatap-monitoring -f
custom-values.yaml
```

- If the namespace does not exist, create it automatically and install

```
helm upgrade --install whatap-operator whatap/whatap-operator -n whatap-monitoring
```




```
--create-namespace -f custom-values.yaml
```

- If you were previously using the WhaTap Kubernetes agent, run the following commands to install the agent:

```
kubectl delete ns whatap-monitoring
kubectl delete clusterrole whatap
kubectl delete clusterrolebinding whatap
kubectl create ns whatap-monitoring
kubectl create secret generic whatap-credentials --namespace whatap-monitoring --from-literal
WHATAP_LICENSE=$WHATAP_LICENSE --from-literal WHATAP_HOST=$WHATAP_HOST --from-literal
WHATAP_PORT=$WHATAP_PORT
helm install whatap-operator whatap/whatap-operator --namespace whatap-monitoring
```

- ⚠ If the previous WhaTap Kubernetes agent was installed via [yaml](#) files or Helm, a clean install is required. It is recommended to uninstall the existing Kubernetes agent before using the Operator.

- The operator is deployed as a deployment. Run the following command to check if the operator is running:

```
kubectl get pods -n whatap-monitoring | grep -i operator
```

- The result should show the `whatap-operator` pod in **Running** status.

```
whatap-operator-pod
```

2. Create the WhatapAgent CR

- Apply the [whatap_operator.yaml](#) file downloaded from the installation screen to create the `WhatapAgent` custom resource. If you wrote the CR file yourself, use that file name.

```
kubectl apply -f whatap_operator.yaml
```

- The `WhatapAgent` CR defines deployment and configuration for WhaTap agents. It supports setting up the Kubernetes agent, auto-installing APM, and collecting OpenMetrics via the Open Agent.

Minimal Configuration

This is a minimal configuration that enables the WhaTap master and node agents for basic Kubernetes monitoring.

whatap_operator.yaml

```
apiVersion: monitoring.whatap.com/v2alpha1
kind: WhatapAgent
metadata:
  name: whatap
  namespace: whatap-monitoring
spec:
  features:
    k8sAgent:
      masterAgent:
        enabled: true
      nodeAgent:
        enabled: true
        runtime: containerd           # Container runtime on the node
        runtimeSocketPath: /var/run/containerd/containerd.sock # Container runtime socket path
    gpuMonitoring:
      enabled: false
```

Set `runtime` and `runtimeSocketPath` to match the container runtime on your nodes. Selecting the container runtime on the installation screen applies the values below automatically.

Container runtime	runtime	runtimeSocketPath
containerd (default)	containerd	/var/run/containerd/containerd.sock

Container runtime	runtime	runtimeSocketPath
Docker Engine	docker	/var/run/docker.sock
CRI-O	crio	/var/run/crio/crio.sock

In a GKE environment, add the following setting under `k8sAgent` .

```
k8sAgent:
  gke:
    enabled: true
```

When installed with the minimal configuration, `whatap-master-agent` and `whatap-node-agent` are additionally deployed as shown below.

```
whatap-master-agent, whatap-node-agent
```

ⓘ When installing the WhaTap agent with the operator, refer to [WhaTap CR configuration options](#) for details on CR settings (resource constraints, tolerations, imagePullSecret).

Default Configuration

```
apiVersion: monitoring.whatap.com/v2alpha1
kind: WhatapAgent
metadata:
  name: whatap
spec:
  features:
    ### Uncomment to enable APM auto-installation -
    ### Automatically injects the APM agent into application Pods to enable performance monitoring.
    # apm:
    # instrumentation:
    # targets:
    #   - name: hello-world
    #   enabled: true
```

```

#   language: "java"      # Supported languages: java, python, nodejs
#   whatapApmVersions:
#     java: "2.2.58"      # (Optional) Pin a specific APM agent version
#   namespaceSelector:
#     matchNames:
#       - default          # Namespace containing the application to monitor
#   podSelector:
#     matchLabels:
#       app: "hello-world" # Label of the application Pod to monitor
#   config:
#     mode: default        # APM agent mode setting

### Uncomment to enable Kubernetes monitoring -
### Enables monitoring of Kubernetes cluster, nodes, and containers.
# k8sAgent:
#   masterAgent:
#     enabled: true        # Enable master agent (cluster-level metric collection)
#   nodeAgent:
#     enabled: true        # Enable node agent (node and container-level metrics)
#     runtime: containerd  # Container runtime on the node (containerd, docker, crio)
#     runtimeSocketPath: /var/run/containerd/containerd.sock # Container runtime socket path

### Uncomment to enable GPU monitoring -
### Enables NVIDIA GPU metrics collection.
# gpuMonitoring:
#   enabled: true          # Enable GPU monitoring (NVIDIA DCGM-EXPORTER is installed as a sidecar in the whatap-node-agent)

### Uncomment to enable OpenMetrics collection (Prometheus-compatible format) -
### Collects metrics in Prometheus format.
# openAgent:
#   enabled: true          # Enable OpenAgent
#   targets:
#     - targetName: kube-apiserver
#       type: ServiceMonitor # Target type: ServiceMonitor, PodMonitor, StaticEndpoints
#     namespaceSelector:
#       matchNames:
#         - "default"       # Namespace from which to collect metrics
#     selector:
#       matchLabels:
#         component: apiserver # Labels of the service/Pod to monitor

```

```
# provider: kubernetes
# endpoints:
#   - port: "https" # Metric endpoint port
#   path: "/metrics" # Metric path
#   interval: "30s" # Scraping interval for this endpoint
#   scheme: "https" # HTTP scheme (http or https)
#   tlsConfig:
#     insecureSkipVerify: true # Skip TLS certificate verification
#   metricRelabelConfigs:
#     - source_labels: ["__name__"]
#       regex: "apiserver_request_total" # Filter to collect specific metrics
#       action: "keep" # Keep only matching metrics
```

ⓘ The Operator installation method does not support Istio sidecar injection settings. Use the Helm or YAML installation method if you need to configure an Istio environment. Control plane metrics such as kube-apiserver are collected through the OpenAgent configuration. For details, see [OpenAgent configuration](#).

Example Configurations

In addition to supporting Kubernetes agents, the WhaTap Operator also provides APM auto-installation and custom metric agent setup. Use the following examples to deploy various configurations.

GPU Monitoring

Instead of installing the Kubernetes monitoring agent, you can deploy the DCGM-EXPORTER container as a sidecar to the WhaTap node agent Pod.

Without additional configuration, simply apply the yaml below to collect metrics that are displayed in the **GPU dashboard**.

ⓘ If you configured DCGM-EXPORTER manually instead of using the built-in sidecar in the node agent, refer to the [OpenAgent Configuration](#) to collect dcgm metrics.

GPU Monitoring Auto Setup

GPU Monitoring Auto Setup

```
apiVersion: monitoring.whatap.com/v2alpha1
kind: WhatapAgent
metadata:
  name: whatap
spec:
  features:
    openAgent:
      enabled: true # Enable OpenAgent
    k8sAgent:
    masterAgent:
      enabled: true
    nodeAgent:
      enabled: true
    gpuMonitoring:
      enabled: true # Enable GPU monitoring
```

Note: During auto-installation, the Open Agent automatically detects and scrapes the DCGM Exporter.

GPU Node Toleration

If your GPU node has taints, you must add toleration settings.

GPU Node Toleration

```
apiVersion: monitoring.whatap.com/v2alpha1
kind: WhatapAgent
metadata:
  name: whatap
spec:
  features:
    k8sAgent:
    nodeAgent:
      enabled: true
      tolerations:
        - key: "nvidia.com/gpu"
          operator: "Exists"
```

```

    effect: "NoSchedule"
  - key: "gpu"
    operator: "Exists"
    effect: "NoSchedule"
gpuMonitoring:
  enabled: true

```

APM Auto Installation

Automatically install APM in the Kubernetes cluster without enabling Kubernetes monitoring agents.

❗ For more details, refer to the [APM Auto-Installation Guide](#).

APM Auto Installation

```

apiVersion: monitoring.whatap.com/v2alpha1
kind: WhatapAgent
metadata:
  name: whatap
spec:
  features:
    apm:
      instrumentation:
        targets:
          - name: hello-world
            enabled: true
            language: "java"
            whatapApmVersions:
              java: "2.2.58"
            namespaceSelector:
              matchNames:
                - default
            podSelector:
              matchLabels:
                app: "hello-world"
      config:
        mode: default

```

OpenAgent

Configure only the OpenAgent component to collect Prometheus-style metrics, without enabling Kubernetes monitoring agents or APM instrumentation.

ⓘ For more details, refer to the [OpenAgent Configuration](#).

```
apiVersion: monitoring.whatap.com/v2alpha1
kind: WhatapAgent
metadata:
  name: whatap
spec:
  features:
    openAgent:
      enabled: true
      targets:
        - targetName: kube-apiserver
          type: ServiceMonitor
          namespaceSelector:
            matchNames:
              - "default"
          selector:
            matchLabels:
              component: apiserver
              provider: kubernetes
          endpoints:
            - port: "https"
              path: "/metrics"
              interval: "30s"
              scheme: "https"
              tlsConfig:
                insecureSkipVerify: true
          metricRelabelConfigs:
            - source_labels: ["__name__"]
              regex: "apiserver_request_total"
              action: "keep"
```


Using Kubernetes Monitoring and APM Together

```
apiVersion: monitoring.whatap.com/v2alpha1
kind: WhatapAgent
metadata:
  name: whatap
spec:
  features:
    apm:
      instrumentation:
        targets:
          - name: hello-world
            enabled: true
            language: "java"
        whatapApmVersions:
          java: "2.2.58"
        namespaceSelector:
          matchNames:
            - default
        podSelector:
          matchLabels:
            app: "hello-world"
        config:
          mode: default
    k8sAgent:
      masterAgent:
        enabled: true
      nodeAgent:
        enabled: true
    openAgent:
      enabled: true
      targets:
        - targetName: kube-apiserver
          type: ServiceMonitor
          namespaceSelector:
            matchNames:
              - "default"
      selector:
        matchLabels:
```

```
    component: apiserver
    provider: kubernetes
endpoints:
- port: "https"
  path: "/metrics"
  interval: "30s"
  scheme: "https"
tlsConfig:
  insecureSkipVerify: true
metricRelabelConfigs:
- source_labels: ["__name__"]
  regex: "apiserver_request_total"
  action: "keep"
```

APM Auto-Installation Configuration

After installing the WhatapOperator, you can define the `WhatapAgent` CR (Custom Resource) to automatically inject the APM agent into specific Kubernetes Pods or Namespaces.

Prerequisites

- A running Kubernetes environment
- Helm version 3.2 or higher
- WhaTap license key (required)
- WhaTap collector information (IP address and port)
- whatap-operator (see the [Operator Installation](#) guide)

Basic CR Configuration

! The CR examples in this document were verified against whatap-operator Helm chart **1.9.9**. The CR structure can differ between chart versions, so run `kubectl explain whatapagent.spec` if your installed version differs.

Below is a basic sample configuration for the CR.

For more advanced settings, refer to [Example: APM Configuration](#).

```
apiVersion: monitoring.whatap.com/v2alpha1
kind: WhatapAgent
metadata:
  name: default          # Resource name
  namespace: whatap-monitoring # Target namespace for installation
spec:
  features:
    apm:
      instrumentation:
```

```

targets:
- name: "java-sample"           # APM target name (for identification)
  enabled: "true"               # Whether auto-injection is enabled
  language: "java"              # Language: java, python, nodejs, etc.
  whatapApmVersions:
    java: "2.2.58"             # APM agent version for the specified language
  namespaceSelector:
    matchNames:
      - default                 # Target namespace for APM injection
  podSelector:
    matchLabels:
      app: hello-world          # Pod label to match for injection
  config:
    mode: default               # 'default' or 'custom'. 'custom' requires configMapRef
  envs:
    WHATAP_JAVA_AGENT_PATH: /whatap-agent/whatap.agent.java.jar
    OKIND: hello

```

Example: APM Configuration

Example 1: Auto-inject APM into Pods in a Specific Namespace

- Automatically inject the Java APM agent into Pods that exist in the `backend` namespace and have the label `app=shop`.
- Currently, Whatap APM supports only the `latest` version of the Java agent. (Support for other languages will be added in the future.)
- Configuration values are provided via a `ConfigMapRef`. (The configMap must exist in the same namespace as the application.)

```

apiVersion: monitoring.whatap.com/v2alpha1
kind: WhatapAgent
metadata:
  name: whatap
  namespace: whatap-monitoring
spec:
  features:
    apm:

```

```

instrumentation:
  targets:
    - name: shop-api
      enabled: "true"
      language: "java"
  whatapApmVersions:
    java: "latest"
  namespaceSelector:
    matchNames:
      - backend
  podSelector:
    matchLabels:
      app: shop
  config:
    mode: custom
    configMapRef:
      name: shop-api-config

```

shop-api-config.yaml

```

apiVersion: v1
kind: ConfigMap
metadata:
  name: shop-api-config
  namespace: backend
data:
  whatap.conf: |
    mtrace_enabled=true
    mtrace_basetime=0
    trace_normalize_enabled=true

```

Example 2: Monitor Different Workloads Using Separate ConfigMaps

Inject different `ConfigMapRef` settings (`request-router-config` , `shop-api-config`) into the `app=router` Pod in the `system` namespace and the `app=shop` Pod in the `backend` namespace, respectively.

```

apiVersion: monitoring.whatap.com/v2alpha1
kind: WhatapAgent
metadata:

```

```
name: whatap
namespace: whatap-monitoring
spec:
  features:
    apm:
      instrumentation:
        targets:
          - name: request-router
            enabled: "true"
            language: "java"
            whatapApmVersions:
              java: "latest"
            namespaceSelector:
              matchNames:
                - system
            podSelector:
              matchLabels:
                app: router
            config:
              mode: custom
              configMapRef:
                name: request-router-config
          - name: shop-api
            enabled: "true"
            language: "java"
            whatapApmVersions:
              java: "latest"
            namespaceSelector:
              matchNames:
                - backend
            podSelector:
              matchLabels:
                app: shop
            config:
              mode: custom
              configMapRef:
                name: shop-api-config
```

shop-api-config.yaml

```

apiVersion: v1
kind: ConfigMap
metadata:
  name: shop-api-config
  namespace: backend
data:
  whatap.conf: |
    mtrace_enabled=true
    mtrace_basetime=0
    trace_normalize_enabled=true

```

request-router-config.yaml

```

apiVersion: v1
kind: ConfigMap
metadata:
  name: request-router-config
  namespace: backend
data:
  whatap.conf: |
    mtrace_enabled=false
    mtrace_basetime=100
    trace_normalize_enabled=true

```

Fields

```

apiVersion: monitoring.whatap.com/v2alpha1
kind: WhatapAgent
metadata:
  name: whatap
spec:
  features:
    apm:
      instrumentation:
        targets:
          - name: <name>
            enabled: <true | false>
            language: <java | python | nodejs>
        whatapApmVersions:

```

```

<language> : <version>
namespaceSelector:
  matchNames:
    - <namespace>
podSelector:
  matchLabels:
    <key>: <values>
config:
  mode: default

```

Field	Description
<code>name</code>	Target name. Used only for identification; it does not affect monitoring.
<code>enabled</code>	If set to <code>"true"</code> , the APM agent will be automatically injected into the target.
<code>language</code>	Programming language of the target application. Examples: <code>java</code> , <code>python</code> , <code>nodejs</code> .
<code>whatapApmVersions</code>	Specify the WhaTap APM agent version for each language.
<code>namespaceSelector</code>	List of namespaces to target for APM injection.
<code>podSelector</code>	Select Pods to inject APM using label-based matching.
<code>config.mode</code>	<code>"default"</code> (default) or <code>"custom"</code> . Setting <code>"custom"</code> requires <code>config.configMapRef.name</code> .

Selectors

`WhatapAgent` supports Kubernetes standard label selectors.

namespaceSelector

Field	Description
<code>matchNames</code>	List of namespace names to target.

Field	Description
<code>matchLabels</code>	Select namespaces using key-value label pairs (AND condition).
<code>matchExpressions</code>	Advanced label selector expressions (AND condition).

podSelector

Field	Description
<code>matchLabels</code>	Select Pods using key-value label pairs (AND condition).
<code>matchExpressions</code>	Advanced label selector expressions (AND condition).

matchExpressions Operators

Operator	Description
<code>In</code>	The label value must be in the specified list.
<code>NotIn</code>	The label value must not be in the specified list.
<code>Exists</code>	The specified label key must exist (leave <code>values</code> field empty).
<code>DoesNotExist</code>	The specified label key must not exist (leave <code>values</code> field empty).

K8sAgent Configuration

Field	Description
<code>agentImageVersion</code>	Agent image version (default: <code>"latest"</code>).
<code>masterAgent.enabled</code>	Whether to enable the master agent.
<code>masterAgent.resources</code>	Resource requests and limits for the master agent.

Field	Description
<code>masterAgent.envs</code>	List of environment variables to add to the master agent.
<code>nodeAgent.enabled</code>	Whether to enable the node agent.
<code>nodeAgent.resources</code>	Resource requests and limits for the node agent.
<code>nodeAgent.envs</code>	List of environment variables to add to the node agent.
<code>gpuMonitoring.enabled</code>	Whether to enable GPU monitoring.
<code>apiserverMonitoring.enabled</code>	Whether to enable API server monitoring.
<code>etcdMonitoring.enabled</code>	Whether to enable etcd monitoring.
<code>schedulerMonitoring.enabled</code>	Whether to enable scheduler monitoring.

Example of a selector combining multiple conditions

```
namespaceSelector:
  matchNames:
    - production
    - staging
  matchLabels:
    environment: production
  matchExpressions:
    - {key: tier, operator: In, values: [frontend, backend]}
    - {key: restricted, operator: DoesNotExist}

podSelector:
  matchLabels:
    app: web-service
    version: v2
  matchExpressions:
    - {key: tier, operator: In, values: [backend, backend2, backend3]}
    - {key: environment, operator: NotIn, values: [dev, test]}
    - {key: ready, operator: Exists}
```

• Detailed Explanation: namespaceSelector

The namespace selector filters target namespaces using three methods:

1. **matchNames:** Select namespaces by name directly
 1. Only selects namespaces named `production` or `staging` .
 2. This is the most basic form of filtering.
2. **matchLabels:** Match labels using key-value pairs (AND condition)
 1. Only selects namespaces that have the label `environment=production` .
 2. All key-value pairs must match (i.e., multiple labels must all be satisfied).
3. **matchExpressions:** Define complex conditions using advanced expressions (AND condition)
 1. The `tier` label must have a value of either `frontend` or `backend` (`In` operator).
 2. The `restricted` label must not exist (`DoesNotExist` operator).
 3. All expressions must evaluate to true (AND condition).

- **Detailed Explanation: podSelector**

The pod selector filters target Pods using two methods:

1. **matchLabels:** Match labels using key-value pairs (AND condition)
 1. Only selects Pods that have both `app=web-service` and `version=v2` labels.
 2. All specified labels must match exactly.
2. **matchExpressions:** Define complex conditions using advanced expressions (AND condition)
 1. The `tier` label must be one of `backend` , `backend2` , or `backend3` (`In` operator).
 2. The `environment` label must not be `dev` or `test` (`NotIn` operator).
 3. The `ready` label must exist (`Exists` operator).
 4. All expressions must evaluate to true (AND condition).

Test

If you want to test APM auto-injection, apply the following yaml:

```
test-values.yaml
```

```

apiVersion: monitoring.whatap.com/v2alpha1
kind: WhatapAgent
metadata:
  name: default          # Resource name
  namespace: whatap-monitoring # Target namespace for installation
spec:
  features:
    apm:
      instrumentation:
        targets:
          - name: "java-sample"      # APM target name (for identification)
            enabled: "true"          # Whether auto-injection is enabled
            language: "java"         # Language: java, python, nodejs, etc.
        whatapApmVersions:
          java: "latest"             # APM agent version for the specified language
        namespaceSelector:
          matchNames:
            - default                # Target namespace for APM injection
        podSelector:
          matchLabels:
            app: hello-world         # Label of Pod to inject APM
        config:
          mode: default              # 'default' or 'custom'. 'custom' requires configMapRef

```

OpenAgent Configuration

OpenAgent scrapes metrics from Prometheus endpoints and forwards them to the WhaTap server. It is a Go-based agent that collects Prometheus-style metrics and sends them to WhaTap.

Configuration

When installing with standalone YAML

- By default, the Open Agent is created in the whatap-monitoring namespace. You can adjust this if needed.

```
# Whatap Open Agent standalone manifests
# Adjust the namespace as needed. The example uses whatap-monitoring.
---
#apiVersion: v1
#kind: Namespace
#metadata:
# name: whatap-monitoring
---
# Whatap connection information (license/host/port)
# If you already created a Secret with the same name, you can delete this Secret block
# and update the Deployment's secretKeyRef.name to match your existing Secret name.
apiVersion: v1
kind: Secret
metadata:
  name: whatap-open-credentials
  namespace: whatap-monitoring
type: Opaque
stringData:
  WHATAP_LICENSE: "<YOUR_LICENSE>"
  WHATAP_HOST: "<WHATAP_HOST>"
  WHATAP_PORT: "6600"
---
apiVersion: v1
```



```

kind: ServiceAccount
metadata:
  name: whatap-open-agent-sa
  namespace: whatap-monitoring

---

apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  name: whatap-open-agent-role
rules:
  - apiGroups: ["*"]
    resources: ["pods", "services", "endpoints", "endpointslices", "namespaces", "secrets"]
    verbs: ["get", "list", "watch"]
  - nonResourceURLs: ["/metrics"]
    verbs: ["*"]
  - apiGroups: [""]
    resources: ["pods/exec"]
    verbs: ["create"]
  - apiGroups: [""]
    resources: ["configmaps"]
    verbs: ["*"]
  - nonResourceURLs: ["/metrics"]
    verbs: ["*"]

---

apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
  name: whatap-open-agent-role-binding
subjects:
  - kind: ServiceAccount
    name: whatap-open-agent-sa
    namespace: whatap-monitoring
roleRef:
  kind: ClusterRole
  name: whatap-open-agent-role
  apiGroup: rbac.authorization.k8s.io

```



```

---
apiVersion: v1
kind: ConfigMap
metadata:
  name: whatap-open-agent-config
  namespace: whatap-monitoring
data:
  scrape_config.yaml: |
    features:
      openAgent:
        enabled: true
        globalInterval: "60s"
        globalPath: "/metrics"
      targets:
        - targetName: kube-apiserver
          type: ServiceMonitor
          namespaceSelector:
            matchNames:
              - "default" # Change to the namespace where the API server service resides
          selector:
            matchLabels:
              component: apiserver
              provider: kubernetes
      endpoints:
        - port: "https"
          path: "/metrics"
          interval: "30s"
          scheme: "https"
          timeout: "30s"
          tlsConfig:
            insecureSkipVerify: true
      metricRelabelConfigs:
        - source_labels: [_name_]
          regex: "apiserver_request_total"
          action: keep
        - source_labels: [verb]
          target_label: http_verb
          replacement: "${1}"
          action: replace

```



```

    - target_label: metric_src
      replacement: "whatap-open-agent"
      action: replace

---
apiVersion: apps/v1
kind: Deployment
metadata:
  name: whatap-open-agent
  namespace: whatap-monitoring
  labels:
    name: whatap-open-agent
spec:
  replicas: 1
  selector:
    matchLabels:
      name: whatap-open-agent
  template:
    metadata:
      labels:
        name: whatap-open-agent
    spec:
      serviceAccountName: whatap-open-agent-sa
      volumes:
        - name: config-volume
          configMap:
            name: whatap-open-agent-config
        - name: logs-volume
          emptyDir: {}
      containers:
        - name: whatap-open-agent
          image: public.ecr.aws/whatap/open_agent:latest
          imagePullPolicy: Always
          resources:
            requests:
              memory: "300Mi"
              cpu: "100m"
            limits:
              memory: "1000Mi"

```




```

cpu: "1000m"
env:
  - name: WHATAP_LICENSE
    valueFrom:
      secretKeyRef:
        name: whatap-open-credentials
        key: WHATAP_LICENSE
  - name: WHATAP_HOST
    valueFrom:
      secretKeyRef:
        name: whatap-open-credentials
        key: WHATAP_HOST
  - name: WHATAP_PORT
    valueFrom:
      secretKeyRef:
        name: whatap-open-credentials
        key: WHATAP_PORT
  - name: debug
    value: "false"
volumeMounts:
  - name: config-volume
    mountPath: /app/scrape_config.yaml
    subPath: scrape_config.yaml
  - name: logs-volume
    mountPath: /app/logs
# Add container/node tolerations, affinity, and other settings as needed.

```

Helm installation



1. Run the following commands to add the WhaTap Helm repository.

```

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

```

2. For first-time installation, run the following command to install the agent application. Refer to the example [values.yaml](#) below.



- First Install Existing User

#If you are installing the agent for the first time, run the following commands to install the agent application.

```
kubectrl create ns whatap-monitoring
helm install whatap-kube-agent --namespace whatap-monitoring whatap/kube -f values.yaml
```

*#Existing WhaTap Kubernetes Agent users should run the following commands to install the agent.
#If you previously installed the WhaTap Kubernetes Agent using YAML files, a Clean Install is required.
Remove the existing Kubernetes agent, then install it using the Helm chart.*

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

values.yaml

```
whatap:
  license: #Enter your license key
  host: #Enter the collector server IP
  port: #Enter the collector server port
  createSecret: true #default : true, if credentials already exist, set false
  existingSecretName: "whatap-credentials" #default : whatap-credentials
  collect_custom_resources: true
  containerRuntime: "containerd"
  deployment:
    containers:
      controlPlaneHelper:
        enabled: false
        image: "whatap/kube_mon:1.8.8"
        name: "whatap-control-plane-helper"
        port: 9496
        resources:
          limits:
            cpu: "200m"
            memory: "350Mi"
          requests:
```



```

    cpu: "100m"
    memory: "300Mi"
masterAgent:
  image: "whatap/kube_mon:1.8.8"
  name: "whatap-master-agent"
  port: 6600
  resources:
    limits:
      cpu: "200m"
      memory: "350Mi"
    requests:
      cpu: "100m"
      memory: "300Mi"
  label: "whatap-master-agent"
  name: "whatap-master-agent"
  replicas: 1

daemonSet:
  containers:
    nodeAgent:
      image: "whatap/kube_mon:1.8.8"
      limits:
        cpu: "200m"
        memory: "350Mi"
      name: "whatap-node-agent"
      requests:
        cpu: "100m"
        memory: "300Mi"
    nodeHelper:
      image: "whatap/kube_mon:1.8.8"
      limits:
        cpu: "200m"
        memory: "350Mi"
      name: "whatap-node-helper"
      requests:
        cpu: "100m"
        memory: "100Mi"
  initContainers:
    nodeDebugger:

```



```

enabled: false
image: "whatap/kube_mon:1.8.8"
name: "whatap-node-debug"
label: "whatap-node-agent"
name: "whatap-node-agent"

clusterrole:
  extraResources:
    - "*"

deploymentOpen:
  enabled: true
  containers:
    openAgent:
      image: "public.ecr.aws/whatap/open_agent:latest"
  config:
    features:
      openAgent:
        enabled: true
        #globalInterval: "60s"
        #globalPath: "/metrics"
      targets:
        - targetName: "kube-apiserver"
          type: "ServiceMonitor"
          enabled: true
          namespaceSelector:
            matchNames:
              - "default" # Adjust according to the cluster's default labels
          selector:
            matchLabels:
              component: "apiserver"
            provider: "kubernetes"
          endpoints:
            - port: "https"
              path: "/metrics"
              interval: "30s"
              scheme: "https"
              timeout: "30s"
            tlsConfig:

```



```

    insecureSkipVerify: true # For self-signed certificates, this is for convenience; for security, set to false
    and specify a CA
    metricRelabelConfigs:
      - source_labels: [__name__]
        regex: "apiserver_request_total"
        action: keep
      - source_labels: [verb]
        target_label: http_verb
        replacement: "${1}"
        action: replace
      - target_label: metric_src
        replacement: "whatap-open-agent"
        action: replace

```

After installing the WhatapOperator, you can configure open metrics targets using the `WhatapAgent` CR (Custom Resource).

1. **PodMonitor:** Dynamic discovery using Pod label selectors (similar to Prometheus Operator's PodMonitor)
2. **ServiceMonitor:** Dynamic discovery using Service label selectors (similar to Prometheus Operator's ServiceMonitor)
3. **StaticEndpoints:** Directly specifying fixed IP addresses and ports (similar to Prometheus `static_configs`)

```

features:
  openAgent:
    enabled: true
  targets:
    # 1. PodMonitor: Dynamic discovery using Pod label selectors
    - targetName: my-app-pod-metrics
      type: PodMonitor
      namespaceSelector:
        matchNames:
          - "production"
      selector:
        matchLabels:
          app: my-app

```

```

endpoints:
- port: "web-metrics" # Port name defined in the Pod spec or the actual port number
  path: "/metrics" # Default is /metrics; override if needed
  interval: "15s" # Default comes from global settings; override if needed
  scheme: "http"
  timeout: "10s"
  metricRelabelConfigs: # Metric relabeling after scraping
    - source_labels: [__name__]
      regex: "http_requests_total"
      action: keep
    - source_labels: [method]
      target_label: http_method
      replacement: "${1}"
      action: replace

# 2. ServiceMonitor: Dynamic discovery using Service label selectors
- targetName: my-service-metrics
  type: ServiceMonitor
  namespaceSelector:
    matchNames:
      - "default"
  selector:
    matchLabels:
      service: my-backend-service
  endpoints:
- port: "http-metrics" # Port name defined in the Service spec or the actual target port number
  path: "/actuator/prometheus"
  interval: "30s"
  metricRelabelConfigs:
    - source_labels: [__name__]
      regex: "jvm_.*"
      action: keep
    - source_labels: [area]
      target_label: memory_area
      replacement: "${1}"
      action: replace

# 3. StaticEndpoints: Specify fixed IP addresses and ports directly
- targetName: my-external-db-metrics
  type: StaticEndpoints
# enabled: true # Whether the target is active (default: true, optional)

```

```

endpoints:
- address: "192.168.1.100:9100"
  scheme: "http"
  path: "/metrics"
  interval: "60s"
metricRelabelConfigs:
- source_labels: [__name__]
  regex: "node_(cpu|memory).*"
  action: keep
- source_labels: [instance]
  target_label: server
  replacement: "${1}"
  action: replace

```

PodMetrics and ServiceMetrics Configuration Fields

- **targetName:** Name of the target (used for logging and identification)
- **type:** Type of target (`PodMonitor` or `ServiceMonitor`)
- **namespaceSelector:** Specifies which namespaces to scrape
 - `matchNames` : Select namespaces by name
 - `matchLabels` : Select namespaces by label
 - `matchExpressions` : Use expressions to select namespaces
- **selector:** Selects which Pods or Services to scrape
 - `matchLabels` : Select by label
 - `matchExpressions` : Select using expressions
- **relabelConfigs:** Adds or modifies labels based on metadata (such as meta labels) during the target discovery phase.
(Similar to Prometheus `relabel_configs`)
- **endpoints:** Defines endpoints to scrape
 - `port` : Port name or number to scrape
 - `path` : Metrics path (default: `globalPath` , override if needed)
 - `interval` : Scrape interval (default: `globalInterval` , override if needed)

- `scheme` : Protocol used (`http` or `https` , default: `http`)
- `timeout` : Scrape timeout duration
- `honorLabels` : Whether to prioritize labels exposed by the target
- `metricRelabelConfigs` : Post-scrape relabeling (similar to Prometheus `metric_relabel_configs`)

StaticEndpoints Configuration Fields

- **targetName**: Name of the target (used for logging and identification)
- **type**: Target type (`StaticEndpoints`)
- **endpoints**: Defines static endpoints to scrape
 - `address` : Target address to scrape (format: `IP:PORT` or `HOSTNAME:PORT`)
 - `path` : Metrics path (default: `/metrics`)
 - `scheme` : Protocol to use (`http` or `https` , default: `http`)
 - `interval` : Scrape interval (default: `60s`)
 - `tlsConfig` : TLS settings
 - `metricRelabelConfigs` : Post-scrape metric relabeling

addNodeLabel Feature

If the target object is scheduled on a node, you can use the `addNodeLabel` option to add the name of the scheduled node as a label in the metrics.

- **Where to configure**: Can be set at the target level or endpoint level
- **Default**: `false`
- **Behavior**: If set to `true` , a `node` label is added to all metrics, with the name of the node where the object is running

Usage Example

```
# Target-level addNodeLabel setting
- targetName: node-exporter
  type: PodMonitor
```



```

namespaceSelector:
  matchNames:
    - "monitoring"
selector:
  matchLabels:
    app: node-exporter
endpoints:
  - port: "metrics"
    path: "/metrics"
    interval: "30s"
    addNodeLabel: true # Add node label at this endpoint
    metricRelabelConfigs:
      # Example: Relabel using the node label
      - source_labels: [node]
        target_label: kubernetes_node
        action: replace

# Endpoint-level addNodeLabel setting
  - targetName: dcgm-exporter
    type: PodMonitor
    namespaceSelector:
      matchNames:
        - "kube-system"
    selector:
      matchLabels:
        app: dcgm-exporter
    endpoints:
      - port: "metrics"
        path: "/metrics"
        interval: "30s"
        addNodeLabel: true # Add node label only for this endpoint
        metricRelabelConfigs:
          - source_labels: [node]
            target_label: gpu_node
            action: replace

```

Basic Auth Configuration

OpenAgent supports HTTP Basic Authentication to scrape endpoints that require authentication. The username and password are securely managed using Kubernetes Secrets.

- **Configuration location:** Configure under the `basicAuth` field inside `endpoints`.
- **username:** Secret information containing the username
 - `name` : Name of the Secret
 - `key` : Key within the Secret
 - `namespace` : Namespace where the Secret is located (If omitted, the same namespace as the agent is used.)
- **password:** Secret information containing the password
 - `name` : Name of the Secret
 - `key` : Key within the Secret
 - `namespace` : Namespace where the Secret is located

Configuration Example

```
endpoints:
- port: "metrics"
  path: "/metrics"
  basicAuth:
    username:
      name: my-auth-secret
      key: username
    password:
      name: my-auth-secret
      key: password
      namespace: monitoring # Specify if the Secret is in a different namespace
```

TLS Configuration

OpenAgent supports TLS (Transport Layer Security) when connecting to HTTPS endpoints. The following are available options related to TLS settings.

How HTTP vs HTTPS is Determined

OpenAgent determines whether to use HTTP or HTTPS based on the following rules:

- **PodMonitor and ServiceMonitor targets**
 - If the port name is `"https"` , HTTPS is used by default.
 - In all other cases, HTTP is used by default.
- **StaticEndpoints targets**
 - If TLS settings are defined, HTTPS is used by default.
 - Otherwise, HTTP is used.
- **All target types**

TLS Configuration Options

TLS settings can be configured in the `tlsConfig` section.

```
endpoints:  
- port: "https"  
  path: "/metrics"  
  scheme: "https" # Explicitly use HTTPS  
tlsConfig:  
  insecureSkipVerify: true # Skip certificate verification
```

insecureSkipVerify

The `insecureSkipVerify` option allows skipping server certificate verification.
This option is useful in the following cases:

- When connecting to a server with a self-signed certificate
- When certificate verification is not required in development or testing environments
- When connecting to trusted servers within an internal network

Caution

For production environments, it is recommended to set `insecureSkipVerify: false` for security reasons. If you are using a self-signed certificate, it is safer to add it to a trusted Certificate Authority (CA) instead.

Configuration Example

TLS Configuration Example with ServiceMonitor

```
- targetName: kube-apiserver
type: ServiceMonitor
namespaceSelector:
  matchNames:
    - "default"
selector:
  matchLabels:
    component: apiserver
    provider: kubernetes
endpoints:
  - port: "https" # HTTPS is used by default because the port name is "https"
    path: "/metrics"
    interval: "30s"
    scheme: "https" # Explicitly specify HTTPS (optional)
  tlsConfig:
    insecureSkipVerify: true # Skip certificate verification
  metricRelabelConfigs:
    - source_labels: [__name__]
      regex: "apiserver_request_total"
      action: keep
    - source_labels: [verb]
      target_label: http_verb
      replacement: "${1}"
      action: replace
```

Target Relabeling Configuration

`relabelConfigs` is a feature that filters targets or manipulates labels based on the target's own metadata **before** scraping is performed.

(It is similar to Prometheus `relabel_configs`.)

1. Target Filtering

- Excludes targets that do not meet specific metadata conditions from being scraped.

- Common actions: `keep` , `drop`

2. Meta Label Promotion

- Metadata labels starting with `__meta__` are removed after scraping.
- To persist them as permanent labels, use the `replace` action at this stage to rename them.

Meta Labels List

Below is a list of meta labels provided by OpenAgent.

Common

- `__meta_kubernetes_namespace` : Object namespace
- `__address__` : Target address (IP:Port)

PodMonitor

- `__meta_kubernetes_pod_name` : Pod name
- `__meta_kubernetes_pod_ip` : Pod IP
- `__meta_kubernetes_pod_ready` : Ready for Pod (`true` / `false`)
- `__meta_kubernetes_pod_phase` : Pod status (`Running` , `Pending` , etc.)
- `__meta_kubernetes_pod_node_name` : Name of the node running the pod
- `__meta_kubernetes_pod_label_<labelname>` : Pod label
- `__meta_kubernetes_pod_annotation_<annotationname>` : Pod annotation

ServiceMonitor

- `__meta_kubernetes_service_name` : Service name
- `__meta_kubernetes_service_cluster_ip` : Cluster IP
- `__meta_kubernetes_service_label_<labelname>` : Service label

```
targets:
```

```

- targetName: my-pod-monitor
  type: PodMonitor
  selector:
    matchLabels:
      app: my-app
  # Labeling in the target discovery phase
  relabelConfigs:
    # 1. PStore the Pod name as the pod_name label (the metric will include the pod_name label)
    - source_labels: ["__meta_kubernetes_pod_name"]
      target_label: "pod_name"
      action: "replace"

    # 2. Exclude Pods that do not have the specific label (scrape_enabled=true) from scraping
    - source_labels: ["__meta_kubernetes_pod_label_scrape_enabled"]
      regex: "true"
      action: "keep"

```

Metric Relabeling Configuration

OpenAgent supports metric relabeling functionality similar to Prometheus's `metric_relabel_configs`. This feature allows you to filter or modify metrics after they are scraped.

Relabeling Configuration Fields

- **source_labels:** List of source labels (array)
- **separator:** Separator used to join source label values (default: `;`)
- **target_label:** The label to which the result is written
- **regex:** Regular expression applied to the joined source label values
- **replacement:** Replacement value (can reference regex capture groups, e.g., `${1}`)
- **action:** Action to perform (`keep` , `drop` , or `replace`)

Supported Actions

- **keep:** Keep only metrics that match the regex

- **drop:** Drop metrics that match the regex
- **replace:** Replace the value of the target label with the replacement

Special Labels

- **name:** A special label that refers to the metric name

Example

Example 1: Keep only a specific metric

This configuration keeps only the `http_requests_total` metric and drops all others.

```
metricRelabelConfigs:
- source_labels: [__name__]
  regex: "http_requests_total"
  action: keep
```

- **Example Behavior**

Suppose the following metrics were originally scraped:

```
http_requests_total{method="GET", status="200"} 100
http_errors_total{method="GET", status="500"} 5
node_cpu_seconds_total{cpu="0", mode="idle"} 1000
```

After applying the above `metricRelabelConfigs`, only the `http_requests_total` metric will be kept. All other metrics will be dropped:

```
http_requests_total{method="GET", status="200"} 100
```

Example 2: Filter Metrics Using Regex

This configuration keeps only metrics whose names start with `node_cpu` or `node_memory`.

```
metricRelabelConfigs:
```

```
- source_labels: [__name__]
  regex: "node_(cpu|memory).*"
  action: keep
```

- **Example Behavior**

Suppose the following metrics were originally scraped:

```
node_cpu_seconds_total{cpu="0", mode="idle"} 1000
node_memory_MemTotal_bytes{} 16777216
node_disk_io_time_seconds_total{device="sda"} 100
http_requests_total{method="GET", status="200"} 100
```

After applying the `metricRelabelConfigs`, only the metrics that start with `node_cpu` or `node_memory` will be kept.

Using regular expressions allows you to filter multiple metric patterns at once.

```
node_cpu_seconds_total{cpu="0", mode="idle"} 1000
node_memory_MemTotal_bytes{} 16777216
```

Example 3: Rename Label

This configuration copies the value of the `method` label into a new label called `http_method`.

```
metricRelabelConfigs:
- source_labels: [method]
  target_label: http_method
  replacement: "${1}"
  action: replace
```

- **Example Behavior**

Suppose the following metrics were originally scraped:

```
http_requests_total{method="GET", path="/api", status="200"} 100
http_requests_total{method="POST", path="/api/users", status="201"} 50
```

After applying the above `metricRelabelConfigs`, each metric will include a new label `http_method` that duplicates the value of the existing `method` label.

The original label remains unchanged, and the new label is added.

`${1}` refers to the value of the source label.

```
http_requests_total{method="GET", path="/api", status="200", http_method="GET"} 100
http_requests_total{method="POST", path="/api/users", status="201", http_method="POST"} 50
```

Example 4: Combine Multiple Source Labels

```
metricRelabelConfigs:
- source_labels: [__name__, status]
  regex: "http_requests_total;(200|500)"
  action: keep
```

This configuration keeps only the `http_requests_total` metrics where the `status` label is either `200` or `500`.

• Example Behavior

Suppose the following metrics were originally scraped:

```
http_requests_total{method="GET", path="/api", status="200"} 100
http_requests_total{method="POST", path="/api/users", status="201"} 50
http_requests_total{method="GET", path="/api/error", status="500"} 10
http_requests_total{method="GET", path="/api/error", status="404"} 5
```

After applying the `metricRelabelConfigs`, only the `http_requests_total` metrics with a `status` label of `200` or `500` will be retained.

When combining multiple source labels, they are joined using the default separator `;`, which can be changed using the `separator` field.

```
http_requests_total{method="GET", path="/api", status="200"} 100
http_requests_total{method="GET", path="/api/error", status="500"} 10
```

Example 5: Add a Static Label

```
metricRelabelConfigs:
- target_label: metric_src
  replacement: "whatap-open-agent"
```

```
action: replace
```

This configuration adds a static label `metric_src="whatap-open-agent"` to all metrics.

If no `source_labels` are specified, the `replacement` value is used directly as the label value.

This method is useful for adding static labels such as environment, region, or application name to all metrics.

• Example Behavior

Suppose the following metrics were originally scraped:

```
http_requests_total{method="GET", path="/api", status="200"} 100
node_cpu_seconds_total{cpu="0", mode="idle"} 1000
```

After applying the `metricRelabelConfigs`, all metrics will include the label `metric_src="whatap-open-agent"`. This is useful for marking the origin of the metric or tagging it with information such as environment (`production` , `staging`), region (`us-east` , `eu-west`), or application name.

```
http_requests_total{method="GET", path="/api", status="200", metric_src="whatap-open-agent"} 100
node_cpu_seconds_total{cpu="0", mode="idle", metric_src="whatap-open-agent"} 1000
```

Example: Combined Behavior

Suppose the following metrics were originally scraped:

```
apiserver_request_total{code="200", resource="pods", verb="GET"} 100
some_other_metric{label="value"} 50
```

After applying the above `metricRelabelConfigs` :

1. First rule (`keep apiserver_request_total`) is applied:
 - The `apiserver_request_total` metric is retained.
 - The `some_other_metric` is dropped.
2. Second rule (`replace verb -> http_verb`) is applied:
 - Since the retained `apiserver_request_total` metric contains the `verb` label, its value (`GET`) is copied into a new label named `http_verb` .

As a result, the final metric collected will look like this:

```
apiserver_request_total{code="200", resource="pods", verb="GET", http_verb="GET"} 100
```

Kubernetes Metrics Collection Example

The following is an example of collecting metrics from the Kubernetes API server:

- This configuration selects a Service in the `kube-system` namespace that has the labels `component=apiserver` and `provider=kubernetes`, and scrapes metrics from its endpoint.
- The `metricRelabelConfigs` are used to:
 - collect only the `apiserver_request_total` metric,
 - transform the `verb` label into a new `http_verb` label,
 - and add a static label `metric_src="whatap-open-agent"` to all metrics.

Kubernetes Metrics Collection Example

```
# scrape_config.yaml
features:
  openAgent:
    enabled: true
  targets:
    - targetName: kube-apiserver
      type: ServiceMonitor
      namespaceSelector:
        matchNames:
          - "kube-system"
      selector:
        matchLabels:
          component: apiserver
          provider: kubernetes
      endpoints:
        - port: "https"
          path: "/metrics"
          interval: "30s"
      metricRelabelConfigs:
```

```
- source_labels: [_name_]
  regex: "apiserver_request_total"
  action: keep
- source_labels: [verb]
  target_label: http_verb
  replacement: "${1}"
  action: replace
# Add Static Labels
- target_label: metric_src
  replacement: "whatap-open-agent"
  action: replace
```

WhaTap CR Configuration Options

WhaTap CR configuration options provide ways to control agent behavior by precisely specifying image, environment variables, resources, scheduling, authentication, and priority.

Image configuration

```
apiVersion: monitoring.whatap.com/v2alpha1
kind: WhatapAgent
metadata:
  name: whatap
spec:
  features:
    k8sAgent:
      # Full image name
      customImageFullName: public.ecr.aws/whatap/kube_agent:1.8.8
    masterAgent:
      enabled: true
      masterAgentContainer:
        # Overridable per container
        customImageFullName: public.ecr.aws/whatap/kube_agent:1.8.8
    nodeAgent:
      enabled: true
      nodeAgentContainer:
        # Overridable per container
        customImageFullName: public.ecr.aws/whatap/kube_agent:1.8.8
      nodeHelperContainer:
        # Overridable per container
        customImageFullName: public.ecr.aws/whatap/kube_agent:1.8.8
    openAgent:
      enabled: true
      # Overridable per container
      customImageFullName: public.ecr.aws/whatap/open_agent:latest
```

1. Set options via container env

The Kubernetes agent can set options through container env variables.

- The operator automatically adds default env variables (WHATAP_LICENSE/HOST/PORT, WHATAP_MEM_LIMIT, and so on). Values you put under `nodeAgent.nodeAgentContainer.envs` are merged after the defaults.

```
# When configuring the whatap-node-agent container options in the WhaTap node agent DaemonSet
apiVersion: monitoring.whatap.com/v2alpha1
kind: WhatapAgent
metadata:
  name: whatap
spec:
  features:
    k8sAgent:
      masterAgent:
        enabled: true
      nodeAgent:
        enabled: true
      nodeAgentContainer:
        envs:
          - name: debug
            value: "true"
    gpuMonitoring:
      enabled: false
```

2. Configure resources

The operator includes default values for k8s agent installation.

```
# When configuring the whatap-node-agent container options in the WhaTap node agent DaemonSet
apiVersion: monitoring.whatap.com/v2alpha1
kind: WhatapAgent
metadata:
  name: whatap
spec:
  features:
    k8sAgent:
      masterAgent:
        enabled: true
      # Master agent container resources (container-level override)
      masterAgentContainer:
        resources:
```

```

requests:
  cpu: "150m"
  memory: "256Mi"
limits:
  cpu: "300m"
  memory: "512Mi"
nodeAgent:
  enabled: true
# Node agent component-level default resources
resources:
  requests:
    cpu: "100m"
    memory: "300Mi"
  limits:
    cpu: "200m"
    memory: "350Mi"
# Node agent container override
nodeAgentContainer:
  resources:
    requests:
      cpu: "150m"
      memory: "350Mi"
    limits:
      cpu: "300m"
      memory: "512Mi"
# Helper container override
nodeHelperContainer:
  resources:
    requests:
      cpu: "100m"
      memory: "100Mi"
    limits:
      cpu: "200m"
      memory: "350Mi"
gpuMonitoring:
  enabled: false

```

- If a container defines `resources` , those values override the component defaults for that container.

Container	requests	limits
whatap-master-agent	cpu: 100m memory: 300Mi	cpu: 200m memory: 350Mi
whatap-node-agent	request: 100m limit: 300Mi	request: 200Mi limit: 350Mi
whatap-node-helper	request: 100m limit: 100m	request:200Mi limit: 350Mi

3. tolerations, affinity, nodeSelector

tolerations

```
spec:
  features:
    k8sAgent:
      masterAgent:
        enabled: true
        tolerations:
          - key: "node.kubernetes.io/not-ready"
            operator: "Exists"
            effect: "NoExecute"
            tolerationSeconds: 300
      nodeAgent:
        enabled: true
        tolerations:
          - key: "dedicated"
            operator: "Equal"
            value: "observability"
            effect: "NoSchedule"
```

affinity

The master and node agents have the following default tolerations built in; values set in the CR are merged after these defaults.

- `node-role.kubernetes.io/master:NoSchedule`
- `node-role.kubernetes.io/control-plane:NoSchedule`

```
spec:
  features:
    k8sAgent:
      masterAgent:
        enabled: true
        affinity:
          nodeAffinity:
            requiredDuringSchedulingIgnoredDuringExecution:
              nodeSelectorTerms:
                - matchExpressions:
                    - key: kubernetes.io/os
                      operator: In
                      values: ["linux"]
      nodeAgent:
        enabled: true
        affinity:
          podAntiAffinity:
            preferredDuringSchedulingIgnoredDuringExecution:
              - weight: 100
            podAffinityTerm:
              topologyKey: kubernetes.io/hostname
              labelSelector:
                matchLabels:
                  name: whatap-node-agent
```

nodeSelector

- Apply nodeSelector to the master agent

```
spec:
  features:
    k8sAgent:
      masterAgent:
        enabled: true
        nodeSelector:
          nodepool: system      # Example: pin to the system node pool
```

```
kubernetes.io/os: linux    # Linux nodes only
```

- Apply `nodeSelector` to the node agent (DaemonSet)

```
spec:
  features:
    k8sAgent:
      nodeAgent:
        enabled: true
        nodeSelector:
          nodepool: worker    # Example: worker nodes only
          kubernetes.io/arch: amd64 # AMD64 architecture nodes only
```

ⓘ The node agent is a DaemonSet deployed to all nodes. If you set `nodeSelector`, Pods are created only on nodes that satisfy the conditions.

- In a GPU cluster, to deploy only to GPU nodes, use the following:

```
spec:
  features:
    k8sAgent:
      nodeAgent:
        enabled: true
        nodeSelector:
          nvidia.com/gpu.present: "true"
      gpuMonitoring:
        enabled: true
```

- Apply `nodeSelector` to OpenAgent

```
spec:
  features:
    openAgent:
      enabled: true
      nodeSelector:
        nodepool: system    # Pin OpenAgent to the system node pool
      kubernetes.io/os: linux
```

```
# Optional: combine with affinity for finer scheduling
affinity:
  nodeAffinity:
    requiredDuringSchedulingIgnoredDuringExecution:
      nodeSelectorTerms:
        - matchExpressions:
            - key: kubernetes.io/hostname
              operator: In
              values: ["ip-10-0-0-12"]
```

4. imagePullSecret

```
spec:
  features:
    k8sAgent:
      # Global pull secret (default for both master and node)
    imagePullSecrets:
      - name: global-regcred
    masterAgent:
      enabled: true
      # Master-only override
    imagePullSecrets:
      - name: master-regcred
    nodeAgent:
      enabled: true
      # Node-only override
    imagePullSecrets:
      - name: node-regcred
    openAgent:
      enabled: true
    imagePullSecrets:
      - name: openagent-regcred
```

5. PriorityClassName

```
spec:
  features:
```

```
k8sAgent:  
  masterAgent:  
    enabled: true  
    priorityClassName: system-cluster-critical  
  nodeAgent:  
    enabled: true  
    priorityClassName: system-node-critical  
openAgent:  
  enabled: true  
  priorityClassName: system-cluster-critical
```

Installation Of The Master And Node Agents

The following guides you to the master and node agent installation procedure as follows: This procedure is the same as in **Install guide**.

ⓘ EKS Fargate is to be supported later.

Selecting an installation method

Check the access key and then select the [Choose installation method](#) tab on the [Setup guide](#) screen. Select a desired installation method out of the **Yaml** and **Helm** chart installation methods supported by the WhaTap Kubernetes agent, and then select an appropriate item according to your Kubernetes environment.

CP Kubernetes Monitoring Demo(EKS) ▾ Agent Installation

Precautions before installation

The Kubernetes agent is installed in the **whataap-monitoring namespace**. The namespace in which the agent is installed cannot be changed. The **6606 port** must be open to transfer data to **WhaTap Server 127.0.0.1**. (TCP Outbound)

Installation guide Chat contact

- 1 Check access key
- 2 **Choose installation method**
- 3 Apply additional features
- 4 Download installation file
- 5 Enter execution command

Additional installation

- 1 Additional installation of application monitoring

Choose installation method

Select the WhaTap Kubernetes agent image version.*

[Stable \(recommended\)](#) [Preview](#)

*Stable image version: whataap/kube_mon

Choose an installation method.*

[YAML](#) [Helm](#)

Choose a Kubernetes version.*

☒ 1.16 or more ☐ below 1.16

Select a container runtime.*

☒ containerd ☐ Docker Engine ☐ CRI-O

Is it GKE?

☐ Yes ☒ No

Is it OpenShift?

☐ Yes ☒ No

Is it Istio?

☐ Yes ☒ No

Is it a GPU Cluster?

☐ Yes ☒ No

[< Previous](#) [Next >](#)

1. Select any of [Stable](#) and [Preview](#) versions of the WhaTap Kubernetes agent image that are recommended.
2. Select a desired **installation method** out of [yaml](#) and [helm](#) chart.
3. Select an applicable **Kubernetes version**: [1.16 or later](#) or [less than 1.16](#).
4. Select **Container runtime** from [containerd](#), [Docker Engine](#), and [CRI-O](#).
5. Select a **User Kubernetes environment** (GKE, OpenShift, Istio, GPU Cluster), if applicable.
6. After selecting an installation method, click [Next](#) at the lower right of the screen or select the [Apply additional features](#) tab on the left to move to the next step.

Added support for WhaTap Kubernetes agent's image repository.

WhaTap Kubernetes agent added support for Public ECR in addition to the existing DockerHub. Depending on the operation environment, you can select one of the two image repositories for installation.

Public ECR

- Repository: `public.ecr.aws/whatap/kube_mon` **recommended**

Because ECR is not affected by the pull limit policy, it is recommended for stable use in environments with frequent updates or large-scale deployments.

DockerHub

- Repository: `whatap/kube_mon`

DockerHub may enforce pull restrictions (policy) for unauthenticated clients. If you have a large number of nodes or concurrent pull requests, there is a high possibility that the installation fails due to rate limiting. For more information, see the official [DockerHub policy guide](#) document.

 If you encounter a pull limit, see [the following](#).

Additional features

WhaTap Kubernetes Monitoring provides various features in addition.


Precautions before installation

The Kubernetes agent is installed in the **whatap-monitoring namespace**. The namespace in which the agent is installed cannot be changed.

The **6600 port** must be open to transfer data to **WhaTap Server 52.68.36.166/52.193.60.176**. (TCP Outbound)

Installation guide
[Chat contact](#)

- 1 Check access key
- 2 Choose installation method
- 3 **Apply additional features**
- 4 Download installation file
- 5 Enter execution command

Additional installation

- 1 Additional installation of application monitoring

Apply additional features

Please select if there are additional features you would like to apply. As more features are added, the amount of data collected and agent usage can increase. Even after installation, you can enable/disable the feature through agent settings.

☐ **kube-apiserver**

You can easily monitor the availability and performance of the cluster by collecting kube-apiserver's health and performance metrics through the kube-apiserver dashboard menu, tracking kube-apiserver's response time, throughput, request failures, etc.

☐ **Assign access to all CustomResource**

If you select Assign access to all CustomResources, the whatap service accounts acquire the get, list, and watch roles for all cluster resources.

The following features require user settings to activate. For more information, see the [WhaTap Docs](#).

etcd [View in WhaTap Docs](#)

You can use the etcd dashboard to monitor task processing status and collected performance metrics, through which you can trace etcd's task delays, leader election status, load, and the like to identify potential problems occurring in the cluster.

kube-scheduler [View in WhaTap Docs](#)

The kube-scheduler dashboard menu allows you to identify any propagation of Pod placement issue early as possible across the cluster through the kube-scheduler performance, Pod preemption attempts and eviction trends, and Pending trends.

[< Previous](#)
[Next >](#)

The more additional features are applied, the greater the amount of data collected and the resources used by the agent may increase. After installation, you can enable or disable the feature through the agent configuration.

After selecting whether or not to apply additional features, click [Next](#) at the lower right of the screen, or select the [Download installation file](#) tab on the left to move to the next step.

Control Plane monitoring activation

For more information about **control plane monitoring** activation and individual features, see the following.

- **kube-apiserver Dashboard** [Shortcut](#)
- **kube-apiserver Metrics Search** [Shortcut](#)
- **etcd Dashboard** [Shortcut](#)
- **kube-scheduler Dashboard** [Shortcut](#)

Custom Resource monitoring activation

- [Yaml](#) Helm Chart

In the WhaTap Kubernetes agent installation [yaml](#) file, add `customResource` to use in the `resources` under `rules` of the `ClusterRole` resource.

```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  creationTimestamp: "2025-01-15T01:48:26Z"
  name: whatap
  resourceVersion: "125200773"
  uid: 54b0bcdb-1f12-4779-b0cb-e8de2f6be64a
rules:
- apiGroups:
  - '*'
  resources:
  - namespaces
  - pods
  - pods/log
  - configmaps
  - services
  - endpoints
  - daemonsets
  - ingresses
  - deployments
  - nodes
  - persistentvolumes
  - persistentvolumeclaims
  - events
  - replicaset
  - roles
  - rolebindings
  - clusterroles
  - clusterrolebindings
  - jobs
  - cronjobs
  - statefulsets
  - serviceaccounts
```

```

- configmaps
- storageclasses
- horizontalpodautoscalers
- replicationcontrollers
- customresourcedefinitions # Addition of the CRD collection role
- samplecrds # Addition of the name of custom resource to monitor
verbs:
- get
- list
- watch
- apiGroups:
  - ""
resources:
- pods/exec
verbs:
- create
- apiGroups:
  - ""
resources:
- configmaps
verbs:
- '*'
- nonResourceURLs:
  - /metrics
verbs:
  - '*'

```

When installing the WhTtap Kubernetes agent by using the Helm chart, add the custom resource names to be monitored in the default `values.yaml` file. Add the custom resource name in `resources` as follows:

```

collect_custom_resource: true
clusterrole:
  extraResources:
    - "customresourcedefinitions" ##Addition of the CRD collection role
    - "samplecrds" ##Addition of the custom resource to collect

```

! The **Object Manifest** menu can perform custom resource monitoring.

ClusterRole permissions for newly collected resources

The following collection features work only when view permissions are added to the `resources` item under `rules` of the `ClusterRole` resource.

Resource	Feature	Description
<code>networkpolicies</code>	NetworkPolicy event collection	Required to collect the <code>kube_network_policy</code> category
<code>resourcequotas</code>	Namespace ResourceQuota collection	Required to collect the <code>quota_*</code> fields of the <code>kube_stat</code> category

- Without the permission, only that collection is skipped and the rest of the collection continues.

Downloading the installation file

Before downloading the installation file, make sure that the appropriate items for the Kubernetes environment are selected. After confirming your selection, click [Download installation file](#) to download the default configuration file (`values.yaml`) required for installation.

CP

Kubernetes Monitoring Demo(EKS) ▾ Agent Installation

Precautions before installation

The Kubernetes agent is installed in the **whatap-monitoring namespace**. The namespace in which the agent is installed cannot be changed. The **6606 port** must be open to transfer data to **WhaTap Server 127.0.0.1**. (TCP Outbound)

Installation guide

- Check access key
- Choose installation method
- Apply additional features
- Download installation file**
- Enter execution command

Additional installation

- Additional installation of application monitoring

Download installation file

Download the installation file configured based on the following information: Or back to previous steps.

Installation method

Helm

Kubernetes version

1.16 or more

Container Runtime

containerd

Clicking the Download Installation Files button will download a total of 1 files and folders:

- values.yaml

Download Installation File

< Previous

Next >

Choose an installation method.*

Helm

YAML

*Helm 3.0 version or higher is supported.

Choose a Kubernetes version.*

1.16 or more

below 1.16

Select a container runtime.*

containerd

Docker Engine

CRI-O

Is it GKE?

Yes

No

Is it OpenShift?

Yes

No

Is it Istio?

Yes

No

Is it a GPU Cluster?

Yes

No

After downloading, click [Next](#) at the lower right of the screen or select the [Enter execution command](#) tab to move to the next step.

Enter execution command

In the [Choose installation method](#) tab, proceed with the installation by entering the execution command according to the selected environment and installation instructions.

88

**Precautions before installation**

The Kubernetes agent is installed in the **whatap-monitoring namespace**. The namespace in which the agent is installed cannot be changed. The **6606 port** must be open to transfer data to **WhaTap Server 127.0.0.1**. (TCP Outbound)

Installation guide[Chat contact](#)

- 1 Check access key
- 2 Choose installation method
- 3 Apply additional features
- 4 Download installation file
- 5 Enter execution command**

Additional installation

- 1 Additional installation of application monitoring**

Enter execution command

Execute the following command to add the WhaTap Helm repository.

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

[Copy](#)

If you are installing the agent for the first time, run the following command to install the agent application.

```
helm install whatap-kube-agent whatap/kube -f values.yaml
```

[Copy](#)

If you are a WhaTap Kubernetes user, run the following command to install the agent.

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

[Copy](#)

*If you installed the WhaTap Kubernetes agent using the traditional yaml file method, a clean install is required. You can delete the existing Kubernetes agent and install via Helm chart.

If the Kubernetes agent installation is completed successfully, you can check the status of the master agent and node agent by running the following command.

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

[Copy](#)

If the **STATUS** item is **Running**, as shown below, installation is complete.

```
[root@k8s-master ~]# kubectl get pod -n whatap-monitoring
NAME                                READY   STATUS    RESTARTS   AGE
whatap-master-agent-b4d7cbcc8-nblzt 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@k8s-master ~]#
```

[< Previous](#)[Install additional monitoring >](#)[Monitoring start guide 📄](#)

Kubernetes environment

If you select the **GKE** or **OpenShift** environment in the [Choose installation method](#) tab, see the following instructions. If not applicable, skip to the next step [Create Kubernetes agent](#).

- [GKE Environment](#) **OpenShift Environment**

If you select the **Google Kubernetes Engine (GKE)** environment in the [Choose installation method](#) tab, execute the following command to set the roles after creating the user cluster.

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

❗ For more information about installation with the gcloud commands, see [Google Cloud Document](#)

If you select the **OpenShift** environment in the [Choose installation method](#) tab, execute the following command to set the roles.

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

Creating the Kubernetes agent

When installing the WhatTap Kubernetes agent, the installation instructions are provided based on the **Helm** chart and **Yaml** file default installation instructions as follows.

- [Yaml](#) **Helm Chart**

Yaml file-based installation

If you select the [yaml](#) file installation method in the [Choose installation method](#) tab, execute the following command to create the agent.

```
kubectl apply -f whatap_kube_agent.yaml
```

- ⓘ The **amount of resources used by the Master Agent** is somewhat **proportional** to the **number of nodes in your environment**. In other words, if there are a large number of nodes in the cluster, OOM (Out Of Memory) may occur when resources exceed the default memory limit (350 MB). The memory limit allocated to the master agent can be changed through the agent installed [yaml](#) file.

Installing the Helm chart default

- ⓘ The WhaTap Kubernetes agent uses the `whatap-monitoring` namespace by default. If the namespace is unavailable or you need a specific namespace, see [the following](#).

If you select the [helm](#) chart installation method in the [Choose installation method](#) tab, see the following instructions.

1. Check the [helm](#) version by using the following command. It supports the version of 3.x or later.

```
helm Version

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

2. Check the network connectivity to the WhaTap Helm repository.

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

# Output example
# pong
```

3. Execute the following command to add the WhaTap Helm repository.

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

4. Install the agent application as follows.
 - For users who install the agent for the **first time**, run the following command to install the agent application.

```
helm install whatap-kube-agent whatap/kube -f values.yaml
```

- In case of **existing** WhaTap Kubernetes agent users, run the following command to install the agent application.

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

⚠ If you have installed the WhaTap Kubernetes agent by using the existing [yaml](#) file, Clean Install is required. After deleting the existing Kubernetes agent, installation is possible using the [helm](#) chart.

Major configuration items for [values.yaml](#)

To modify the WhaTap Kubernetes agent settings, modify the default configuration file (`_ values.yaml`). The major settings in the [values.yaml](#) file are as follows:

Key	Type	Default	Description
<code>whatap.license</code>	String	<code># <license-key></code>	Whatap license key
<code>whatap.host</code>	String	<code># <whatap-server-host></code>	Host address of the Whatap server
<code>whatap.port</code>	Int	<code># <whatap-server-port></code>	Whatap port number
<code>containerRuntime</code>	String	<code>"docker"</code>	Container runtime in use Select any of <code>"docker"</code> , <code>"containerd"</code> , and <code>"crio"</code> .

Key	Type	Default	Description
<code>daemonSet.name</code>	String	<code>whatap-node-agent</code>	Name of DaemonSet
<code>daemonSet.label</code>	String	<code>whatap-node-agent</code>	Label to assign to DaemonSet
<code>daemonSet.initContainers.nodeDebugger.enabled</code>	Bool	<code>true</code>	Whether to enable the Whatap node debugger container
<code>daemonSet.containers.nodeHelper.image</code>	String	<code>whatap/kube_mon</code>	Image of the nodeHelper container
<code>daemonSet.containers.nodeHelper.requests.memory</code>	String	<code>100Mi</code>	nodeHelper MEMORY request
<code>daemonSet.containers.nodeHelper.requests.cpu</code>	String	<code>100m</code>	nodeHelper CPU request
<code>daemonSet.containers.nodeHelper.limits.memory</code>	String	<code>350Mi</code>	nodeHelper MEMORY limit
<code>daemonSet.containers.nodeHelper.limits.cpu</code>	String	<code>200m</code>	nodeHelper CPU limit
<code>daemonSet.containers.nodeAgent.image</code>	String	<code>whatap/kube_mon</code>	Image of the nodeAgent container
<code>daemonSet.containers.nodeAgent.requests.memory</code>	String	<code>300Mi</code>	nodeAgent MEMORY request
<code>daemonSet.containers.nodeAgent.requests.cpu</code>	String	<code>100m</code>	nodeAgent CPU request

Key	Type	Default	Description
<code>daemonSet.containers.nodeAgent.limits.memory</code>	String	<code>350Mi</code>	nodeAgent MEMORY limit
<code>daemonSet.containers.nodeAgent.limits.cpu</code>	String	<code>200m</code>	nodeAgent CPU limit
<code>deployment.name</code>	String	<code>whatap-master-agent</code>	Name of Deployment
<code>deployment.label</code>	String	<code>whatap-master-agent</code>	Label to assign to Deployment
<code>deployment.replicas</code>	Int	<code>1</code>	Number of replicas in Deployment
<code>deployment.containers.controlPlaneHelper.enabled</code>	Boolean	<code>true</code>	Whether to enable the whatap-control-plane-helper container
<code>deployment.containers.controlPlaneHelper.image</code>	String	<code>"whatap/kube_mon"</code>	Image of the controlPlaneHelper container
<code>deployment.containers.controlPlaneHelper.port</code>	String	<code>9496</code>	Port number of the controlPlaneHelper container.
<code>deployment.containers.controlPlaneHelper.resources.requests.memory</code>	String	<code>"300Mi"</code>	MEMORY request of the controlPlaneHelper container.
<code>deployment.containers.controlPlaneHelper.resources.requests.cpu</code>	String	<code>"100m"</code>	CPU request of the controlPlaneHelper container

Key	Type	Default	Description
<code>deployment.containers.controlPlaneHelper.resources.limits.memory</code>	String	"350Mi"	MEMORY limit of the controlPlaneHelper container.
<code>deployment.containers.controlPlaneHelper.resources.limits.cpu</code>	String	"200m"	CPU limit of the controlPlaneHelper container.
<code>deployment.containers.masterAgent.name</code>	String	whatap-master-agent	Name of the masterAgent container.
<code>deployment.containers.masterAgent.image</code>	String	"whatap/kube_mon"	Image of the masterAgent container
<code>deployment.containers.masterAgent.port</code>	Int	6600	Port number of the masterAgent container.
<code>deployment.containers.masterAgent.resources.requests.memory</code>	String	"300Mi"	MEMORY request of the masterAgent container
<code>deployment.containers.masterAgent.resources.requests.cpu</code>	String	"100m"	CPU request of the masterAgent container
<code>deployment.containers.masterAgent.resources.limits.memory</code>	String	"350Mi"	MEMORY limit of the masterAgent container
<code>deployment.containers.masterAgent.resources.limits.cpu</code>	String	"200m"	CPU request of the masterAgent

Key	Type	Default	Description
			container

! For the guide to **Installation troubleshooting** based on the Helm chart, see [the following](#).

Checking the agent installation

If the Kubernetes agent installation has been completed, you can check the status of the master agent and node agent by executing the following command:

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

In case the status is Running

If **STATUS** is in **Running**, it indicates the installation has been finished as follows. After installing the master and node agents, start monitoring the resources in your Kubernetes environment. Go to [Whatap Monitoring Service](#) and then check the performance data collection in [Project List](#) and [Dashboard](#) > [Container Map](#).

```
[root@kube-master ~]# kubectl get pod -n whatap-monitoring
NAME                                READY   STATUS    RESTARTS   AGE
whatap-master-agent-b4d7cbcc8-wblzt 1/1     Running   0           5m5s
whatap-node-agent-qw9dl             2/2     Running   0           5m5s
whatap-node-agent-rgjk8             2/2     Running   0           5m5s
whatap-node-agent-xvwt2             2/2     Running   0           5m5s
[root@kube-master ~]#
```

In case the status is not Running

If the **STATUS** is not **Running**, it indicates that the installation has not been completed successfully as follows.

```
[root@k8s-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-6zcln	2/2	Running	0	31s
whatap-node-agent-dsfxf	2/2	Running	0	31s
whatap-node-agent-n2vnq	2/2	Running	0	31s

```
[root@k8s-master ~]#
```

1. Execute the following command on the master node to determine the cause of the Pod creation failure.

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

2. Check the **Events:** item at the bottom of the execution result.

```
memory: 400Mi
Environment:
  WHATAP_LICENSE: x4l5g23t0qt5a-xecoppt4t37o6-xajnmkvjtjama
  WHATAP_HOST: 13.124.11.223/13.209.172.35
  WHATAP_PORT: 6600
Mounts:
  /bin/entrypoint.sh from start-script-volume (ro,path="/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@k8s-master ~]# kubectl describe pod whatap-master-agent-b4d7cbcc8-q79bw -n whatap-monitoring
```

- ✓ When using the `kubectl` command, the `-n {namespace}` option performs the task for specific namespaces. Because the WhatTap agent is installed in the namespace named `whatap-monitoring` by default, add the `-n whatap-monitoring` option.

- ⓘ • When you select the [Monitoring start guide](#) button at the lower right of the screen, you can see the instructions for main menus of the Whatap Kubernetes monitoring. If you click the button at the bottom of each main menu guide, the corresponding menu appears.



- To proceed with the application agent installation, select [Install additional monitoring](#) at the lower right of the screen. For more information, see [the following](#).
- To update the agent to the latest version, see [the following](#).

Application Agent Installation

Agent installation steps to monitor applications such as Java, Python, and Node.js in containers.

The installation of micro-application agent is the same as the procedure of Application Monitoring Agent Installation. The difference is that the containerization task has been added.

To proceed with installation of the application agent, in the list on the left of [Management](#) > [Agent Installation](#), click the [Additional installation of application monitoring](#) tab under [Additional installation](#), and then click [Additional installation of application monitoring](#) at the bottom.

**Precautions before installation**

The Kubernetes agent is installed in the **whatap-monitoring namespace**. The namespace in which the agent is installed cannot be changed. The **6606 port** must be open to transfer data to **WhaTap Server 127.0.0.1**. (TCP Outbound)

Installation guide

Chat contact

1 Check access key

2 Choose installation method

3 Apply additional features

4 Download installation file

5 Enter execution command

Additional installation

1 Additional installation of application monitoring

Additional installation of application monitoring

If you install WhaTap application monitoring, integrated analysis of containers and applications is possible.

You can track the three-level correlation of container level, service level, and transaction level in the MSA environment, and visually check the call relationships between pods, containers, and services. Additionally, by integrating metrics-trace-log information for observability, quick cause analysis and action are possible when a problem occurs.

*Supported products: Java, PHP, Node.js, Python, Go

Additional installation of application monitoring >

Main feature guide 

< Previous

Checking the access key

Installation guide

▼ Unfold

Chat contact

Docker Java

Docker Node.js

Docker Python

Docker PHP

Docker Go

▼ 1. Checking the access key

The access key is the unique ID to enable the WhaTap service.

If the access key is no longer valid, use of the service is restricted and the service can be renewed or extended through  the chat contact at the upper right of the screen.

Be careful not to expose the access key to outside.

 Copy

▶ 2. Download agent

▶ 3. Configuring the container environment variables

▶ 4. Checking the agent installation

In the [Check Access Key](#) section of the installation guide page, select [Getting the access key](#). If an access key has been already issued, the key appears.

⚠ The **access key** cannot be duplicated for other projects, and it is different from the **cluster project** authentication key.

Containerizing the application agent

See the following to proceed with application agent containerization.

 Installing Docker Java

This step allows you to install the agent to monitor Java applications in the container.

 Installing Docker Node.js

This step allows you to install the agent to monitor Node.js applications in the container.

 Installing Docker Python

This step allows you to install the agent to monitor Python applications in the container.

 Installing the docker PHP

This step allows you to install the agent to monitor PHP applications in the container.

 Installing the docker Go

This step allows you to install the agent to monitor Go applications in the container.

Installing Docker Java

At the bottom of [Management](#) > [Agent installation](#), click [Install Application](#) > Select the [Docker Java](#) tab in Install guide.

This process adds the settings for applying the agent to the JVM option of the Java application running on the Docker container and packages the container images as follows:

- ⓘ • EKS Fargate is to be supported later.
- Git example code is provided to help you understand the Java application integration process. See [the following](#).

Download agent

- [Docker Install](#) Direct Installation

To install the WhaTap application agent, create the [whatap.conf](#) file. Change `{YOUR_PROJECT_ROOT}` to the top-level path of the project to install the agent.

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

Setting the Kubernetes internal application agent options

The settings required for the Kubernetes internal application agent can be configured in the [whatap.conf](#) file or through the container `env` field when releasing the agent.

- The options in the [whatap.conf](#) file have higher priority than those set in the container `env` field.
- If no value is set in [whatap.conf](#) or container `env`, the default value is used.
- Use the In case of 'license' and 'whatap.server.host' options, use the **container environment variables** settings. For more information, check [\[the following\]](#) (#application-conf-check).

If any settings in the configuration file are changed or important settings are lost, the configuration may not work properly. Therefore, it is recommended to configure **container environment variables** in the Kubernetes environment. For more information about other agent control options, see [the following](#).

❗ Option unavailable in the Kubernetes environment

- `whatap.name` : Unique name by which the collection server identifies agents. The name is generated based on the object on which the agent is running. Random assignment of users may cause problems with agent identification.
- `whatap.onode` : The Kubernetes cluster node name is set by default. It is used to receive information about the node to which the agent belongs. When specified by the user, it may be difficult to accurately identify the node.

❗ Checking the `whatap.server.host` setting of the application

In some container images, you cannot directly set environment variables with `.`, such as `whatap.server.host`. If this keeps the application agent from starting normally, directly modify the `whatap.conf` file to set the `whatap.server.host` value. The procedure how to check the `whatap.server.host` setting of the application is as follows:

1. Enter into the WhaTap application container.
2. By executing the following command, check whether the `whatap.server.host` value is included.

```
""bash
$ env | grep whatap.server.host
""
```

If no value is output, it indicates that the container image cannot support configuration of the environment variables with `.`. In this case, use the configuration file instead of environment variables. Open the `whatap.conf` file and then manually set the `whatap.server.host` value as follows:

```
""bash
whatap.server.host={WhaTap Collection Server IP}
""
```

- [Including the latest version of the agent in the application image](#)
- Including the specified version of the agent in the application image

The following guides you to include **the latest version of the agent** in the application image.

1. Refer to the table below to check the Java APM agent version included in each kube_mon version.

Java APM version info as of July 2, 2025



whatap/kube_mon version	Java APM version
latest	2.2.61
1.8.7	2.2.61
1.8.6	2.2.55
1.8.5	2.2.52
1.8.4	2.2.50
1.8.3	2.2.50
1.8.2	2.2.48
1.8.1	2.2.47
1.8.0	2.2.42
1.7.16	2.2.41
1.7.15	2.2.39
1.7.14	2.2.39
1.7.13	2.2.39



whatap/kube_mon version	Java APM version
1.7.12	2.2.38
1.7.11	2.2.38
1.7.10	2.2.38
1.7.9	2.2.38
1.7.8	2.2.36
1.7.7	2.2.35
1.7.6	2.2.35
1.7.5	2.2.34
1.7.4	2.2.33
1.7.3	2.2.33
1.7.2	2.2.33
1.7.1	2.2.32
1.7.0	2.2.32
1.6.1	2.2.32
1.6.0	2.2.31
1.5.9	2.2.31

whatap/kube_mon version	Java APM version
1.5.8	2.2.30
1.5.7	2.2.28
1.5.6	2.2.27
1.5.5	2.2.27
1.5.4	2.2.26
1.5.3	2.2.26
1.5.2	2.2.26
1.5.1	2.2.26
1.5.0	2.2.26
1.4.9	2.2.25

- To include the WhaTap agent in the application image, add the following in the Dockerfile's final image build step. Change `{YOUR_PROJECT_ROOT}` to the top-level path of the project to install the agent.

```
# Create the WhaTap agent directory in the user container
RUN mkdir -p /whatap

# Copy the WhaTap Java agent to the user container
COPY --from=whatap/kube_mon /data/agent/micro/whatap.agent.kube.jar /whatap

# Copy the created whatap.conf file to the user container
COPY {YOUR_PROJECT_ROOT}/whatap.conf /whatap/
```

- Add the following to the startup command of Dockerfile.

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

See the following definition example of the following Dockerfile.

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
```

4. Build the Docker. Change `{YOUR_DOCKERFILE_DIR}` to the path where Dockerfile is located.

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

The following guides you to include **the specified version of the agent** in the application image.

1. To include the WhaTap agent in the application image, add the following in the Dockerfile's final image build step. Change `{YOUR_PROJECT_ROOT}` to the top-level path of the project to install the agent.

```
# Create the WhaTap agent directory in the user container
RUN mkdir -p /whatap

# Copy the WhaTap Java agent to the user container
COPY --from=whatap/kube_mon /data/agent/micro/whatap.agent-*.jar /whatap

# Copy the created whatap.conf file to the user container
COPY {YOUR_PROJECT_ROOT}/whatap.conf /whatap/
```

2. Check the version of the WhaTap agent.

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

Output example
whatap.agent-X.Y.Z.jar

3. Add the following to the startup command of Dockerfile. In **X.Y.Z**, enter the agent version that you checked before.

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

See the following definition example of the following Dockerfile.

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. Build the Docker. Change `{YOUR_DOCKERFILE_DIR}` to the path where Dockerfile is located.

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

1. Select [Download](#) to download the installation file ([whatap.agent.java.tar.gz](#)).
2. Upload the dockerfile to the server and then unzip the file.

❗ To download to the installation server, execute the following command.



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

- When unzipped, the **whatap** directory is created. Check the settings in the **whatap.conf** file under the **whatap** directory, and then modify it as follows:

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



Configuring the options for the Kubernetes internal application agent

The settings required for the Kubernetes internal application agent can be configured in the **whatap.conf** file or through the container **env** field when releasing the agent.

- The options in the **whatap.conf** file have higher priority than those set in the container **env** field.
- If no value is set in **whatap.conf** or container **env**, the default value is used.
- Use the In case of **license** and **whatap.server.host** options, use the **container environment variables** settings. For more information, check [the following] (#application-conf-check2).

If any settings in the configuration file are changed or important settings are lost, the configuration may not work properly. Therefore, it is recommended to configure **container environment variables** in the Kubernetes environment. For more information about other agent control options, see [the following](#).



Option unavailable in the Kubernetes environment

- whatap.name** : Unique name by which the collection server identifies agents. The name is generated based on the object on which the agent is running. Random assignment of users may cause problems with agent identification.
- whatap.onode** : The Kubernetes cluster node name is set by default. It is used to receive information about the node to which the agent belongs. When specified by the user, it



may be difficult to accurately identify the node.

4. Add the following to the startup command of Dockerfile. Check the agent file under the **whatap** directory, and then enter the version in **X.Y.Z**.

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

See the following definition example of the following Dockerfile.

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**: Monitoring target (sample application)
- **whatap.agent-X.Y.Z.jar**: WhaTap agent
- If the agent's file name is **whatap.agent-1.2.3.jar**, change the X.Y.Z part to 1.2.3.
- For more information about addition of JVM options, see [the following](#).



Checking the **whatap.server.host** setting of the application

In some container images, you cannot directly set environment variables with `ENV` such as `whatap.server.host`. If this keeps the application agent from starting normally, directly modify the **whatap.conf** file to set the `whatap.server.host` value. The procedure how to check the `whatap.server.host` setting of the application is as follows:



1. Enter into the WhaTap application container.
2. By executing the following command, check whether the `whatap.server.host` value is included.

```
```bash
$ env | grep whatap.server.host
```
```

If no value is output, it indicates that the container image cannot support configuration of the environment variables with `.`. In this case, use the configuration file instead of environment variables. Open the [whatap.conf](#) file and then manually set the `whatap.server.host` value as follows:

```
```bash
whatap.server.host={WhaTap Collection Server IP}
```
```



- For Java 17 or later, add the `--add-opens=java.base/java.lang=ALL-UNNAMED` option related to the reflection.

Setting the security key

Set a security key to query SQL variables and perform HTTP queries, or use the Thread stop function.

- Java Agent 2.2.2 or later

After creating the [security.conf](#) file in the `$WHATAP_HOME` path, enter a 6-character password consisting of alphabets and numbers as follows:

security.conf

```
paramkey=ABCDEF # SQL variable and HTTP query lookup
threadkill=ABCDEF # Thread stop function
```

- Java agent 2.2.2 or earlier

After creating the `paramkey.txt` file in the `$WHATAP_HOME` path, enter a 6-character password consisting of alphabets and numbers as follows:

`paramkey.txt`

```
ABCDEF # SQL variable and HTTP query lookup, thread stop function
```

- ✓ When updating the Java agent version to 2.2.2, the key values in the existing `paramkey.txt` file are automatically applied to the `security.conf` file. For example, if you used `FEDCBA` in `paramkey.txt` and updated to the version 2.2.2, it is applied to the `security.conf` file as follows:

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

If the `paramkey.txt` file does not exist, the key value of the `security.conf` file is automatically created as the key value of `WHATAP`.

- ⓘ • For agent settings related to recording SQL parameters, see [the following](#).
• For agent settings related to recording the HTTP parameters, see [the following](#).

Configuring the container environment variables

After building Docker, set **container environment variables** in the Kubernetes environment. Add the following items in the `yaml` file for releasing the application.

```
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"
```

See the following example:

```
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"

```

❗ Roles of environment variables

- `NODE_IP` : Collects the IP address of the node where the current Pod has been hosted.
- `NODE_NAME` : Collects the name of the node where the current Pod is running.
- `POD_NAME` : Collects the name of the current Pod.
- `OKIND` (optional): Groups applications for Pods. If set to a deployment name, the corresponding Pods are grouped together.
- `license` : Key for agent authentication.
- `whatap.server.host` : Host IP of the WhaTap collection server
- `whatap.micro.enabled` : Enables integration with containers.

Checking the agent installation

- To check whether the agent has been installed, go to [Dashboard](#) > [Application Service Dashboard](#).
- If you have manually installed the downloaded file but cannot see the agent in [Dashboard](#), check the following.
 - Execute the command, `ps -ef | grep whatap` in the container, and then check whether the agent options have been applied properly.
 - Check the content of the container in the `/whatap/logs` path. Agent logs are output in the format of `logs/{whatap configuration file name} -yyyymmdd.log`.

Installing Docker Node.js

At the bottom of **Management > Agent installation**, click **Install Application** > Select the **Docker Node.js** tab in Install guide.

This process applies the WhaTap monitoring agent to the Node.js application running on the Docker container and packages the container image as follows.

- ❗ • EKS Fargate is to be supported later.
- To help you understand the Node.js application integration process, Git example codes are provided. See [the following](#).

Download agent

Execute the following command to install WhaTap.

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

To update the agent, execute the following command.

```
npm update whatap
```

Agent CONFIG.

Add the following code to the first line of your application's main module.

Example

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

When using ECMAScript (ES), add the following code:

```
import WhatapAgent from 'whatap';
WhatapAgent.NodeAgent;
```

Configuring the container environment variables

After building Docker, set **container environment variables** in the Kubernetes environment. Add the following items in the **yaml** file for releasing the application.

```
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: WHATAP_OKIND
  value: {YOUR_OKIND_NAME}
- name: WHATAP_LICENSE
  value: {licenseKey}
- name: WHATAP_SERVER_HOST
  value: {proxyServer}
- name: WHATAP_MICRO_ENABLED
  value: "true"
```

See the following example:

```
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: WHATAP_OKIND
      value: #DeploymentName
    - name: WHATAP_LICENSE
      value: {licenseKey}
    - name: WHATAP_SERVER_HOST
      value: {proxyServer}
    - name: WHATAP_MICRO_ENABLED
      value: "true"

```

❗ Roles of environment variables

- `NODE_IP` : Collects the IP address of the node where the current Pod has been hosted.
- `NODE_NAME` : Collects the name of the node where the current Pod is running.
- `POD_NAME` : Collects the name of the current Pod.
- `WHATAP_OKIND` (optional): It groups applications for Pods. If set to a deployment name, the corresponding Pods are grouped together.
- `license` : Key for agent authentication.
- `whatap_server_host` : Host IP of the WhaTap collection server
- `whatap_micro_enabled` : Enables integration with containers.

Checking the agent installation

- To check whether the agent has been installed, go to **Dashboard > Application Service Dashboard**.
- If you have manually installed the downloaded file but cannot see the agent in **Dashboard**, check the following.
 - Execute the command, `ps -ef | grep whatap` in the container, and then check whether the agent options have been applied properly.
 - Check the content in the `/whatap/logs` path of the container.

Installing Docker Python

At the bottom of [Management](#) > [Agent installation](#), click [Install Application](#) > Select the [Docker Python](#) tab in Install guide.

This process applies the WhaTap monitoring agent to the Python application running on the Docker container and packages the container image as follows. To release the Kubernetes application, the Docker image is required. Create a Docker image with the whatap-python package installed using the following process.

- ❗ • EKS Fargate is to be supported later.
- To help you understand the Python application integration process, Git example codes are provided. See [the following](#).

Download agent

Install the whatap-python package when building your Python application docker image.

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

See the following definition example of the following Dockerfile.

```
python ver 3.10
```

```
# Install python3.10 In the Docker environment.
FROM python:3.10

# Set the working directory to /app.
WORKDIR /app

# Copy all the files and folders in the current directory to the /app directory in the container.
ADD . /app/

# In Python, any external library can be installed using pip.
# Install the WhaTap Python agent when building the docker image.
```

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

Configuration and execution of the agent

Before running the application, determine the Python agent's working directory and then create a default configuration file. Pass the authentication information to the agent and set whether or not to trace logs through the configuration file ([entrypoint.sh](#)).

Configuration file (entrypoint.sh) example

entrypoint.sh

```
#!/bin/bash

# Set the working directory of the container to the WhaTap path. The agent logs and configuration file are
# generated in the path.
export WHATAP_HOME=${PWD}

# If any permission error occurs, remove the following comment.
#chmod -R 777 $WHATAP_HOME

# The following lists the settings required for agent configuration. Configure them through the yaml file for release
# of applications.
whatap-setting-config \
--host $whatap_server_host \
--license $license \
--app_name $app_name \
--app_process_name $app_process_name

# The following comments are additionally configured for agent grouping and activation of log collection. Use them
# if necessary.

# Agent grouping
# echo "whatap.okind=$whatap.okind" >> whatap.conf

# Activation of log collection
# echo "logsink_enabled=true" >> whatap.conf
```



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

# Add whatap-start-agent before the application startup command to run the agent.
whatap-start-agent unicorn server:app --host 0.0.0.0 --port 8000
```

1. Set the container's working directory with the `WHATAP_HOME` environment variable. Create the agent log and configuration files in the path.

```
export WHATAP_HOME=${PWD}
```

2. Execute the following command to create and configure the `whatap.conf` file in the path set as `WHATAP_HOME`. Variables marked with `$` are required for agent configuration and are set through the `yaml` file for releasing the application.

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

3. Run the agent by adding `whatap-start-agent` before the application startup command as follows:

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



If any role issues, assign roles to the `$WHATAP_HOME` variable as follows:

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



- As an example of the `yaml` file, see [Container environment variable settings](#) as follows:
- The `license` is the user authentication information of WhaTap. **Do not expose to outside.**

Script running example

See the following definition example of the following Dockerfile. This is a complete example to run the `entrypoint.sh` script.

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

# Assign a role to run the entrypoint.sh script in the container.
RUN chmod +x ./entrypoint.sh

# Execute the entrypoint.sh script when generating the container.
CMD ["/entrypoint.sh"]
```

Additional function

The following settings are optional and must be used if necessary. In addition to the following additional settings in the configuration file (`entrypoint.sh`), log and transaction-related settings are also configurable. For additional settings, see [the following](#).

- Agent grouping

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

- Enabling the log collection

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

Configuring the container environment variables

After building Docker, set **container environment variables** in the Kubernetes environment. Add the following

items in the [yaml](#) file for releasing the application.

```
env:
- name: license
  value: XXXXXXXXXXXXXXXX-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: whatap.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}}
```

See the following example:

```
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: XXXXXXXXXXXXXXX-XXXXXXXXXXXX-XXXXXXXXXXXX
- 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}}

```

❗ Roles of environment variables

- `license` : Key for agent authentication.
- `whatap_server_host` : Host IP of the Whatap collection server
- `app_name` : Name to identify the agent for the application. For more information about the agent identification, see [the following](#).
- `app_process_name` : Set the target processes such as CPU and Heap Memory. For example, there are uwsgi and gunicorn.
- `whatap.okind` (optional): Groups applications for the Pod. If set using the deployment name, the corresponding Pods are grouped together.
- `NODE_IP` : Collects the IP address of the node where the current Pod has been hosted.
- `NODE_NAME` : Collects the name of the node where the current Pod is running.
- `POD_NAME` : Collects the name of the current Pod.

Checking the agent installation

- To check whether the agent has been installed, go to [Dashboard](#) > [Application Service Dashboard](#).
- Run the `ps -ef | grep whatap_python` command in the container to check whether the Whatap Python service is running normally.

Installing The Docker PHP

At the bottom of [Management](#) > [Agent installation](#), click [Install Application](#) > Select the [Docker PHP](#) tab in Install guide.

This process applies the WhaTap monitoring agent to the PHP application running on the Docker container and packages the container image as follows.

ⓘ EKS Fargate is to be supported later.

Download agent

Install the whatap-php package when building your PHP application docker image.

- **x64 AArch64**

x64

```
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 /
```

AArch64

```
RUN wget https://s3.ap-northeast-2.amazonaws.com/repo.whatap.io/alpine/aarch64/whatap-php.tar.gz
RUN tar -xvzf whatap-php.tar.gz -C /
```

Agent CONFIG.

- Use the following command to set the installation script. If you enter the WhaTap access key and server IP, the PHP environment data is automatically collected.

```
RUN (echo "[Access Key]"; echo "[Server IP]") | /usr/whatap/php/install.sh
```

- Using the following method, you can directly enter the PHP environment settings.

```
# RUN /usr/whatap/php/install.sh -l [Access Key] -s [Server IP] -v [PHP version x.x] -i [php.ini full path] -p [Process Name] -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]
```

ⓘ Be careful not to expose the **access key** to images that are in Docker Public status.

Configuring the container environment variables

After building Docker, set **container environment variables** in the Kubernetes environment. Add the following items in the **yaml** file for releasing the application.

```
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: WHATAP_PHP_AGENT_PATH
  value: /usr/whatap/php/whatap_php_static
- name: WHATAP_CONTAINER_CONF_PATH
  value: /usr/whatap/php
```

See the following example:

```
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}}
    - name: WHATAP_PHP_AGENT_PATH
      value: /usr/whatap/php/whatap_php_static
    - name: WHATAP_CONTAINER_CONF_PATH
      value: /usr/whatap/php
```

! Roles of environment variables

- `NODE_IP` : Collects the IP address of the node where the current Pod has been hosted.
- `NODE_NAME` : Collects the name of the node where the current Pod is running.
- `POD_NAME` : Collects the name of the current Pod.
- `WHATAP_PHP_AGENT_PATH` : Specifies the PHP agent executable file path.
- `WHATAP_CONTAINER_CONF_PATH` : Specifies the path where the container ID will be stored.

Running the agent

• Command Supervisor

Add the whatap-php startup command before the application startup command.

```
sh -c "/usr/whatap/php/whatap-php start && [Application Startup Command]"
```

In case of starting the application as supervisor, see the following code to add the whatap-php service.

```
RUN echo "[program:whatap-php]" >> /etc/supervisor/conf.d/whatap.conf
RUN echo "command = sh -c '/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

```

Execute the following command to check whether the Whatap service has been run normally. When the application server runs, it begins collecting monitoring information.

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

Checking the agent installation

To check whether the agent has been installed, go to **Dashboard > Application service dashboard**.

If you have manually installed the downloaded file but cannot see the agent in **Dashboard**, check the following.

- Execute the command, `ps -ef | grep whatap` in the container, and then check whether the agent options have been applied properly.
- Check the content of the container in the `/usr/whatap/php/logs` path. The file name of the agent log is displayed as `logs/ {boot or install} -yyyymmdd.log`.

Installing The Docker Go

At the bottom of [Management](#) > [Agent installation](#), click [Install Application](#), and then Select the [Docker Go](#) tab in Install guide.

This guide provides instructions on how to install the WhaTap monitoring agent in Docker container-based Go applications. Choose one of the following two methods:

- **Quick installation method:** Complete installation process from writing Dockerfile to running containers and checking monitoring data
- **Configuration by deployment environment:** Additional configuration methods for each architecture (x64/ARM64), Kubernetes, and Docker Compose environment

⚠ EKS Fargate support is planned.

✅ Before starting the installation, prepare the following information:

- WhaTap project **access key** (check from [Project menu] > [Management] > [Agent installation])
- WhaTap **collection server IP address**

Quick installation method

You can build a Docker image with WhaTap monitoring applied using only 3 files.

Step 1. Prepare required files

Create the following 3 files in the project root.

- [1. Dockerfile](#) 2. `entrypoint.sh` 3. `main.go` (example)

```
# =====
# Build stage - whatap-go-inst build
# =====
```

```
FROM golang:1.21-alpine AS builder
```

```
WORKDIR /app
```

```
# Install whatap-go-inst
```

```
#RUN go install github.com/whatap/go-api-inst/cmd/whatap-go-inst@latest
```

```
RUN wget -qO- https://github.com/whatap/go-api-inst/releases/latest/download/whatap-go-inst_linux_amd64.tar.gz | tar xz  
-C /usr/local/bin/
```

```
# Copy source
```

```
COPY go.mod go.sum ./
```

```
RUN go mod download
```

```
COPY . .
```

```
# Build whatap-go-inst
```

```
RUN whatap-go-inst go build -o /app/main .
```

```
# =====
```

```
# Runtime stage
```

```
# =====
```

```
FROM alpine:latest
```

```
WORKDIR /app
```

```
# Install WhaTap data relay agent
```

```
RUN apk add --no-cache wget
```

```
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 /
```

```
RUN rm whatap-agent.tar.gz
```

```
# Copy application
```

```
COPY --from=builder /app/main .
```

```
# Startup script
```

```
COPY entrypoint.sh /app/
```

```
RUN chmod +x /app/entrypoint.sh
```

```
ENTRYPOINT ["/app/entrypoint.sh"]
```

```
#!/bin/sh
```

```
# Generate whatap.conf (from environment variables)
mkdir -p /app/whatap_home
cat > /app/whatap_home/whatap.conf << EOF
license=${WHATAP_LICENSE}
whatap.server.host=${WHATAP_SERVER_HOST}
app_name=${WHATAP_APP_NAME:-myapp}
EOF

export WHATAP_HOME=/app/whatap_home

# Start data relay agent and run application
/usr/whatap/agent/whatap-agent start
exec ./main
```

```
package main

import "github.com/gin-gonic/gin"

func main() {
    r := gin.Default()
    r.GET("/", func(c *gin.Context) {
        c.JSON(200, gin.H{"message": "Hello WhaTap"})
    })
    r.Run(":8080")
}
```

Step 2. Build image

```
docker build -t my-go-app .
```

Step 3. Run container

Pass WhaTap configuration through environment variables.

```
docker run -d \
-e WHATAP_LICENSE={access key} \
```

```
-e WHATAP_SERVER_HOST={collection server IP} \
-e WHATAP_APP_NAME=myapp \
-p 8080:8080 \
my-go-app
```

Step 4. Check monitoring

1. Generate traffic to the application: `curl http://localhost:8080`
2. Check data in [WhaTap Monitoring Service](#)

✓ Refer to the [Troubleshooting](#) section.

Configuration by deployment environment

Dockerfile configuration by architecture

The quick start example above is based on **x64 (AMD64)**. For ARM64 environments, only change the agent download URL.

WhaTap agent URLs by architecture

| Architecture | Agent download URL |
|-----------------|--|
| x64 (AMD64) | <code>https://s3.ap-northeast-2.amazonaws.com/repo.whatap.io/alpine/x86_64/whatap-agent.tar.gz</code> |
| ARM64 (AArch64) | <code>https://s3.ap-northeast-2.amazonaws.com/repo.whatap.io/alpine/aarch64/whatap-agent.tar.gz</code> |

ARM64 Dockerfile modification example

Only change the agent installation part in the Dockerfile:

Change in Runtime stage

```

RUN apk add --no-cache wget && \
  wget https://s3.ap-northeast-2.amazonaws.com/repo.whatap.io/alpine/aarch64/whatap-agent.tar.gz && \
  tar -xvzf whatap-agent.tar.gz -C / && rm whatap-agent.tar.gz

```

View complete ARM64 Dockerfile

```
FROM golang:1.21-alpine AS builder
```

```
WORKDIR /app
```

*# Install whatap-go-inst**#RUN go install github.com/whatap/go-api-inst/cmd/whatap-go-inst@latest*

```

RUN wget -qO- https://github.com/whatap/go-api-inst/releases/latest/download/whatap-go-
inst_linux_amd64.tar.gz | tar xz -C /usr/local/bin/

```

```
COPY go.mod go.sum ./
```

```
RUN go mod download
```

```
COPY . .
```

```
RUN whatap-go-inst go build -o /app/main .
```

```
FROM alpine:latest
```

```
WORKDIR /app
```

```
RUN apk add --no-cache wget && \
```

```

  wget https://s3.ap-northeast-2.amazonaws.com/repo.whatap.io/alpine/aarch64/whatap-agent.tar.gz && \
  tar -xvzf whatap-agent.tar.gz -C / && rm whatap-agent.tar.gz

```

```
COPY --from=builder /app/main .
```

```
COPY entrypoint.sh /app/
```

```
RUN chmod +x /app/entrypoint.sh
```

```
ENTRYPOINT ["/app/entrypoint.sh"]
```

Kubernetes deployment

After building the Docker image, you can deploy it to a Kubernetes cluster.

Step 1: Push Docker image to registry

```
# Tag image
docker tag my-go-app:latest your-registry/my-go-app:v1.0

# Push to registry
docker push your-registry/my-go-app:v1.0
```

Step 2: Create WhaTap configuration Secret

```
kubectl create secret generic whatap-secret \
  --from-literal=license={access key}
```

Step 3: Deploy Deployment

- [deployment.yaml](#) service.yaml

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: my-go-app
spec:
  replicas: 3
  selector:
    matchLabels:
      app: my-go-app
  template:
    metadata:
      labels:
        app: my-go-app
    spec:
      containers:
        - name: my-go-app
```

```

image: your-registry/my-go-app:v1.0
ports:
- containerPort: 8080
env:
- name: WHATAP_LICENSE
  valueFrom:
    secretKeyRef:
      name: whatap-secret
      key: license
- name: WHATAP_SERVER_HOST
  value: "{collection server IP}"
- name: WHATAP_APP_NAME
  value: "my-go-app"
# Kubernetes metadata (optional)
- name: NODE_IP
  valueFrom:
    fieldRef:
      fieldPath: status.hostIP
- name: NODE_NAME
  valueFrom:
    fieldRef:
      fieldPath: spec.nodeName
- name: POD_NAME
  valueFrom:
    fieldRef:
      fieldPath: metadata.name

```

```

apiVersion: v1
kind: Service
metadata:
  name: my-go-app
spec:
  selector:
    app: my-go-app
  ports:
  - port: 80
    targetPort: 8080
  type: LoadBalancer

```

```
kubectl apply -f deployment.yaml
```

```
kubectl apply -f service.yaml
```

Environment variable description

Required environment variables

| Environment variable | Description | Example |
|----------------------|-----------------------------|---------------|
| WHATAP_LICENSE | WhaTap project access key | x4t2r... |
| WHATAP_SERVER_HOST | WhaTap collection server IP | 13.124.11.223 |
| WHATAP_APP_NAME | Application name (optional) | my-go-app |

Kubernetes metadata (optional)

Add these to pass pod execution location information to WhaTap:

| Environment variable | Description |
|----------------------|---|
| NODE_IP | IP of the node where the pod is running |
| NODE_NAME | Name of the node where the pod is running |
| POD_NAME | Name of the pod |

Docker Compose

```
docker-compose.yml
```

```
version: '3.8'
```

```
services:
```

```
  my-go-app:
```

```
    build: .
```

```
    ports:
```

```
- "8080:8080"
environment:
  - WHATAP_LICENSE={access key}
  - WHATAP_SERVER_HOST={collection server IP}
  - WHATAP_APP_NAME=my-go-app
restart: unless-stopped
```

```
docker-compose up -d
```

Troubleshooting

When monitoring data is not visible

Step 1: Check environment variables

Check if environment variables are correctly set inside the container:

```
docker exec -it <container_id> sh
echo $WHATAP_LICENSE
echo $WHATAP_SERVER_HOST
cat /app/whatap_home/whatap.conf
```

Step 2: Check agent execution

```
# Check agent process
docker exec -it <container_id> ps -ef | grep whatap

# Check agent logs
docker exec -it <container_id> cat /app/whatap_home/logs/whatap-*.log
```

Normal output example:

```
root 12 1 0 Jan21 ? 00:00:03 /usr/whatap/agent/whatap-agent
```

Step 3: Check network

Check if outbound connection to WhaTap collection server (port 6600) is possible:

```
docker exec -it <container_id> nc -zv {collection server IP} 6600
```

When instrumentation is not applied

Check detailed instrumentation logs

```
# Debug output during build
GO_API_AST_DEBUG=1 whatap-go-inst go build ./...
```

Common issues and solutions

| Symptom | Cause | Solution |
|-----------------------------|-------------------------------|---|
| Data not received | Environment variables not set | Check <code>WHATAP_LICENSE</code> , <code>WHATAP_SERVER_HOST</code> |
| Agent fails to start | Network blocked | Allow port 6600 outbound |
| Instrumentation not applied | Initialization missing | Check that <code>whatap-go-inst init</code> is added to Dockerfile |
| ARM64 error | Incorrect binary | Check agent URL for architecture |

Additional support

If the above methods do not resolve the issue:

1. Contact [WhaTap Support Center](#)
2. Provide the following information:

- Complete Dockerfile content
- Agent logs (/app/whatap_home/logs/)
- `docker logs <container_id>` output

Installation Troubleshooting

Installation troubleshooting based on the Helm chart

In case there is a `whatap` Helm repository with the same name

"whatap" already exists with the same configuration, skipping

If another `whatap` repository with the same name already exists, delete the existing `whatap` repository by using the following command and then proceed with installation.

```
helm repo remove whatap
```

In case there is another namespace named `whatap-monitoring` in the cluster

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

If the `whatap-monitoring` namespace exists in the cluster, it is because the user has already installed the Whatap Kubernetes agent based on the [yaml](#) file. In this case, delete the existing `whatap-monitoring` namespace and related resources by using the following commands.

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


In case the `clusterrole` and `clusterrolebinding` resources exist in the cluster

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

If the `clusterrole` and `clusterrolebinding` resources already exist in the cluster, delete them by using the following commands.

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

In case the `whatap-monitoring` namespace is unavailable

By default, the WhaTap Kubernetes agent is installed in the `whatap-monitoring` namespace. However, if the namespace is unavailable or you need to specify a specific namespace, follow the steps below to install the agent in the desired namespace.

Preparation

- **Chart version:** whatap/kube 1.7.11 or later

Check the chart version by using the following command.

```
helm search repo whatap/kube --versions
```

If the version is less than 1.7.11, update the chart version by using the following command.

```
helm repo update
```

- **Helm version:** Helm 3 or later

Installation procedure

Install the agent by executing the following command, after entering the namespace to use in `<namespace>`.

```
helm install whatap-kube-agent whatap/kube --create-namespace --namespace <namespace> --set
namespaceOverrideEnabled=true --wait -f values.yaml
```

Updating and deleting

- **Update**

Execute the following command to update the agent.

```
helm update whatap-kube-agent whatap/kube --namespace <namespace> --wait
```

- **Delete**

Execute the following command to remove the agent.

```
helm uninstall whatap-kube-agent whatap/kube --namespace <namespace>
```

Troubleshooting issues that occur when starting the agent

In case of installation of WhaTap Kubernetes Monitoring via an old YAML file

Unrecognized VM option 'UseCGroupMemoryLimitForHeap'

Error: Could not create the Java Virtual Machine.

Error: A fatal exception has occurred. Program will exit.

If you installed WhaTap Kubernetes Monitoring via the installation [yaml](#) downloaded before **August 24, 2022**, you may encounter an error or failure to start the node agent (whatap-node-agent).

Possible causes are as follows. The installation [yaml](#) file downloaded before August 24, 2022 contains a node agent launch script using the `UseCGroupMemoryLimitForHeap` JVM option.

Starting from WhaTap node agent **1.2.0** (released on November 21, 2022), the JDK used in the agent has been

upgraded, and the upgraded JDK version no longer provides the `UseCGroupMemoryLimitForHeap` option. If you run the node agent 1.2.0 or later with the WhaTap Kubernetes Monitoring installed using a `yaml` file downloaded before **August 24, 2022**, the startup fails with the message `Unrecognized VM option 'UseCGroupMemoryLimitForHeap'`.

Solution

To resolve this issue, reinstall the Kubernetes monitoring as follows:

1. Download the latest `YAML` file by using **Management > Agent Installation**
2. Install the agent by using the `yaml` file downloaded from the Kubernetes environment.
3. Check whether any errors are output during installation in the Kubernetes environment.
4. Make sure the WhaTap monitoring agent Pod is in Running state.

❗ Check the Kubernetes environment and the **Workload > Pod List** menu.

5. Check the logs generated by each monitoring agent through **Management > Kubernetes Agent List**

In case the agent is not installed properly due to DockerHub's image pull limit

If you pull images from multiple nodes simultaneously, the installation may fail with an error as follows. If you encounter any problem, reinstall with the following command by changing the image in `yaml`.

Viewing errors

```
kubectl describe pod/<WhaTap k8s agent's Pod name> -n whatap-monitoring
```

Lookup result

```
> > Failed to pull image "whatap/kube_mon:latest": rpc error: code = Unknown desc = toomanyrequests: You have reached your pull rate limit. Please try again later or log in to increase your rate limit.
```

In this case, change the DockerHub image in the `yaml` file to the public.ecr.aws repository and then reinstall it.

- If you downloaded the default installation yaml from the agent installation screen, you can change the image with the following command. When executing the command, the existing dockerHub image in yaml is replaced with the image of the [public.ecr.aws](#) repository.

```
sed -i 's|image: "whatap/kube_mon|image: "public.ecr.aws/whatap/kube_mon|g' kube_agent.yaml
```

Troubleshooting the application integration issues

In case the Java monitoring data is not displayed in the container map

Depending on the configuration and management for Cgroups in your Kubernetes environment, you may encounter the issues with obtaining container IDs in specific situations. To address this situation, WhaTap changed the mapping mechanism between the containers and applications for Kubernetes monitoring. This change applies to WhaTap **Kubernetes Agent 1.8.8** and **Java Agent 2.2.42**, and addresses the issue where container IDs are missing due to Cgroups.

Solution

Agent Update

To resolve container ID identification issues in the Kubernetes environment, use the latest version that supports accurate identification of data exchanged between the WhaTap Kubernetes agent and APM. Install **Kubernetes Agent 1.8.8 or later** and **Java Agent 2.2.42 or later**.

ⓘ Currently only the Java agents are supported, and APMs for other languages are supported in the future.

Configuring additional environment variables

If Java application information does not appear in **Container Map** even after updating the agent version, you can solve the problem by adding user environment variables. This setting adds the `WHATAP_JAVA_AGENT_PATH` option to the container manifest (yaml) environment variable where the Java APM has been installed, indicating

the absolute path to the Java agent within the container.

The following is an example of adding the corresponding option to the container environment variables where the Java APM has been installed.

```
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: <Project Access Key>
- name: whatap.server.host
  value: <collection server IP address>
- name: whatap.micro.enabled
  value: "true"
- name: WHATAP_JAVA_AGENT_PATH
  value: "/whatap/whatap.agent-2.2.35.jar"
```

How to check mapping between the container and APM

The following explains how to check that the container and applications are mapped correctly.

- If any white micro container exists inside the **Container Map** block in **Dashboard > Container Map**, it indicates a normal mapping.
- In **Analysis > Metrics Search**, if the `container` field for the `app_counter` category contains the container ID in which the application is running, the mapping is successful.

Setting The Kubernetes Agent Options

- When releasing the WhaTap Kubernetes agent, it can be set through the container `env` field.
- It can be configured by modifying the `whatap.conf` file in the container.
- The options in the `whatap.conf` file have higher priority than those set in the container `env` field.
- If no value is set in `whatap.conf` or container `env`, the default value is used.

ⓘ Setting the application agent options in the container

Configure necessary settings in `whatap.conf` for monitoring applications such as Java, Python, and Node.js in the container. For more information about available options in `whatap.conf`, see the following links:

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

Configuring the options for each container

whatap-node-helper

- `collect_volume_detail_enabled` `bool`

Default `true`

In the node agent, use the `exec df` command for all containers to enable searching the container volume information. It enables the function to collect storage usage and volume data for all containers running on the node.

- `use_cached_mount_info_enabled` `bool`

Default `false`

Sets whether to use the mount information (mountinfo) cache. If set to `true`, the node helper caches and reuses the mountinfo from the time of the first disk scan. In this case, Pod PVC volumes newly mounted after

the agent starts may be permanently omitted, so the default is `false`. The cost of re-reading mountinfo is negligible per collection cycle.

- **inject_container_id_to_apm_agent_enabled** `bool`

Default `true`

It enables container ID injection using `exec` in the WhaTap APM container. It automatically extracts the ID of the container identified by the WhaTap APM. You can inject this into the APM tool to associate the performance monitoring data with containers.

- **debug** `bool`

Default `false`

Set whether or not to enable the debugging mode. Set the log level to `DEBUG` when the debugging mode is enabled, otherwise to `INFO`.

- **kube_config_path** `string`

Default `~/.kube/config`

Specify the Kubernetes configuration file path. If not set, use `~/.kube/config`.

- **log_sys_out** `bool`

Default `true`

Set the log system output. When the option value is set to `true`, logging is left simultaneously in the standard output (stdOut) and a file. Any changes are immediately reflected in the logging system.

- **collect_kube_node_process_metric_enabled** `bool`

Default `true`

It is the flag to enable or disable the monitoring for Kubernetes-related processes on the node. If it is set to `true`, monitoring is enabled. If it is set to `false`, monitoring is disabled.

- **collect_kube_node_process_metric_target_list** `list`

Default

```
"kubelet,containerd-shim,containerd,docker,dockerd,crio,metrics-server,coredns,aws-k8s-agent,kube-proxy"
```

Basically, it monitors Kubernetes-related processes. Users can arbitrarily set the processes to monitor by modifying the process list.

① List of default monitoring target processes by WhaTap

- **kubelet**

This controller runs on each node, continuously monitors containers and communicates with the Kubernetes master to make sure that workloads are scheduled and run correctly. It is responsible for checking container status and managing node resources.

- **containerd**

It is a high-performance container runtime for running containers. It handles image management, container creation and execution, networking, and such.

- **dockerd**

It is the main process that manages the creation of Docker containers. This process includes building, releasing, running, networking, and data management for container images.

- **crio**

It is a lightweight container runtime designed for Kubernetes that implements the Container Runtime Interface (CRI) to efficiently manage container workloads.

- **coredns**

This process provides DNS services in the Kubernetes cluster and handles name resolution for services in the cluster. It also performs service discovery and load balancing, and supports seamless communication between services in the cluster.

- **aws-k8s-agent**

This agent manages Kubernetes network interfaces (VPC CNI) in the AWS environment. This process allocates and releases AWS network resources (ENIs) for inter-Pod communication. If a problem occurs, it may cause network connection failures or resource allocation errors.

- **kube-proxy**

This process for routing network traffic is performed in the Kubernetes cluster. This process runs on each node and enables communication between services and Pods. It also uses Linux networking technologies like iptables or IPVS to deliver the cluster's internal requests to the correct Pods. This enables service discovery and load balancing in the Kubernetes cluster.



- **kube-apiserver**

This Kubernetes API server is a central component that handles all requests in the cluster and manages the status of resources. It processes all requests in the cluster, including authentication, authorization, and admission control.

- **etcd**

It is a distributed key-value database that stores Kubernetes cluster status data. It plays an important role in saving and recovering the status of the cluster.

- **kube-controller**

This process manages and runs multiple controllers, maintains the status of cluster resources, and performs automated tasks. It performs various roles such as managing node and Pod status, maintaining replication, etc.

- **kube-scheduler**

It places Pods on appropriate nodes in the cluster. It selects an optimal node considering the Pod's requirements and the status of each node.

whatap-node-agent

- **shutdown** bool

Default `false`

Forcibly disconnect the collection server. This means stopping communication.

- **license** string

Agents are identified using the license issued to the project.

- **whatap.server.host** string

Default `13.124.11.223` or `13.209.172.35`

Set the IP of the collection server. In case of `delimiter`, use `/:`.

- **whatap.server.port** number

Default `6600`

Set the data receiving port of the collection server.

- **ONODE_NAME** `string`

Default `NODE_NAME` or `NODE_IP` environment variable value

Set a name to identify the agent node.

- **tcp_connection_timeout** `number`

Default `3000`

Set the timeout time when connecting whatap-master-agent to the collection server socket.

- **tcp_so_timeout** `number`

Default `60000`

Set the data sending/receiving timeout time after socket connection.

- **debug** `bool`

Default `false`

Set whether or not to enable the debugging mode.

- **gpuEnabled** `bool`

Default `false`

Set whether or not to enable the GPU monitoring.

- **count_interval** `number`

Default `5000`

Set the cycle to transmit metrics (`kube pod` , `container`) to the collection server. The unit of the default value is ms.

ⓘ This option may affect the on-screen data refresh cycle. For more information, see [Screen components affected by each metric](#).

- **collect_interval** `number`

Default `5000`

Set the cycle for the agent to request apiserver to collect metrics (`kube_pod` , `container`).

ⓘ This option may affect the on-screen data refresh cycle. For more information, see [Screen components affected by each metric](#).

- **collect_apiserver_metric_by_leader** `bool`

Default `false`

If the option is enabled, the node agent delegates metrics collection to the master agent. It requires to enable the `collect_apiserver_metric_by_leader` option of whatap-master-agent first. Through metric collection delegation, the apiserver load generated by the existing whatap-node-agent can be reduced.

- **kubelet_root_dir** `string`

Default `/var/lib/kubelet`

Sets the kubelet root directory path. The node agent uses this path to resolve symbolic links for container log paths. Use this in environments like Huawei Cloud Container Engine (CCE) where the kubelet root location differs from the default. In short, this option guides the agent to find log files in environments that do not follow the standard path.

- **container_log_path** `string`

No default

Directly specifies the container log directory path. If not set, the node agent searches the default path based on `kubelet_root_dir` and standard paths such as `/var/log/pods` in sequence. Only use this option to specify explicitly when automatic search is difficult, such as in CCE environments where the log path is non-standard.

ⓘ Only set `kubelet_root_dir` and `container_log_path` when the container runtime or cloud provider uses non-standard log paths. In typical EKS, GKE, AKS, or on-premises environments, leaving the defaults works correctly.

- **collect_conntrack_enabled** `bool`

Default `true`

Set whether to enable collection of the node conntrack usage. When enabled, the `conntrack_count` , `conntrack_max` , and `conntrack_pct` fields are sent to the `kube_node_perf` category. Nodes where the `nf_conntrack` module is not loaded do not send the fields. Supported on node agent 1.9.18 or later.

- **collect_pod_emptydir_enabled** bool

Default `true`

Set whether to enable collection of emptyDir volume usage. When enabled, the node agent queries the kubelet volume directory every 60 seconds and sends the data to the `kube_pod_volume` category. When set to `false`, the existing `df`-based collection path handles emptyDir volumes. Supported on node agent 1.9.22 or later.

- **pod_emptydir_max_files** number

Default `20000`

Set the maximum number of files to query in one emptyDir volume. A volume that hits the limit sends the `truncated` field as `1` to indicate that the value is smaller than the actual one.

- **pod_emptydir_max_volumes** number

Default `300`

Set the maximum number of emptyDir volumes to query on one node.

- **net_send_buffer_enabled** bool

Default `true`

Set whether to keep and resend the data that exceeds the send queue while the network to the collection server is disconnected. When enabled, data is moved to the retention buffer instead of being discarded when the send queue is full, and is sent in order from the oldest data after reconnection. Supported on node agent 1.9.17 or later.

- **net_send_buffer_minutes** number

Default `15`

Set the retention time of the data in the retention buffer. The unit of the default value is minutes. Data that exceeds the retention time is discarded from the oldest.

- **net_send_buffer_max_mb** number

Default `64`

Set the maximum memory to be used by the retention buffer. The unit of the default value is MB. When the limit is exceeded, data is discarded from the oldest, and only the discarded data is counted as `pack_dropped`.

.

whatap-control-plane-helper

- **port** string

Default `9496`

Set the web server port of whatap-control-plane-helper.

- **cycle** number

Default `5`

Set the cycle at which the control-plane-helper scrapes metrics. The unit of the default value is s.

- **collect_control_plane_monitoring_enabled** bool

Default `false`

It determines whether or not to enable the control-plane-monitoring function.

- **kube_config_path** string

Default `~/.kube/config`

Specify the Kubernetes configuration file path. If not set, use `~/.kube/config`.

- **kube_master_url** string

Default `~/.kube/config`

Set the URL of the control-plane to request for scraping. If not set, use `~/.kube/config`.

- **debug** bool

Default `false`

Set whether or not to enable the debugging (logging) for whatap-control-plane-helper.

- **kube_client_tls_verify** bool

Default `true`

If the option value is set to `false`, TLS authentication is not enforced and the authentication process can be skipped.

whatap-master-agent

- **shutdown** `bool`

Default `false`

Forcibly disconnect the collection server. This means stopping communication.

- **license** `string`

Agents are identified using the license issued to the project.

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

Default `13.124.11.223` or `13.209.172.35`

Set the IP of the collection server. In case of `delimiter`, use `/:`.

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

Default `6600`

Set the data receiving port of the collection server.

- **tcp_connection_timeout** `number`

Default `3000`

Set the timeout time when connecting whatap-master-agent to the collection server socket.

- **tcp_so_timeout** `number`

Default `60000`

Set the timeout time when connecting whatap-master-agent to the collection server socket.

- **debug** `bool`

Default `false`

Set whether or not to enable the `DEBUG` level logging in whatap-master-agent.

- **collect_apiserver_metric_by_leader** `bool`

Default `false`

Sets whether or not to enable whatap-master-agent to act as the leader of the scraper. When enabled, whatap-master-agent alone requests metrics to apiserver.

- **collect_control_plane_monitoring_enabled** `bool`

Default `false`

Set whether or not to enable the control-plane monitoring.

- **master_to_control_plane_connection_timeout** `number`

Default `5000`

Set the time until a connection timeout occurs when whatap-master-agent communicates with whatap-control-plane-helper. The unit of the default value is ms.

- **master_to_control_plane_read_timeout** `number`

Default `20000`

Set the time until a read timeout occurs when whatap-master-agent communicates with whatap-control-plane-helper. The unit of the default value is ms.

- **control_plane_debug** `bool`

Default `false`

Set whether or not to enable all debug logging in whatap-master-agent for control-plane monitoring.

- **store_k8s_object_manifest_enabled** `bool`

Default `true`

Set whether or not to enable the Object Manifest collection. When enabled, manifest information for cluster objects is collected once a day.

- **store_k8s_object_manifest_timezone** `string`

Default `UTC`

Set the time zone for collection of object manifests. If there is no separate setting, collection starts at midnight based on UTC by default.

For example, to run it at midnight in Korean time, set the time zone to `GMT+9` or `Asia/Seoul` .

- **store_k8s_object_manifest_max_fetch_limit** `number`

Default `50`

Set the view number of objects to collect at a time. If there is no separate setting, 50 items are requested upon request based on the default value of `50` .

For example, when setting the option `30` in a cluster with a total number of Pods of 90, pods are requested 3 times for 30 pods each.

- **collect_network_policy_enabled** `bool`

Default `true`

Set whether to enable event-based collection of NetworkPolicy objects. When enabled, a snapshot is sent to the `kube_network_policy` category whenever a NetworkPolicy is added, changed, or deleted. The `networkpolicies` view permission is required in the ClusterRole.

- **net_send_buffer_enabled** `bool`

Default `true`

Set whether to keep and resend the data that exceeds the send queue while the network to the collection server is disconnected. This option applies to both the node agent and the master agent. Supported on master agent 1.9.17 or later.

- **net_send_buffer_minutes** `number`

Default `15`

Set the retention time of the data in the retention buffer. The unit of the default value is minutes.

- **net_send_buffer_max_mb** `number`

Default `64`

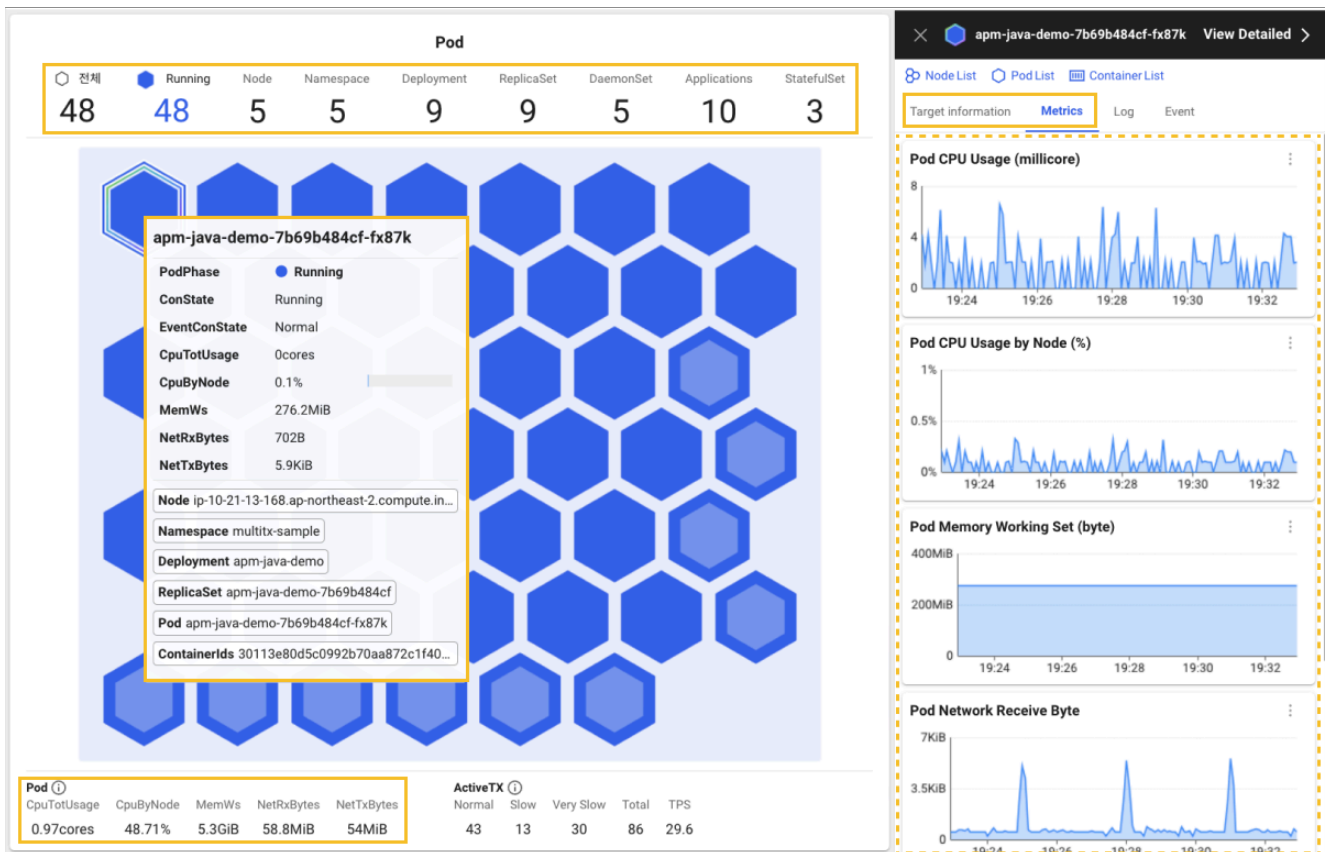
Set the maximum memory to be used by the retention buffer. The unit of the default value is MB.

Screen components affected by each metric

`kube_pod`

- [Dashboard](#) > [Container Map](#)

The Pod view reference data in the following example image is affected by the `kube_pod` metrics collection cycle and collection server transmission cycle.



- [Workload > Pod List](#)

The following example data is affected by the `kube_pod` metrics collection cycle and collection server transmission cycle.

| | 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% |

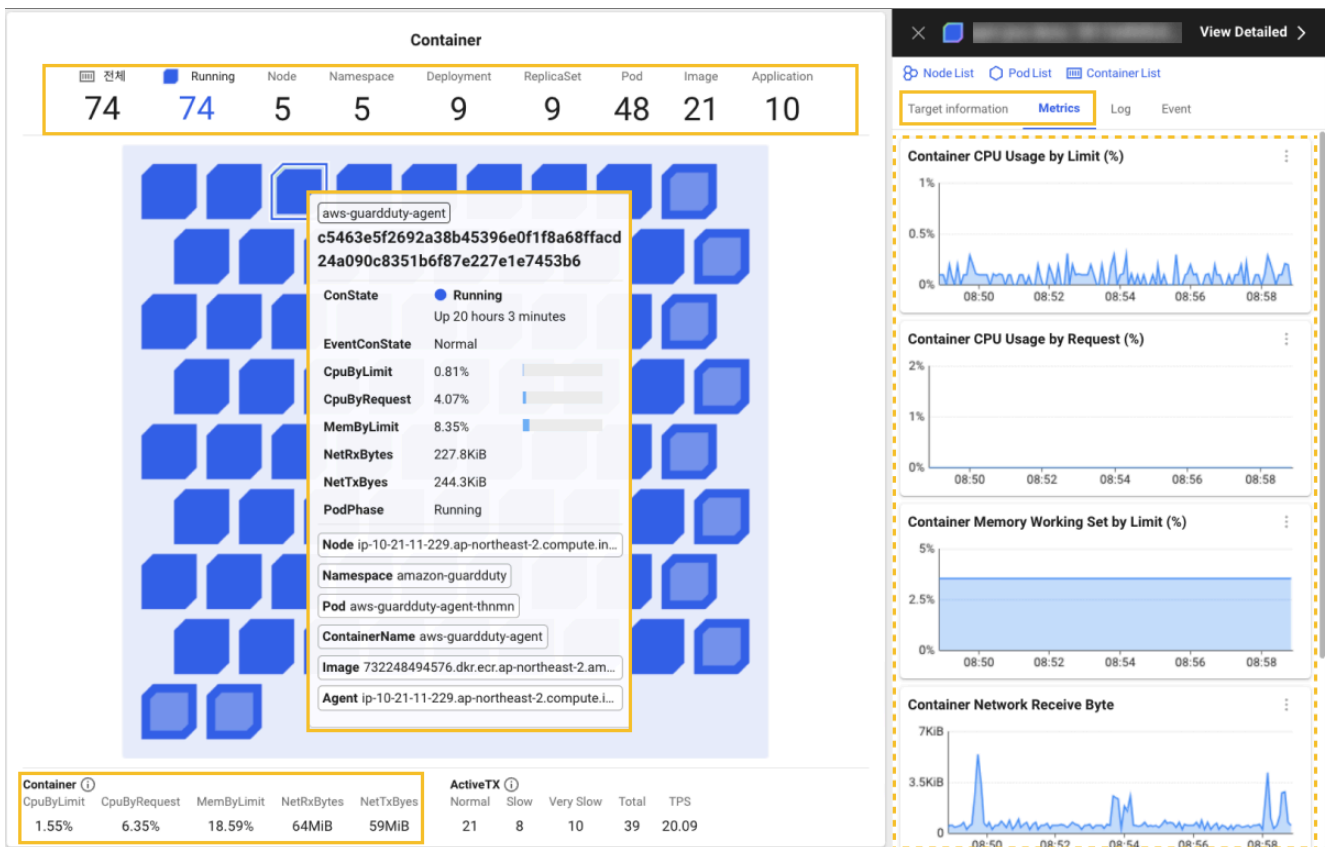
- [Dashboard > Performance Summary](#)

It is affected by **Number of pods running on nodes with agents** chart update cycle using the `kube_pod` metrics.

container

- [Dashboard > Container Map](#)

In the following example, the screen data refresh cycle based on the container view is affected by the `container` metrics.



- [Container > Container Images](#)

The data in the following example is affected by the `container` metrics.

| 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-efs-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 | sha256:16ebeda509ca6f641... | 5 | 0m | 29.07MiB |

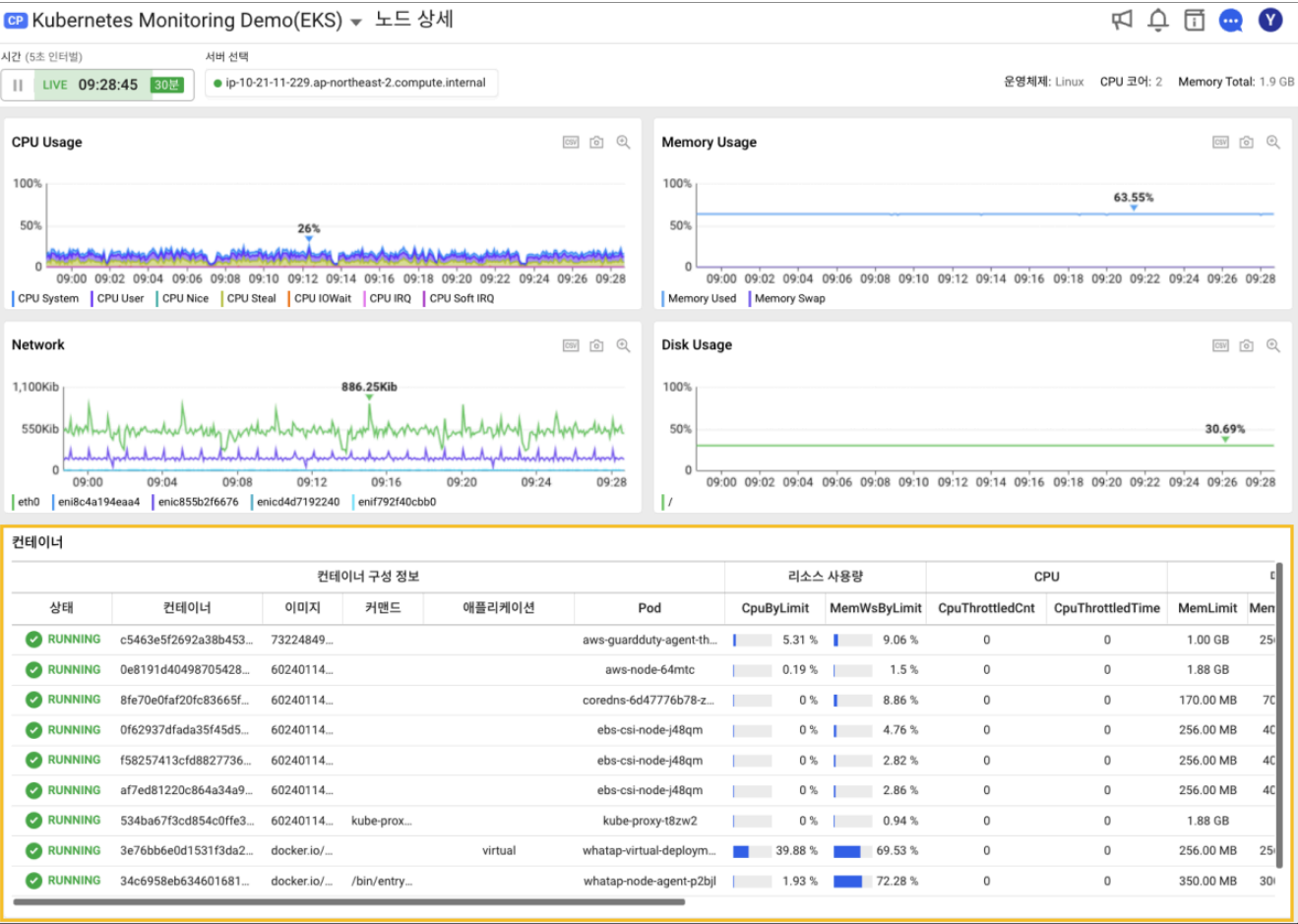
- [Dashboard > Performance Summary](#)

It is affected by the update cycle of the following charts using the `kube_pod` metrics.

- Number of containers running on nodes with agents
- Number of nodes with agents

- 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
- [Cluster](#) > [Node Detail](#)

In the following [Node details](#) screen as an example, the update cycle of the [container](#) list is affected by the `container` metrics.

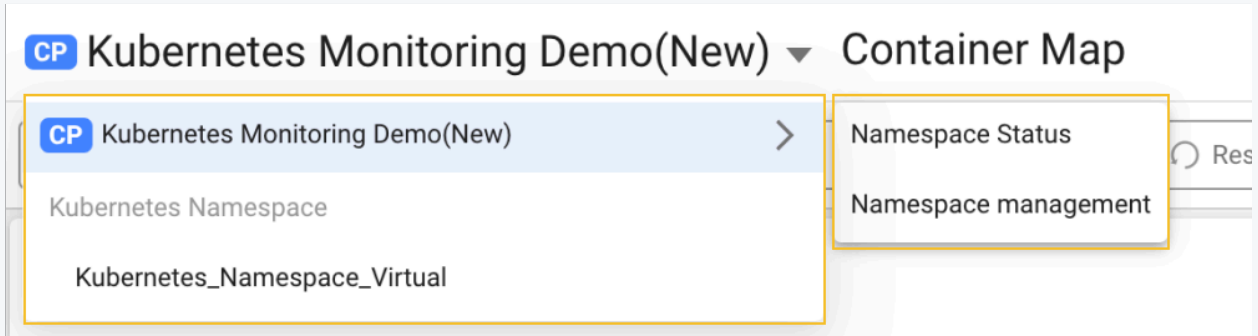


Namespace Management

Home > Select project > **Management** > **Namespace management**

In a cluster project (CP), you can create and manage namespace projects through the **Namespace management** menu. This menu is not available in a namespace project.

! The cluster project provides a list box at the top of the **Dashboard > Container Map** menu to move to a sub-namespace or to the **Namespace status / Namespace management** menus.



When to consider separating namespace projects

Consider separating a cluster project into namespace projects in the following situations.

- The cluster project contains many resources that can be split by Kubernetes namespace.
- Each Kubernetes namespace is owned by a different team or person, and you only need to monitor specific namespaces.

Managing namespace projects

The screenshot shows the 'Kubernetes Monitoring Demo(New)' interface. At the top, a summary bar (labeled 1) displays various resource counts: 38 Containers, 5 Namespaces, 34 Pods, 0 Deployments, 0 ReplicaSets, 4 Agent nodes, 8 Node CPU cores, 18.8GiB Node memory, and 10 Applications. Below this is the 'Namespace List' section (labeled 2), which contains a table of namespaces and their associated resources.

| Name | Group | Container | Namespace | Pod | Deployment | ReplicaSet | Agent node | Node CPU core | Node memory | Application | Setting |
|------------------------------|--------------|-----------|-----------|-----|------------|------------|------------|---------------|-------------|-------------|---------------|
| 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 | | | | | | | | | | | + Add project |
| kube-node-lease | | | | | | | | | | | + Add project |
| virtual-main | | | | | | | | | | | + Add project |

In the 1 area at the top, you can see the total number of targets and the number of key resources. In the 2 Namespace list area, you can search, add, and remove namespace projects.

Creating a namespace project

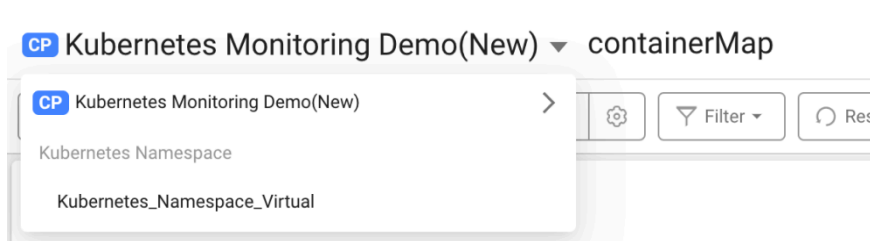
1. In **Namespace list**, click **+ Add project** for the item you want to create a project for.
2. When the **Add project** window appears, confirm the namespace name and click **OK**.

The 'Add project' dialog box is shown. It has a title bar with 'Add project' and a close button. The form contains two main fields: 'Namespace' with a text input containing 'default', and 'Group' with a dropdown menu showing 'Please select a group.'. Below these fields is an orange warning message: 'When a namespace project is added, the cluster project is separated into a additional WhaTap project. Therefore, even if the namespace project is added and deleted, it cannot be checked in the cluster project.' At the bottom right are 'Cancel' and 'OK' buttons.

3. The new namespace project is split into a separate WhaTap project and no longer appears in the cluster project's **Namespace list**.

| Name | Group | Container | Namespace | Pod | Deployment | Replicaset | Agent node | Node CPU core | Node memory | Application | Setting |
|------------------------------|--------------|-----------|-----------|-----|------------|------------|------------|---------------|-------------|-------------|---------------|
| Kubernetes_Namespace_Virtual | Demo Project | 4 | 1 | 2 | 2 | 2 | 1 | 2 | 3.7GB | 4 | |
| default | Demo Project | 2 | 1 | 3 | 2 | 2 | 1 | 2 | 1.2GB | 3 | |
| kube-portal | | | | | | | | | | | + Add project |
| kube-node-lease | | | | | | | | | | | + Add project |
| virtual-main | | | | | | | | | | | + Add project |

- To switch to the separated namespace project, use the project list box at the top of the cluster project's **Dashboard > Container Map** screen.



- ⓘ When you add a namespace project, the namespace is split from the cluster project (CP) into a separate WhaTap project. The new namespace project is issued a new **project access key**.

If application monitoring is already installed in the separated namespace, change the **project access key** in the application monitoring settings to the **project access key** of the new namespace project.

Deleting a namespace project

- In **Namespace list**, click the ⚙ **Configuration** icon at the far right of the project you want to delete.
- Go to **Project management**.
- Click the **Delete project** button at the top right of the screen.

- ⚠ When you delete a namespace project, the user list and all collected data are deleted. Deleted data cannot be restored, so proceed with caution.

- ⓘ For more information about project management, see [the following](#) document.

Correlated Project Management

This document provides information about the **Correlated Project Management** menu. Whatap provides integration visibility based on the data link between projects. You can search for linked projects through tags and quickly check key metrics that may cause problems in the project.

- It provides controls such as viewing, adding, and deleting the linked projects.
- The correlations between added projects can be analyzed.

Basic screen guide

Home > Select Project > **Management** > **Correlated Project Management**

Add a project to correlate

Select the project you want to correlate, then click the add button. (Supported types: Server, MySQL, PostgreSQL)

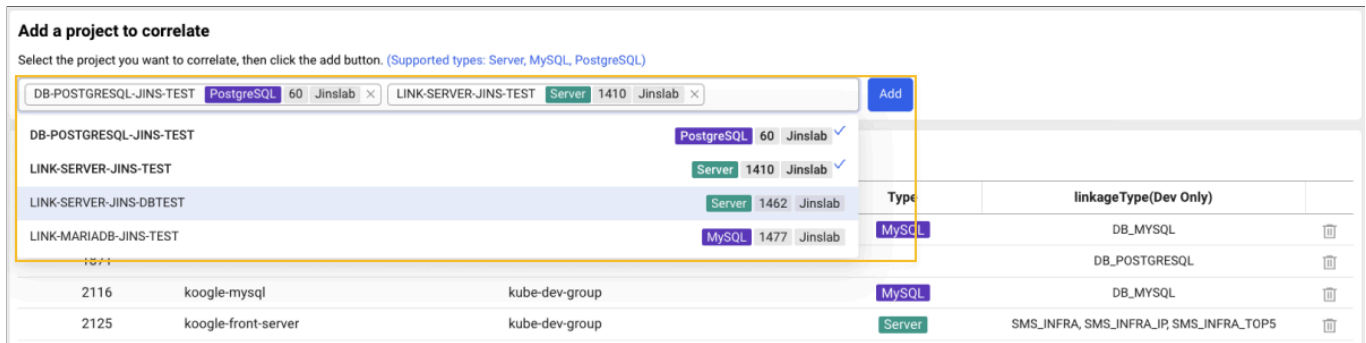
Correlated project list

| pcode | Project name | Group | Type | 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 | |

The following guides you to the basic screen layout of the **Correlated Project Management** menu. It consists of **Add a project to correlate** at the top and **Correlated project list** at the bottom.

Adding a correlated project

You can add any correlated project. By clicking the **Project selection** window, you can view the projects that can be linked as follows:



- In the **Project selection** window, you can perform search by **Project code** and **Project name**.
- If you select a project, the **Add** button is enabled. You can select multiple projects.
- If you click **Add** with a project selected, you can see that the project has been added to **Correlated project list** at the bottom along with a guide message.

Deleting a correlated project

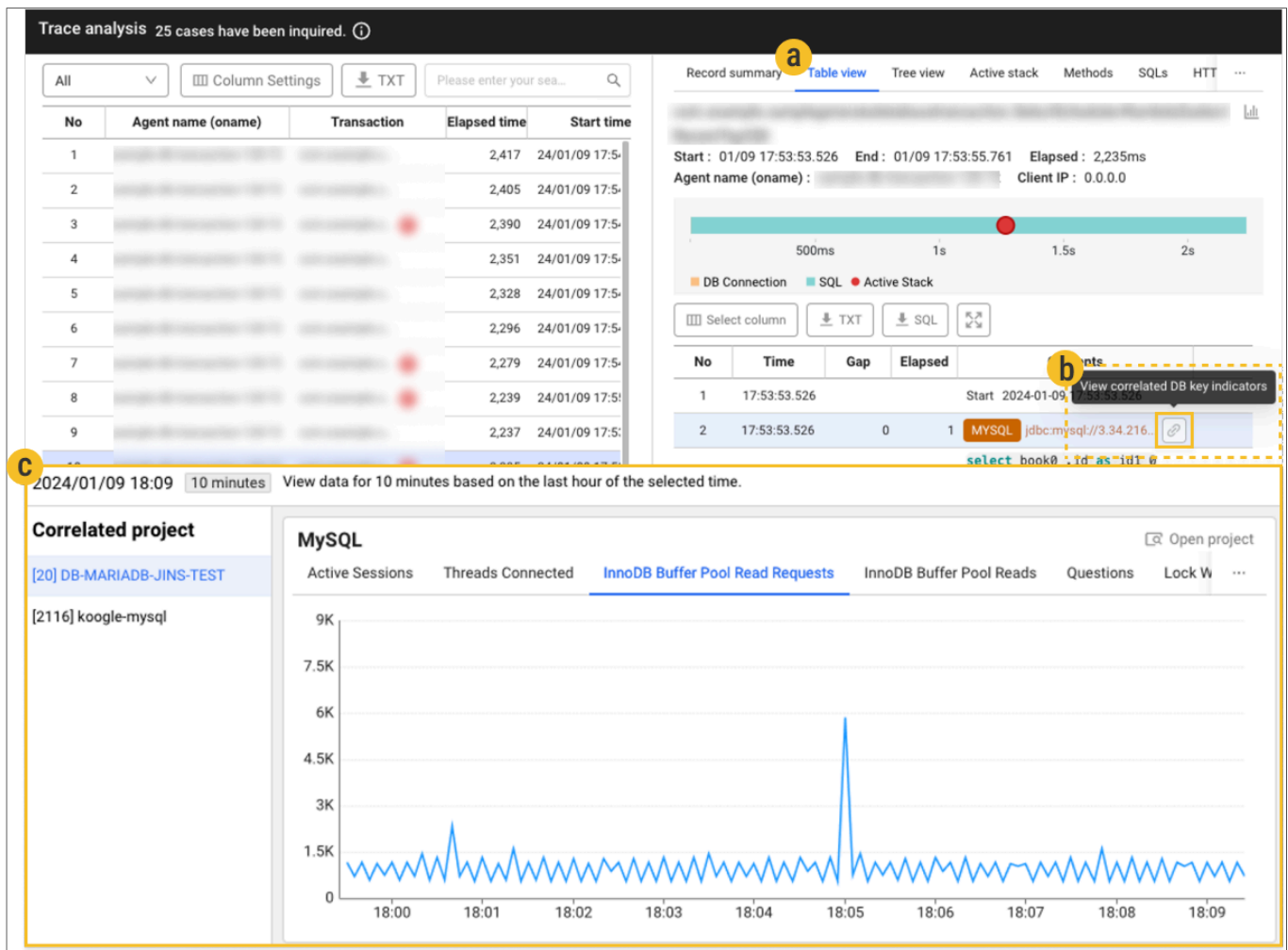
To exclude any added project from the correlations, click the  **Delete** icon on the right of the **Correlated project list**.

-  **Server**, **PostgreSQL**, and **MySQL** products are supported now.
- If the added correlated project does not appear in the list, wait a moment and then select the  **Refresh** icon to refresh information.
- In case of **Delete**, the **Edit project** role is required.

Correlation analysis example

Drag **Hitmap** and go to the **Trace analysis** window > Select a transaction > Select **Table view** on the right of the detailed analysis screen

The following provides a correlation analysis example based on the correlated projects.



1. By dragging **Hitmap** on the **Application service dashboard**, go to the **Trace analysis** window.
2. Select a transaction to view, and then select **a Table view** on the detailed analysis screen to the right.
3. If you select **Correlation icon** in the **b** area, you can see key metrics of the **c** correlated projects. If there are multiple correlated scenarios, the drop-down provides a list of desired analysis scenarios.

! Lookup time

It displays data for 10 minutes based on the last viewed time.

For example, if you drag data in the 34 to 36 minute range of **Hitmap**, the data from 26 to 36 minutes appear.

Kubernetes Agent

Let's learn about the configuration of the installed agent to monitor applications running in Kubernetes.

Agent structure

When installing WhaTap Kubernetes, configure resources for agents under the <whatap-monitoring>Namespace in the Kubernetes environment.

The master agent is for Deployment and the node agents are for DaemonSet. The following explains how to check the WhaTap agent installed in the Kubernetes environment.

- **Master Agent**

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

- **Node Agent**

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

The following is an example for the master agent and node agents running in the <whatap-monitoring>namespace.

```
$ kubectrl get pods -n whatap-monitoring
```

| NAME | READY | STATUS | RESTARTS | AGE |
|--------------------------------------|-------|---------|----------|-------|
| whatap-master-agent-57f4499567-ltbbr | 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 |

The following is the <whatap-monitoring> namespace displayed on the WhaTap Kubernetes container map.

whatap-monitoring Namespace inside the container map

Kubernetes Agent List

Under the **Management** menu of the cluster project (CP), you can go to the **Kubernetes Agent List** menu.

CP EKS-DEMO ▾ Kubernetes Agent List

🔊 🔔 📄 ⋮ 👤

🕒 15:24:17

Master agent installation status

Select Column

📄 CSV

| | Agent Info | | | Node info | | | |
|---|------------|---------|-----------------|-----------|-----|-------------|----------------|
| | State | Name | Version | OnodeName | Age | CapacityCpu | CapacityMemory |
| 🔍 | 🟢 ACTIVE | KMaster | 1.7.11 20240911 | | - | - | - |

Node agent installation status

All 5 🟢 ACTIVE 4 📄 UNINSTALLED 1

Node Name, Agent Name 🔍

Select Column

📄 CSV

| | Agent Info | | | Node info | | | | Project n |
|---|---------------|--------------------------|-----------------|--------------------------|-----------------|-------------|----------------|-----------|
| | State | Name | Version | OnodeName | Age | CapacityCpu | CapacityMemory | PodsCount |
| 🔍 | 🟢 ACTIVE | ip-10-21-110-36.ap-n... | 1.7.11 20240911 | ip-10-21-110-36... > | 1d 03h 46m 52s | 2 | 7.67GiB | 15 |
| 🔍 | 🟢 ACTIVE | ip-10-21-110-132.ap-n... | 1.7.11 20240911 | ip-10-21-110-132... > | 50d 00h 18m 31s | 2 | 3.76GiB | 17 |
| 🔍 | 🟢 ACTIVE | ip-10-21-140-131.ap-n... | 1.7.11 20240911 | ip-10-21-140-131... > | 20h 18m 07s | 2 | 3.77GiB | 13 |
| 🔍 | 🟢 ACTIVE | ip-10-21-110-178.ap-n... | 1.7.11 20240911 | ip-10-21-110-178... > | 12h 00m 48s | 2 | 1.88GiB | 7 |
| 🔍 | 📄 UNINSTALLED | - | - | ip-10-21-110-253.ap-n... | 2d 04h 49m 39s | 2 | 1.88GiB | - |

- You can see the installation progress of the Kubernetes agent. If some agents are not installed (UNINSTALLED) or inactive (INACTIVE), data collection may be distorted, so quick checking is required.
e.g. UNINSTALLED, ACITVE, INACTIVE in the State column
- You can check the CPU Limit assigned to the master agent and the installed node agent count.
For example, if the node agent count is 50 or more, it is recommended to set the master agent's CPU Limit to 500 millicores or more.
- You can see the CPU and memory allocated to the node where the agent has been installed.
e.g. CapacityCpu, CapacityMemory under the Node Info column
- You can see the nodes on which the agents are not installed.
- You can see the key data of the agent.
- You can find the guide to the node agent version.

e.g. Mismatch notification upon version discrepancy between nodes

ⓘ Master agent's resource allocation

Because the amount of data collected and processed by the WhaTap agent differs depending on the size of the user's Kubernetes environment (or node count), sufficient resource allocation to the master agent is required. For example, if [node agent count](#) is so great, it is recommended to increase the master agent's [CPU limit](#).

Refresh

If the  [Refresh](#) icon is selected at the top, you can update the data based on the current time.

Select Column

| Group | Column name | Description |
|------------|-------------------------------|--|
| Agent Info | Name | It indicates the agent alias when the agent name or agent alias is set. |
| | Original Name | It is the agent name. It is usually the same as the node name. |
| | Alias | It indicates the custom agent's alias. |
| | Initial | It is an abbreviation used when indicating the agent. |
| | Id | It is a unique ID of the agent. |
| | Start Time | It is the time when the agent first started running. |
| | Active Time | It is the time when the agent is currently active and starts collecting the monitoring data. |
| | Inactive Time | It is the time when the agent became inactive and data collection stopped. |
| | Version | It is the current software version of the agent. |

| Group | Column name | Description |
|-------------------|-----------------|---|
| | IP | It indicates the IP address of the server where the agent has been installed. |
| | CPU Cores | It indicates the number of CPU cores on the server where the agent has been installed. |
| Node info | OnodeName | It is the name of the node where the agent has been installed. |
| | Onode | It is a hash value that has converted the name of the node where the agent has been installed. |
| | Age | It indicates the time elapsed since node creation. |
| | CapacityCpu | It indicates the total CPU capacity of the node. |
| | CapacityMemory | It indicates the total memory capacity of the node. |
| Project resources | PodsCount | It indicates the number of Pods running on a node where the node agent has been installed. Separated namespaces are excluded. |
| | ContainersCount | It indicates the number of containers running on a node where the node agent has been installed. Separated namespaces are excluded. |

When using the [Select column](#) option, you can select a desired agent to view the agent list.

- The selected column list is kept even if you re-enter the menu.
- If you select [Reset](#), the predefined default key metric columns are selected. In this case, the column selection history is deleted.
- After selecting a column, you can click [Apply](#) at the lower right to apply the selected values.

Download

If the ☒ [Download](#) icon is selected, you can download the data in the CSV format by setting the [range](#) and [display option](#).

Display Detail

If you select the  icon in the master agent's [Detail](#) column, the detailed information window appears on the right. You can see the [environment variables](#), [heap histograms](#), [thread list/dump](#), and [agent logs](#).

In the node agent list, if you select the  icon in the [Detail](#) column of the agent to view, the detailed information window for the node appears on the right.

 For more information about the node details, see [the following](#).

Agent Update

- ⓘ • EKS Fargate is to be supported later.
- The latest version of the WhaTap Kubernetes agent can be checked in hub.docker.com.

If an agent update is required, execute the following command:

- **Master Agent**

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

- **Node Agent**

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

- **Agent application updates upon installation based on the Helm chart**

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

Agent Deletion

Deleting the Kubernetes agent

In the agent installation procedure, check the [yaml](#) file name that matches the downloaded Kubernetes version and Container Runtime, and then proceed with the procedure to delete the Kubernetes agent.

YAML file name of the Kubernetes agent

| Docker | YAML file name |
|--|---|
| Kubernetes 1.16 or later | whatap_kube_1.16.yaml |
| Kubernetes 1.16 or later (GPU version) | whatap_kube_1.16_gpu.yaml |
| Below Kubernetes 1.16 | whatap_kube.yaml |
| Below Kubernetes 1.16 (GPU version) | whatap_kube_gpu.yaml |

| Containerd | YAML file name |
|--|--|
| Kubernetes 1.16 or later | whatap_kube_containerd_1.16.yaml |
| Kubernetes 1.16 or later (GPU version) | whatap_kube_containerd_1.16_gpu.yaml |
| Below Kubernetes 1.16 | whatap_kube_containerd.yaml |
| Below Kubernetes 1.16 (GPU version) | whatap_kube_containerd_gpu.yaml |

| CRIO YAML | File name |
|--|--|
| Kubernetes 1.16 or later | whatap_kube_crio_1.16.yaml |
| Kubernetes 1.16 or later (GPU version) | whatap_kube_crio_1.16_gpu.yaml |

| CRIO YAML | File name |
|-------------------------------------|--|
| Below Kubernetes 1.16 | <code>whatap_kube_crio.yaml</code> |
| Below Kubernetes 1.16 (GPU version) | <code>whatap_kube_crio_gpu.yaml</code> |

In case the YAML file exists

1. Delete the Kubernetes agent installed with the `yaml` file name downloaded during the installation procedure. Modify the notation of the following command to your `yaml` file name, and then run it.

```
kubectrl delete -f < user's own YAML file name >
```

2. If the Kubernetes agent has been deleted successfully, you can see the following:

```
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
```

In case the YAML file does not exist

- ⚠ If you have created custom resources directly in the `whatap-monitoring` namespace, those resources are deleted also.

1. Execute the following command to delete the Kubernetes agent.

```
kubectrl delete namespace whatap-monitoring
```

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

2. If the Kubernetes agent has been deleted successfully, you can see the following:

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

Uninstalling the agent upon installation based on the Helm chart

Execute the following command to uninstall the WhaTap Kubernetes agent application.

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

Container Map

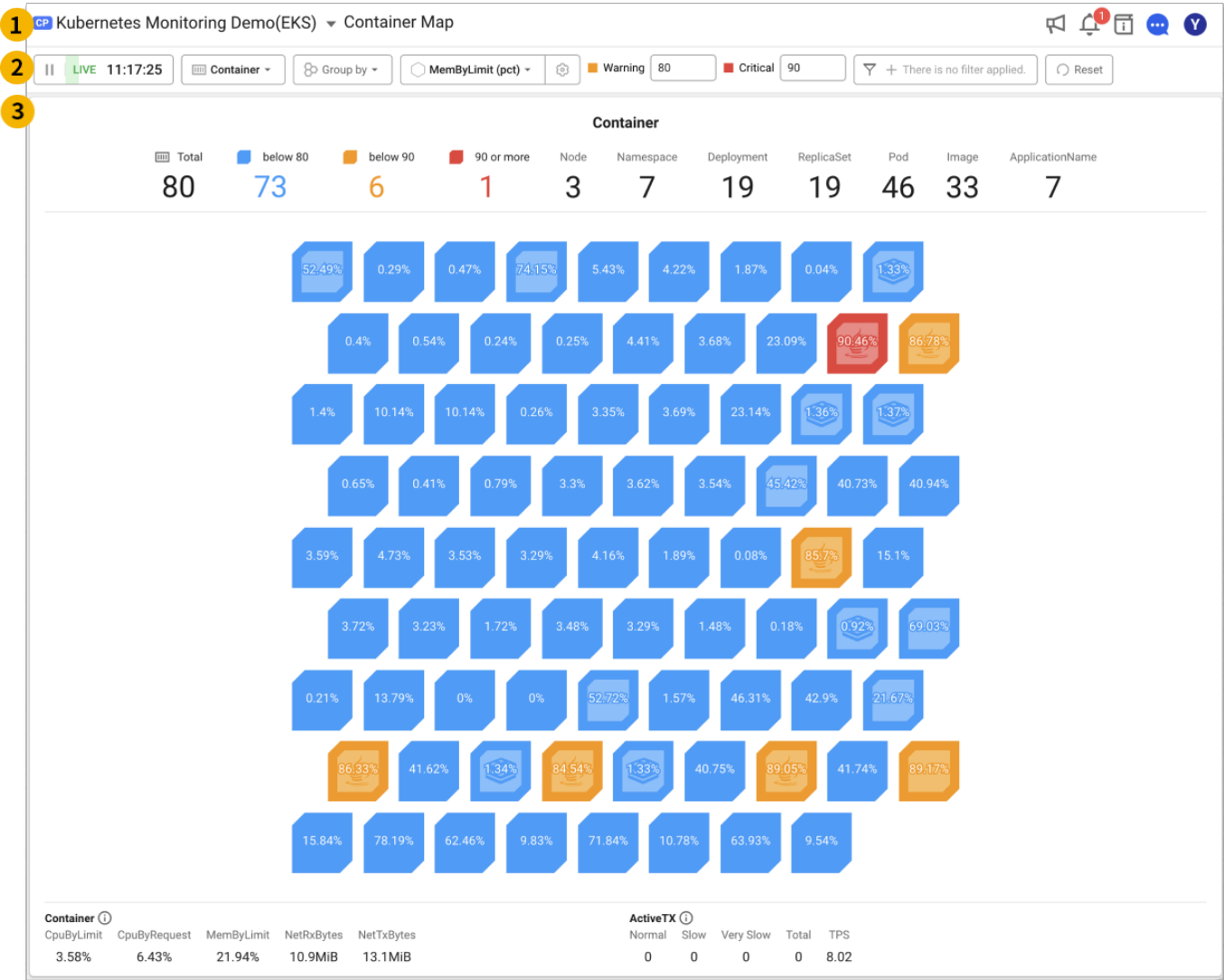
Home > Select Project > [Dashboard](#) > [Container Map](#)

The [Container Map](#) in WhaTap Kubernetes is the starting point for Kubernetes monitoring. Depending on the custom view, each object represents a container or pod.

- Using the container map, you can select targets to analyze and organize groups.
- By setting the status metrics and thresholds, you can see the status of individual containers or pods at a glance.
- You can select individual containers or pods for detailed information. It provides key monitoring data such as [Trace](#), [Metrics](#), [Logs](#), and [Event](#).

The following video clip is the guide to the [Container Map](#) menu.

Basic screen guide



Container ⓘ

CpuByLimit

CpuByRequest

MemByLimit

NetRxBytes

NetTxBytes

3.58%

6.43%

21.94%

10.9MiB

13.1MiB

ActiveTX ⓘ

Normal

Slow

Very Slow

Total

TPS

0

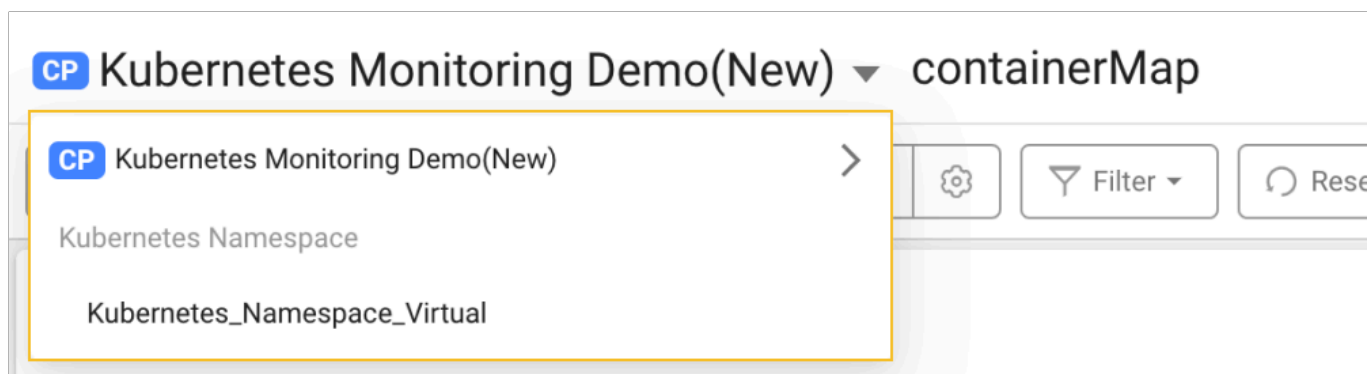
0

0

0

8.02

1 Go to Cluster Project - Namespace Project



When a namespace project is created under the cluster project **CP**, if you select **More** on the container map, a project list box appears. Through the project list box, you can see the project list including cluster projects and move between them.

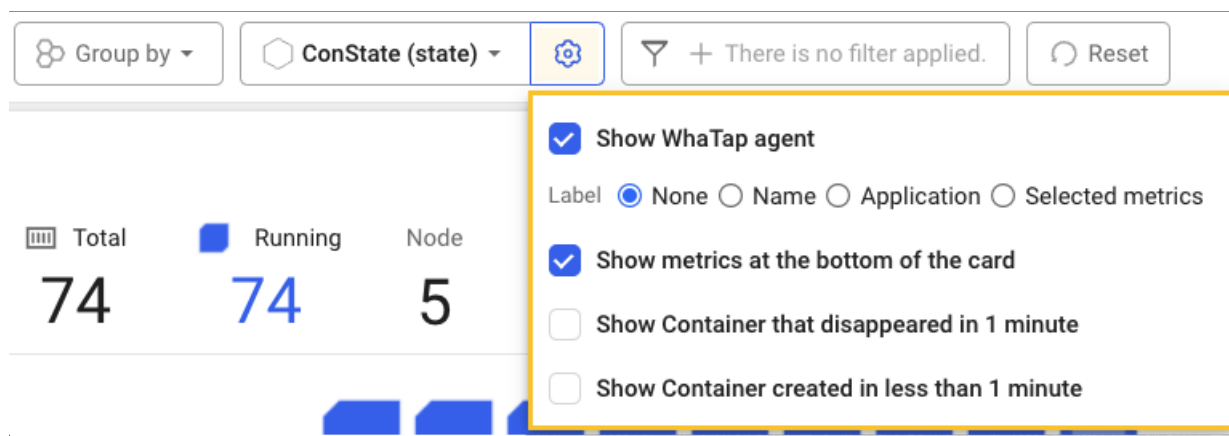
! For more information about the namespace project creation and management, see [the following](#).

2 User configuration



A container map provides multiple user settings that allow you to create any desired screen view. The functions available in the screen are as follows in order.

- **Category:** You can select the targets to analyze. You can also set the [container](#) or [pod](#).
- **Group by:** Through the [Group by](#) (clustering) option, the hierarchical relationships between cluster components can be displayed in two levels. For more information, see [the following](#).
- **ConState(state):** You can select any chart metric.
- If you click , you can select any option to display information of containers or pods on each object.



- [Show WhaTap Agent](#): The container or pod data can be displayed. Select a desired [label](#) option. For more information about the label option, see [the following](#).
- [Show metrics at the bottom of the card](#): You can select whether or not to display the metrics information at the bottom.
- [Show containers or Pods created within 1 minute](#): Option to display containers or Pods created within 1 minute.
- [Show containers or Pods disappeared within 1 minute](#): Option to display containers or Pods that disappeared within 1 minute.
- [Filter](#): You can see the status of containers or pods by filtering them based on your selections.
- [Reset](#): It initializes the screen view set by you.

Metric selection and threshold setting

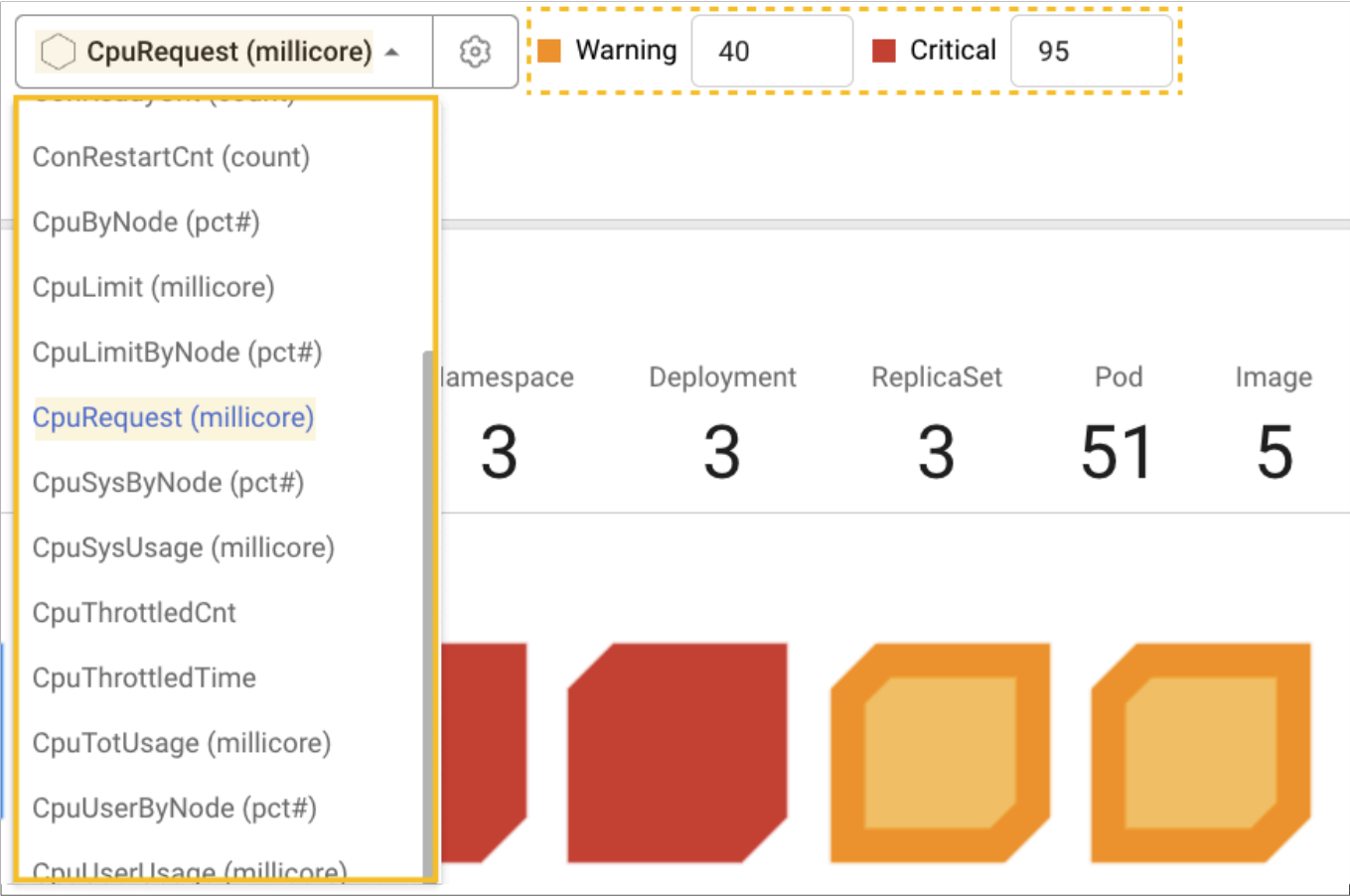


Chart metric selection and threshold setting determine the status on the container map. If the selected metric has numeric data, you can set thresholds. The set thresholds are displayed in color according to the following categories:

- More than 0 and Less than **Warning**: Blue
- More than **Warning** and Less than **Critical**: Orange
- More than **Critical**: Red



- Numerical metric units include **pct** (percentage), **bps**, **millicore**, and such. The **pct#** metric unit can have a value greater than 100%.
- For more information about the chart metric selection and threshold setting, see [the following](#).

Label options

- **None:** Labels are not displayed in the block.
- **Name:** The container or pod name is displayed.
- **Application:** The application in the container or pod is displayed.
- **Selected metrics:** The chart metric selected in `ConState(state)` is displayed.

Real-time and past search

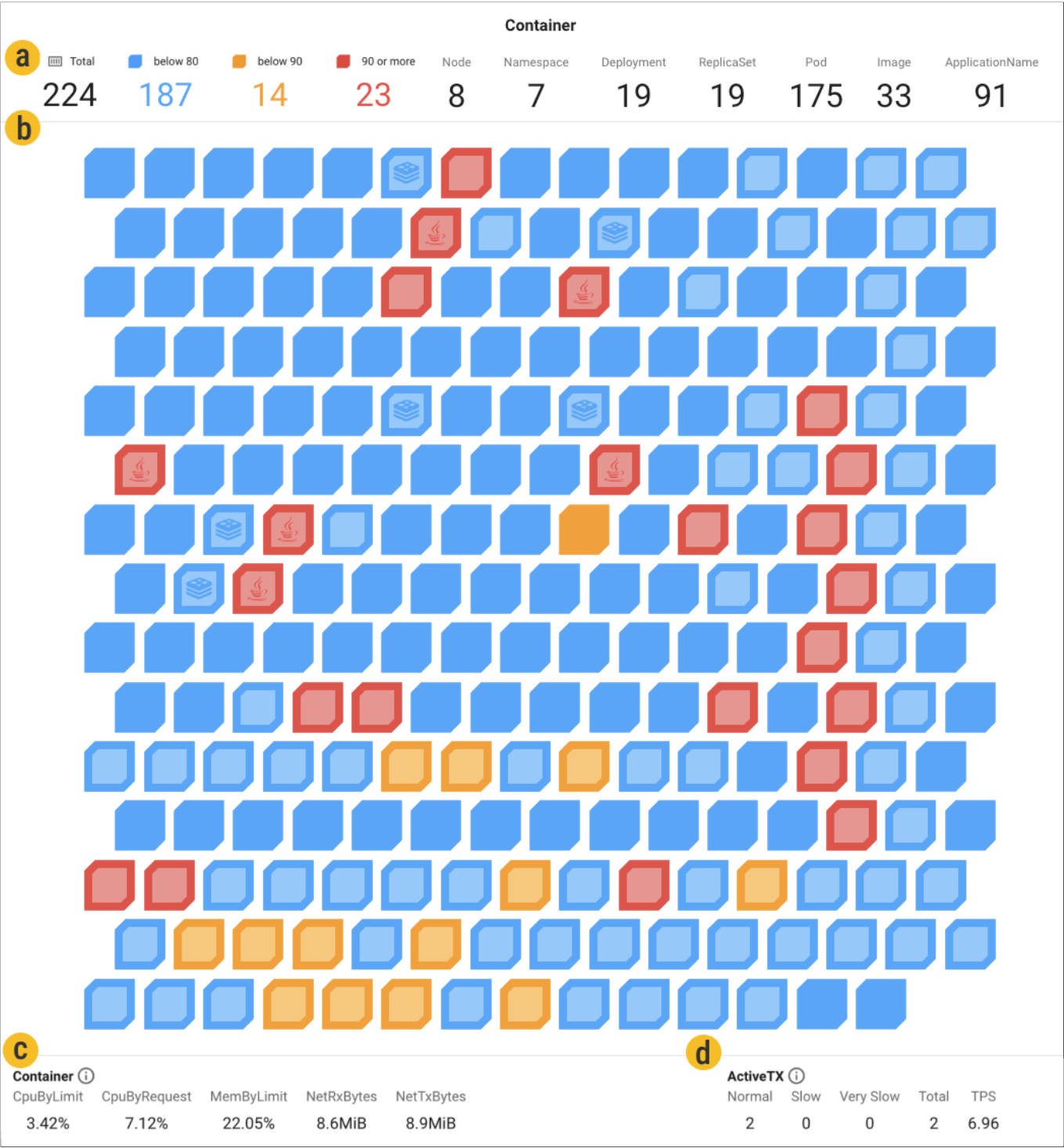
Container maps are based on real-time lookups based on multiple viewpoints. Use the time selector at the top left of the screen to change the real-time inquiry.



| Lookup status | Description |
|--------------------------|--|
| Real-time inquiry (LIVE) | Default mode
Real-time status changes every 10 seconds. |
| Paused inquiry (PAUSE) | Real-time status change OFF
Used to check the current status in detail. |
| Historical inquiry | You can specify a past time to check
Year/Month/Day/Hour/Minute |

ⓘ For more information on how to use the time selector.

3 Container map chart



- **a** The total number of grouped targets and the key resource count are displayed.
- **b** The shapes and colors for each object display the characteristics and status for containers or pods.

- **Linking an external project**

You can check external projects linked through logos within the block. For more information, see [the following](#).

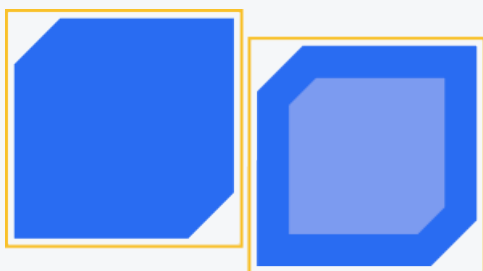
- **Color**

It indicates the status of the container or Pod. For more information about the status other than **Normal**, **Warning**, and **Critical**, see [the following](#).

- **Outline**

- Some objects have borders. The targets are what have installed microagents like Java, Node.js, Python, and such in the container.
- If there are 2 containers in a pod and the micro agents are installed on only 1 of 2 containers, the borders are displayed. In other words, if any of the containers in the pod have micro agents installed, the borders are displayed.

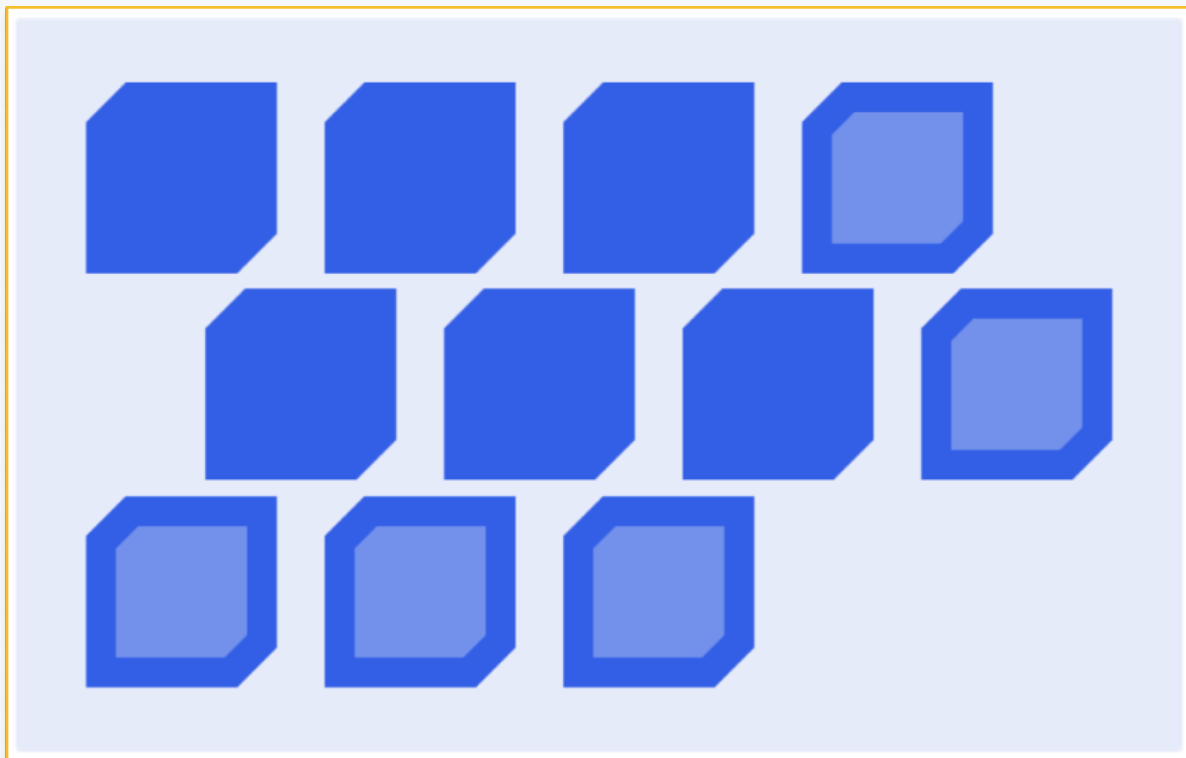
! Block and micro block



In the [Container map](#), the single polygon is called a **block**. At this time, the block with an agent (a block within a block) is called a **micro block**.

! Group

①



When the entire block is selected on the [Container Map](#) screen, the multiple block is called a group.

- **c** It displays the metrics related to the CPU and memories for the grouped targets.
- **d** It displays the transaction information for the applications in the grouped targets.

Container map targets

You can select an interested container or pod to view the summary and detailed information. The information includes the key monitoring data as follows:

- **Target Info**

By clicking on a block, you can specify a target, to view the related resource information and navigate to the corresponding menu link.

- **Resource relations**

It provides a relational chart that visually represents resource information related to the selected target.

- **Trace**

If the WhaTap application agent has been installed in the container, it provides transaction trace information for the application so that you can check it in real time on the dashboard.

- **Metrics**

The dashboard provides metrics such as CPU usage, memory usage, and network traffic of the selected container or Pod.

- **Log**

It provides the standard output logs of each container and the logs generated and stored by containers and applications.

- **Event**

It provides information on the events occurred in Kubernetes and WhaTap events.

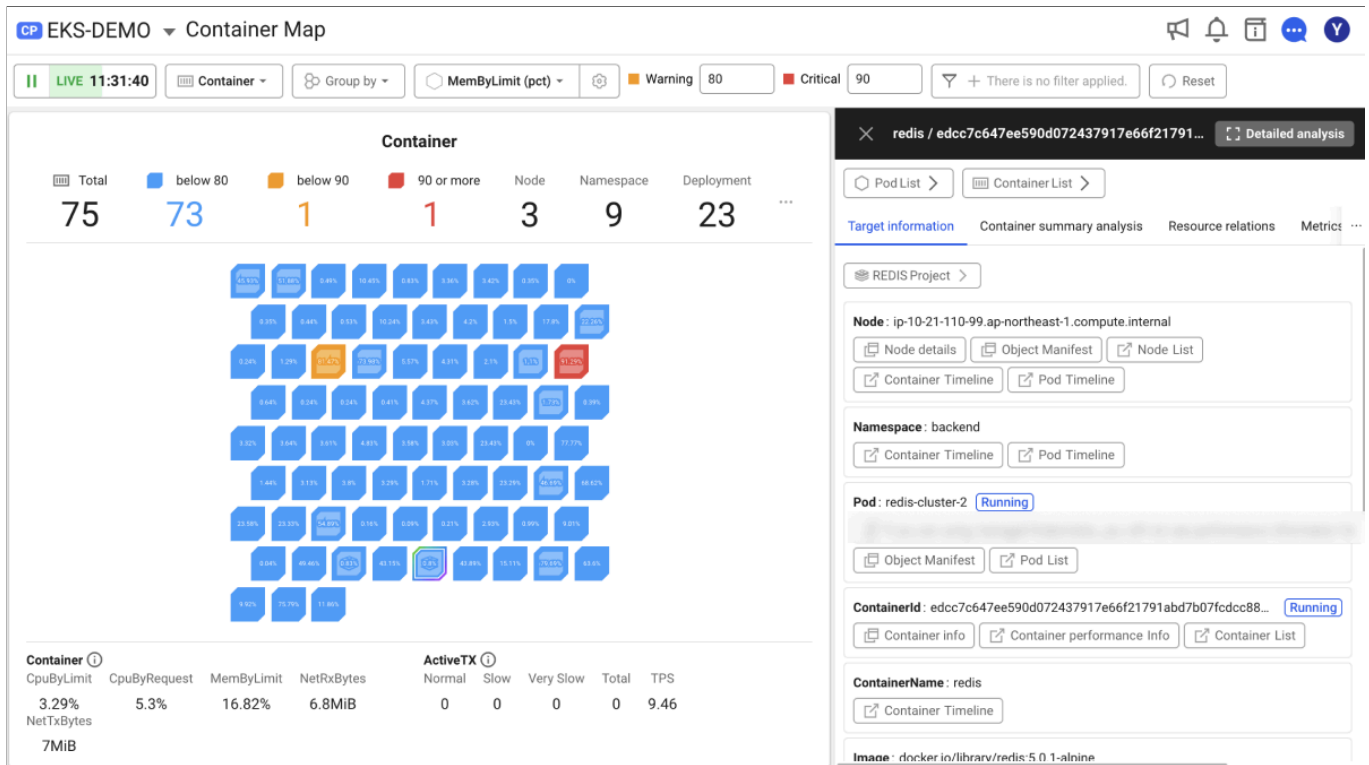
- **Call Information**

It provides a chart visualizing the call flow of the applications running in the selected container.

Based on these data, you can obtain the observability of the Kubernetes environment. Both the **Display Summary** and **Detailed Analysis** provide information based on the above 4 items.

Summary view

Click a specific object or a set of objects on the container map. You can check the summary data of the object through the **Display Summary** screen that appears on the right.



• Target Info

It provides basic information about the selected Pod or container. You can see basic target information such as the node on which the resource is running and the namespace it belongs to, and navigate to the list menu.

- **Node:** You can check [Node details](#), [Object manifest](#), [Node summary analysis](#) and move to the [Node list](#) menu.
- **Deployment:** You can check the [Object manifest](#) and go to the [Deployment list](#) menu.
- **ReplicaSet:** You can check the [Object manifest](#).
- **Pod:** You can check [Pod start analysis](#), [Object manifest](#), [Pod summary analysis](#) and move to the [Pod list](#) menu.
- **ContainerId:** You can check the [Container info](#) and [Container summary analysis](#) and go to the [Container list](#) menu.
- **Image:** You can check the [container image](#) and go to the [Container image](#) menu.
- **ServiceName:** You can check the [Object manifest](#) and go to the [Service list](#) menu.

- ❗ • **Volume:** You can see the volume and mount path used by the container.
 - You can see the name of the **a** volume used in the Pod and the type of the **b** volume.

e.g. **Volume** **a** containerddomainsocket **b** HostPath
 - When the **HostPath** volume type is used, you can see the file path of the **c** host system. This path connects the file system between the Pod and the host system.

e.g. **Host Path** **c** /run/containerd/containerd.sock

- **Resource relations**

The **Resource relationships** tab of the container map allows you to visually see the connections between Kubernetes resources.

- **Trace**

Under the **Trace** tab in the container map, you can trace each individual transactions of the applications running on the container.

- **Metrics**

Under the **Metrics** tab in the container map, you can see various metrics data such as resource quotas and usages for each container in real time.

- **Log**

Under the **Log** tab in the container map, you can manage and view standard logs for each container and application logs on the container.

- **Event**

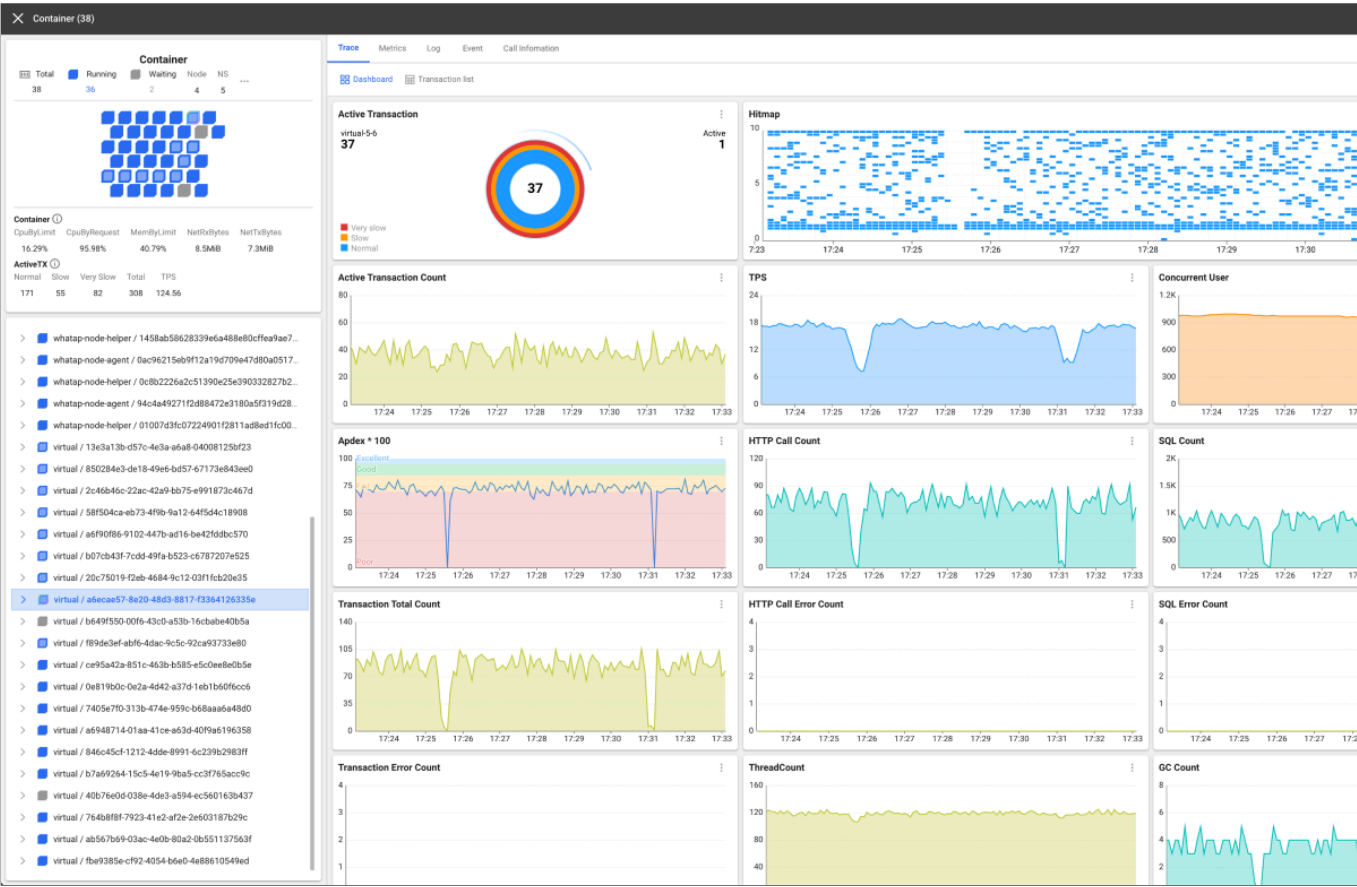
Under the **Event** tab in the container map, you can see integrated views for the events that occurred in the Kubernetes cluster and user threshold events based on the WhaTap's alert function.

- **Call Information**

In the **Call information** tab of the container map, you can see the call flow of the applications running on Kubernetes containers and the status of outbound call targets.

Detailed analysis

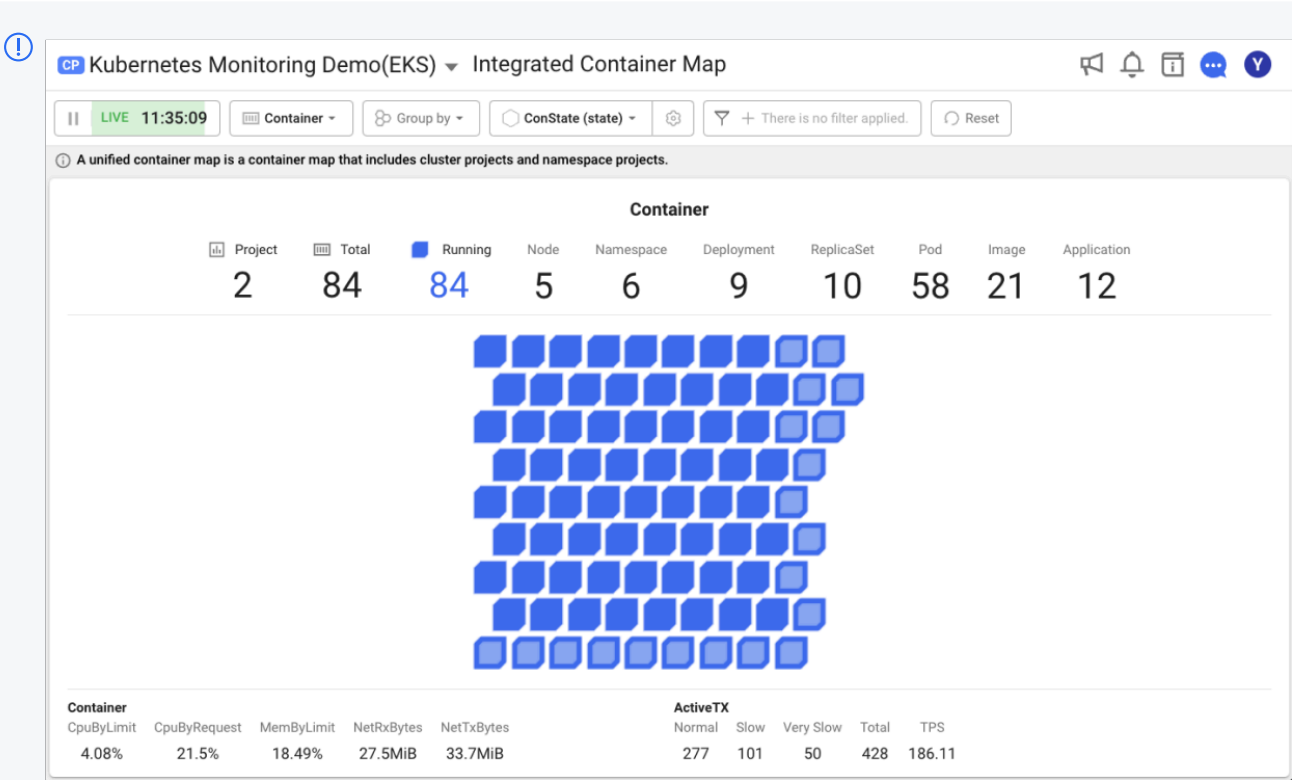
On the upper right of **Display Summary**, click  **Detailed analysis**. The **Detailed analysis** screen appears. For more information about the **Detailed analysis** screen layout, see [the following](#).



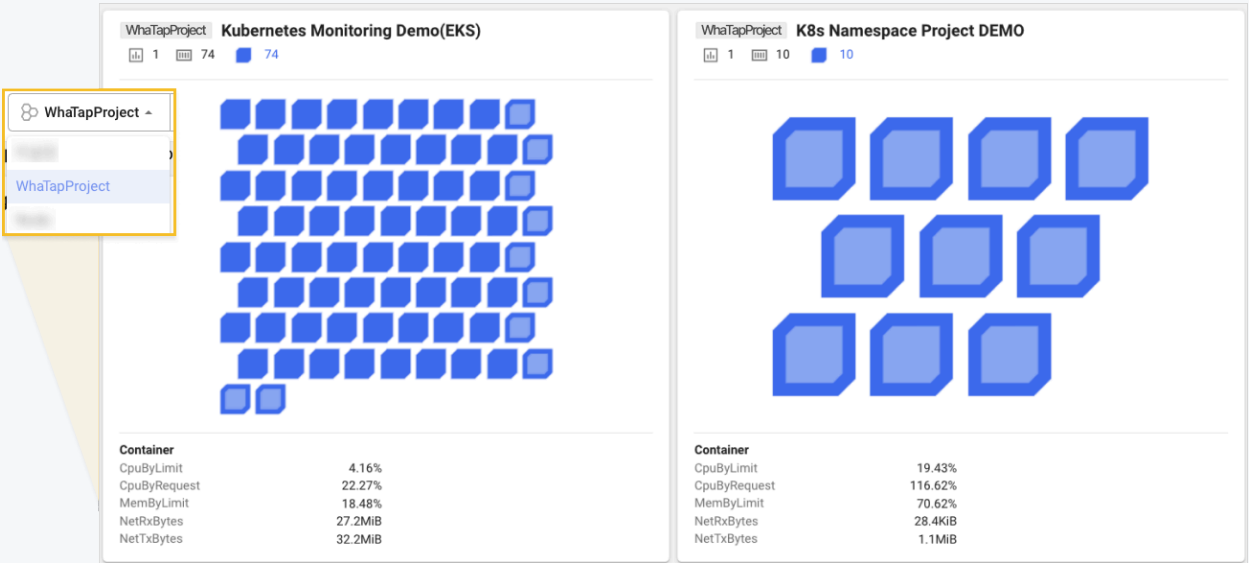
 Integrated Container Map

Home > **Cluster** > **Integrated Container Map**

To view the container map including the cluster project () and namespace project (), see the [Integrated Container Map](#).



Integrated Container Map provides additional [grouping](#) options by project. Among the [grouping](#) options, the [WhaTapProject](#) grouping allows you to view the container map in the cluster project and namespace's individual projects on a single screen as follows.



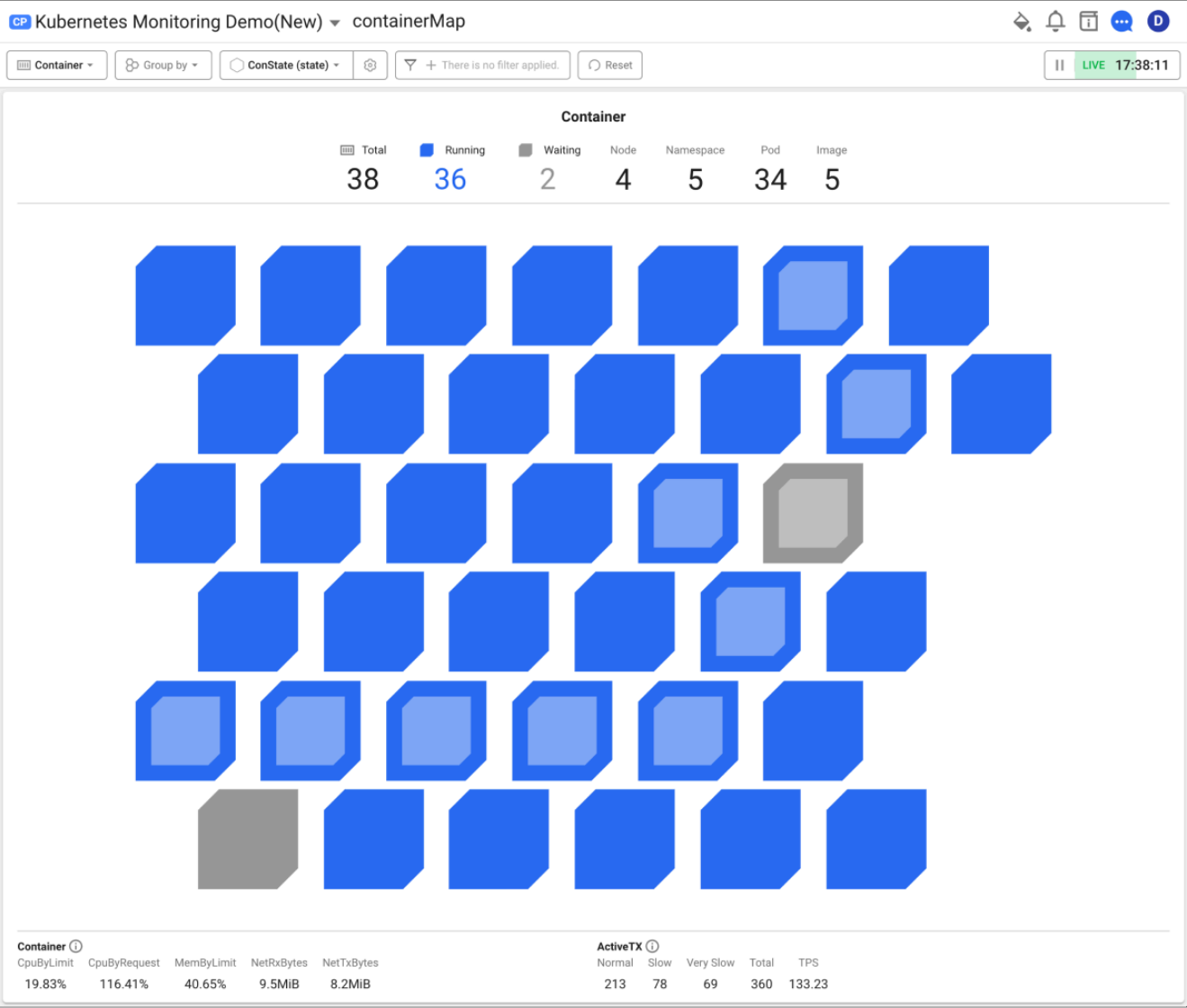
Container View, Pod View

Select Category (Container view, Pod view)

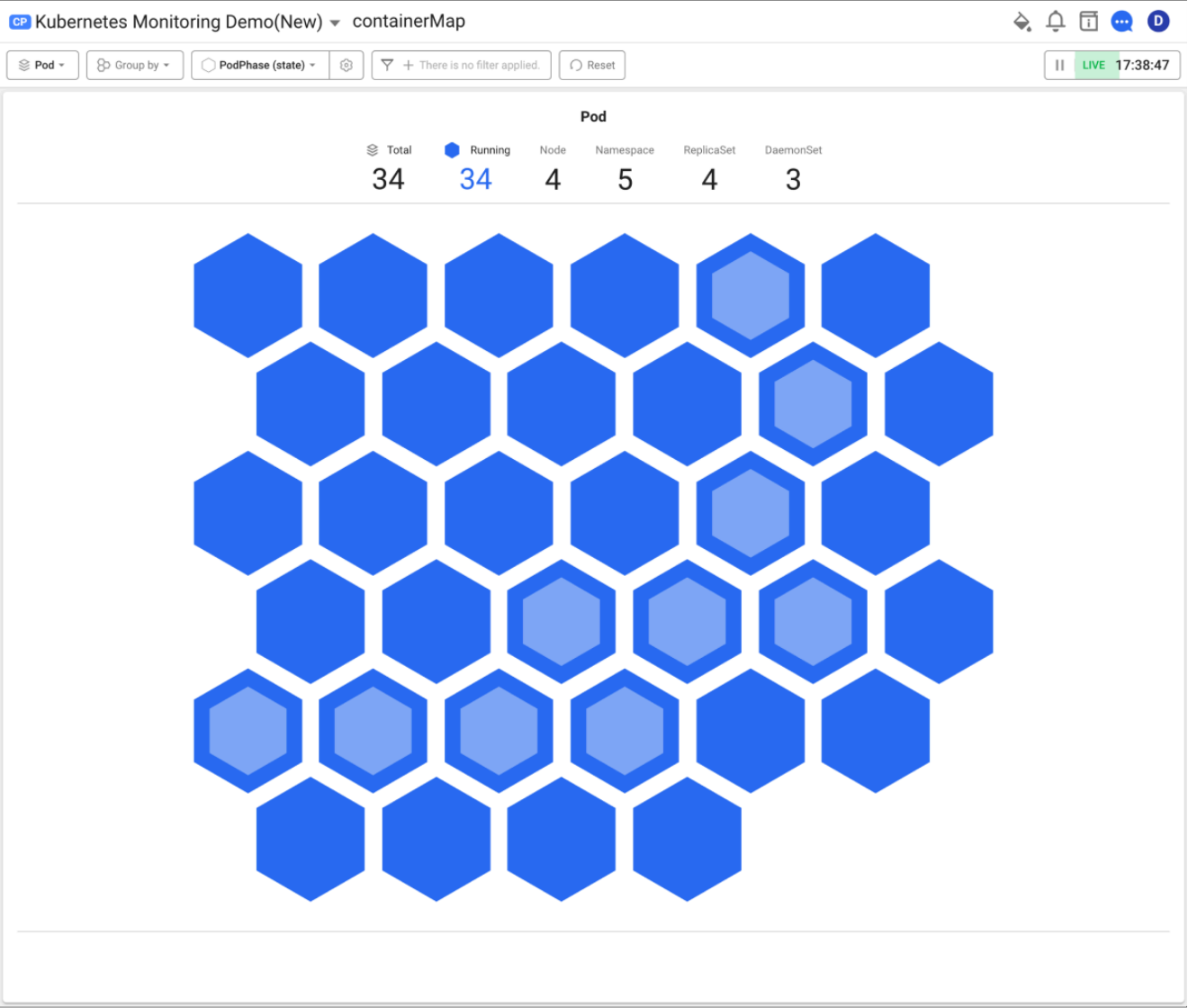
When you enter the [Container map](#) menu for the first time, you can see that all individual containers or pods in your cluster project have been expanded. Because multiple containers can run within a pod, you must have views for each container and pod.

In [Container Map](#) for WhaTap Kubernetes, both **Per-container view**, **Per-pod view** are supported, so condition-based monitoring is possible such as container-level monitoring and pod-level monitoring.

- **Per-container view**



- Per-pod view



Per-container view and **Per-pod view** can be switched depending on the user settings of the [Container Map](#) menu.

⚠ If you are using an older version of the Kubernetes agent, you cannot see the per-pod view. Upgrade the agent.

Master



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

Node

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

Target Grouping

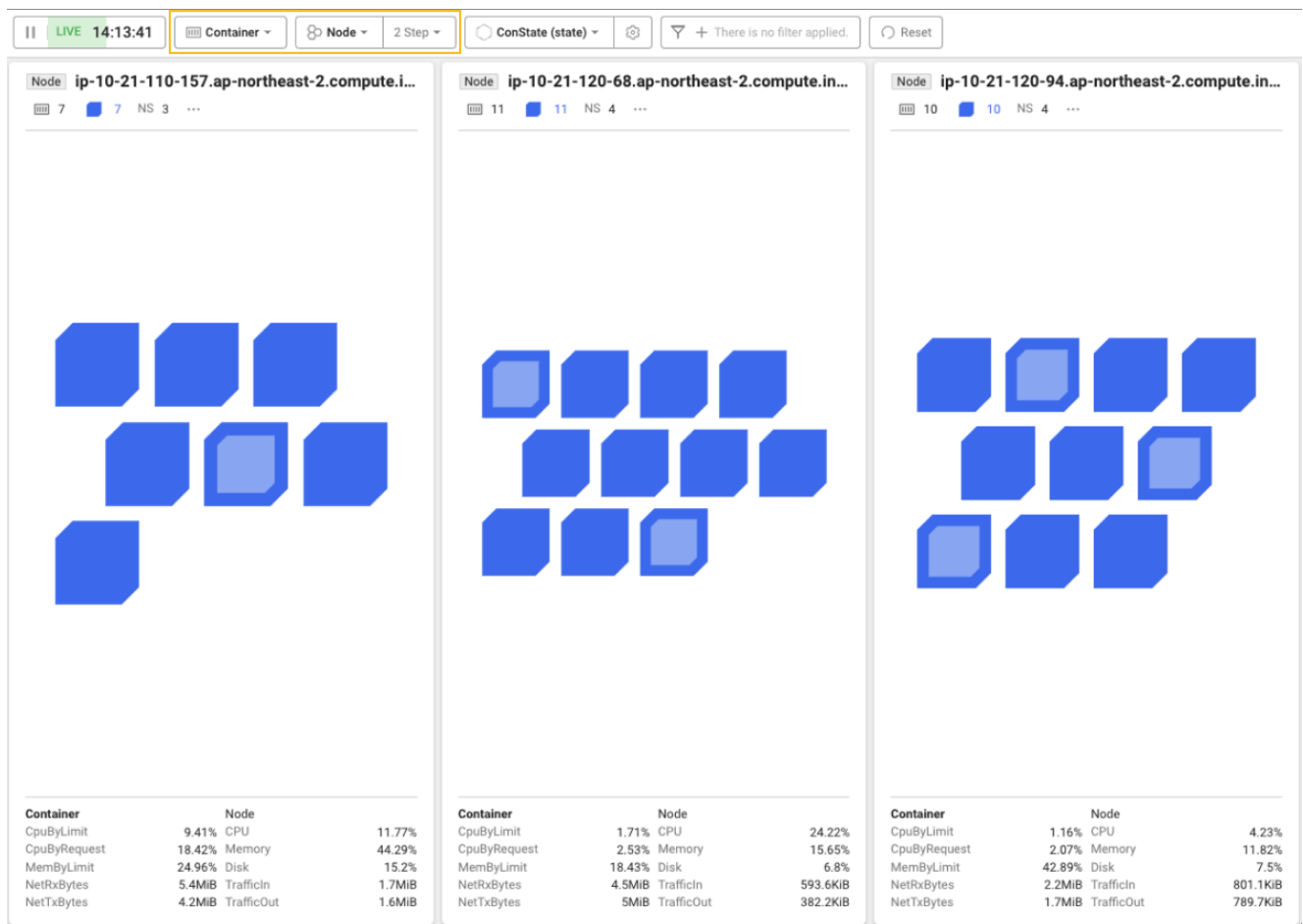
Various groupings by user interests

The screen of the default [Container Map](#) menu displays all containers or pods. However, when observing the containers and pods in Kubernetes, there are many cases in which targets are selected and monitored based on the specific criteria. Of course, the criteria for looking at the target are very diverse. This is because users' interests in the Kubernetes environment are not all the same.

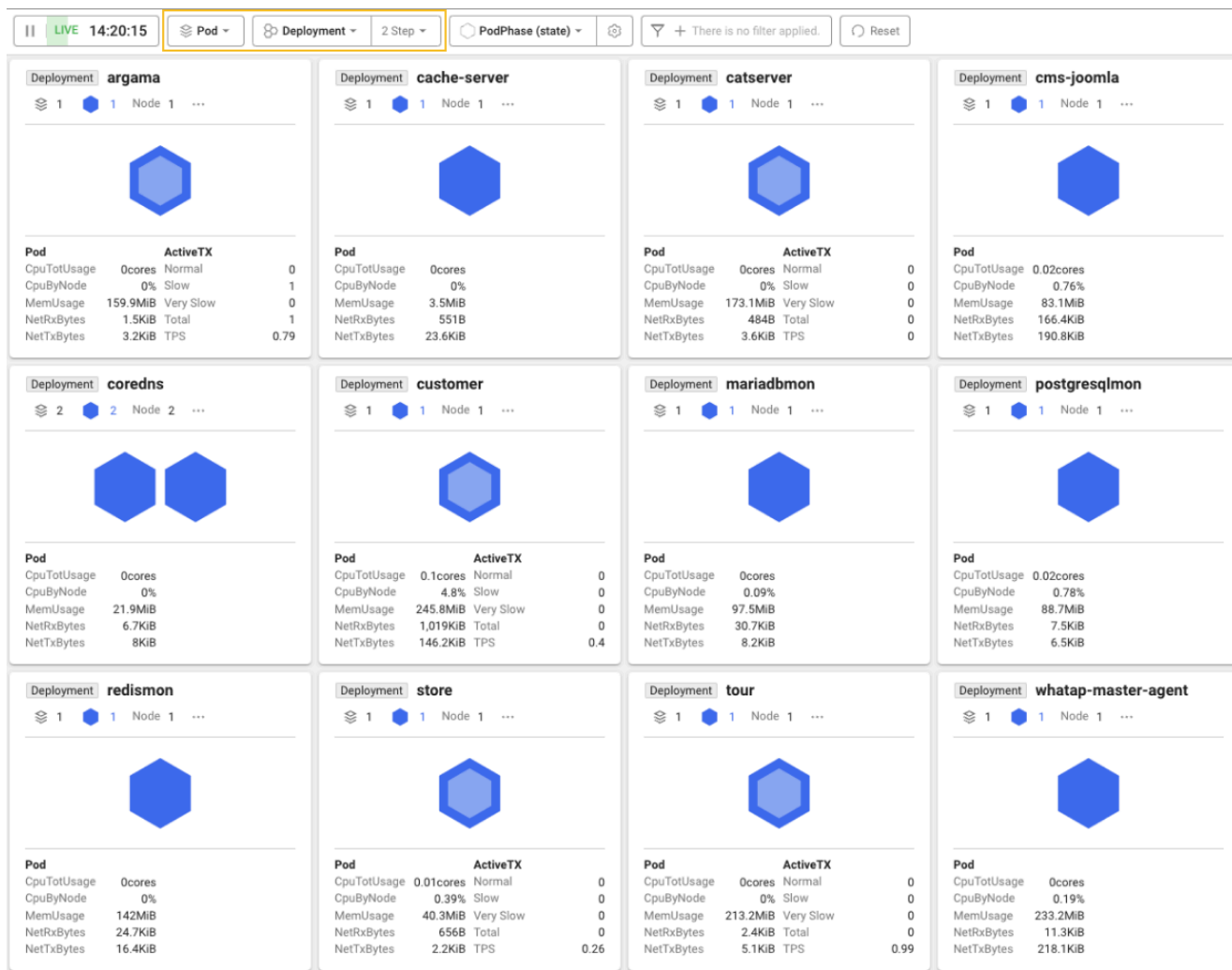
Server staff are interested in the node state for running Kubernetes resources. On the other hand, the application manager of a specific service is interested in the state of deployment, individual pods, or containers. After a new version release, you can monitor the overall status of deployment as well as the containers using a new image.

Therefore, **grouping** is provided to meet these diverse user needs.

- **Per-container view + Node grouping**



- Pod-based view + Deployment grouping



Key grouping properties

The grouping properties supported depending on the container map view are as follows:

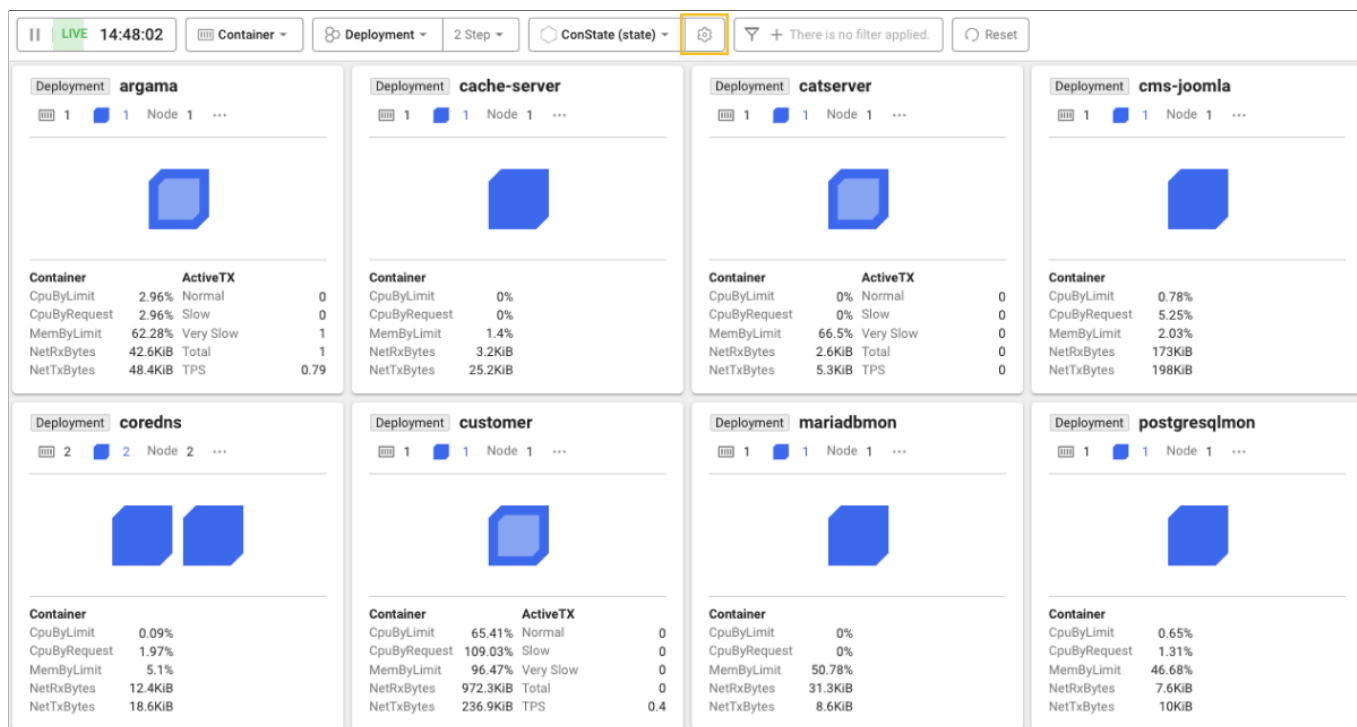
| Reference view | Grouping property |
|----------------|--|
| Container | Not set (All)
Node
Namespace
Deployment |

| Reference view | Grouping property |
|----------------|--|
| | ReplicaSet
ReplicationController
Pod
ContainerName
Image |
| Pod | Not set (All)
Node
Namespace
Deployment
ReplicaSet
ReplicationController
DemonSet
StatefulSet |

ⓘ The grouping settings (including the attribute fields and filtering options) on the screen are stored in the user browser. To initialize the stored settings, select  [Reset](#).

Grouping & Card

A container map consists of one or more cards. If grouping is not applied, all containers/pods in the project are represented in a single large card. Grouping organizes the cards into individual groups.



The above screen displays grouping based on the deployment as the reference property. You can see that the cards are listed for each deployment.

When grouping is applied, the objects that cannot be grouped by the reference property do not appear by default. However, there is a way to make those objects visible on the screen as well. On the screen select the  setting button. If [Hide unknown groups](#) is unchecked, only the objects that do not fall under grouping are collected and represented as separate cards.

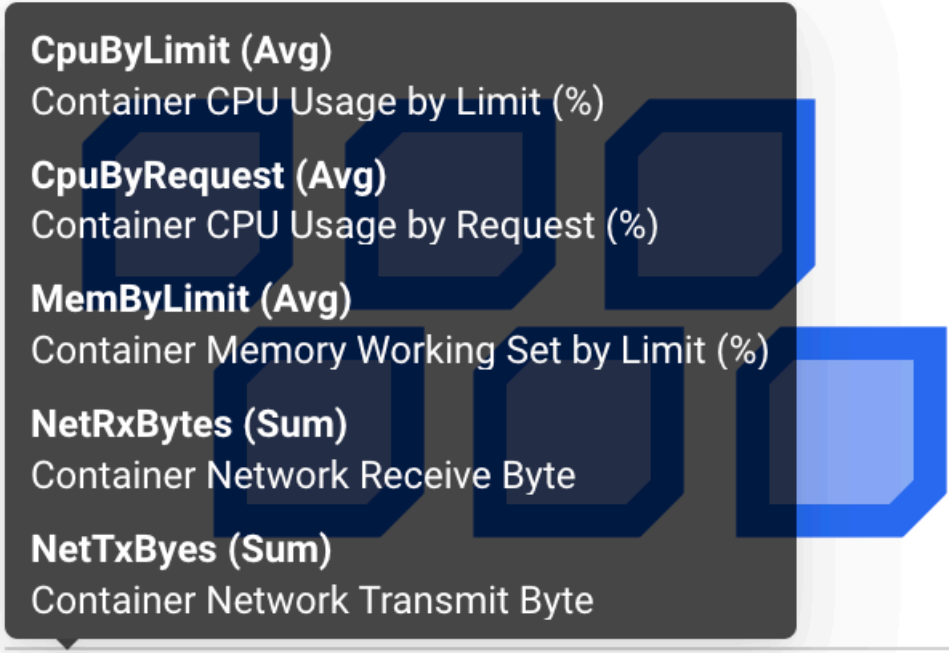
Metrics data by card

Metrics data for grouped targets appears under each card. The following lists the provided metrics:

| Reference view | Resource metric | Application metrics |
|----------------|--|--|
| Container | CpuByLimit
CpuByRequest
MemByLimit
NetRxBytes | Normal TX
Slow TX
Very Slow TX
Total TX |

| Reference view | Resource metric | Application metrics |
|----------------|-----------------|---------------------|
| | NetTxBytes | TPS |
| Pod | CpuTotUsage | |
| | CpuByNode | |
| | MemUsage | |
| | NetRxBytes | |
| | NetTxBytes | |

If the cursor is over the [container](#) (or [pod](#)) or [ActiveTX](#), you can check descriptions for each metric in a pop-up.



CpuByLimit (Avg)
Container CPU Usage by Limit (%)

CpuByRequest (Avg)
Container CPU Usage by Request (%)

MemByLimit (Avg)
Container Memory Working Set by Limit (%)

NetRxBytes (Sum)
Container Network Receive Byte

NetTxBytes (Sum)
Container Network Transmit Byte

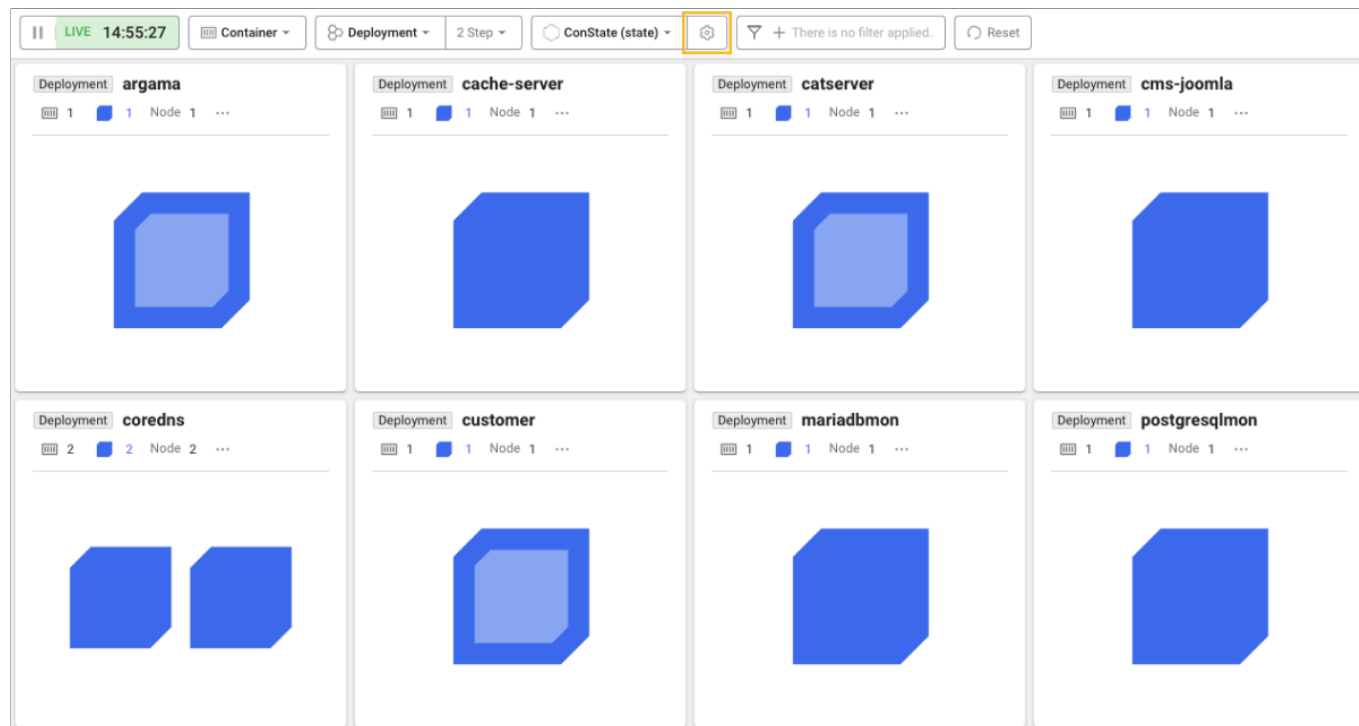
Container **ActiveTX**

- Among the metrics displayed in the pop-up, the items with Avg mean that the metrics values of the same names under the card are the average values of individual targets within the group.
- Among the metrics displayed in the pop-up, the items with Sum mean that the metrics values of the same names under the card are the sum of individual targets within the group.

Hiding the metrics data of the card

You can hide metrics data under the card and focus only on the chart. On the screen select the ⚙️ setting button. Uncheck [Show Metrics Below Card](#). This function is useful when you have a large number of cards.

You can hide metrics data as follows:

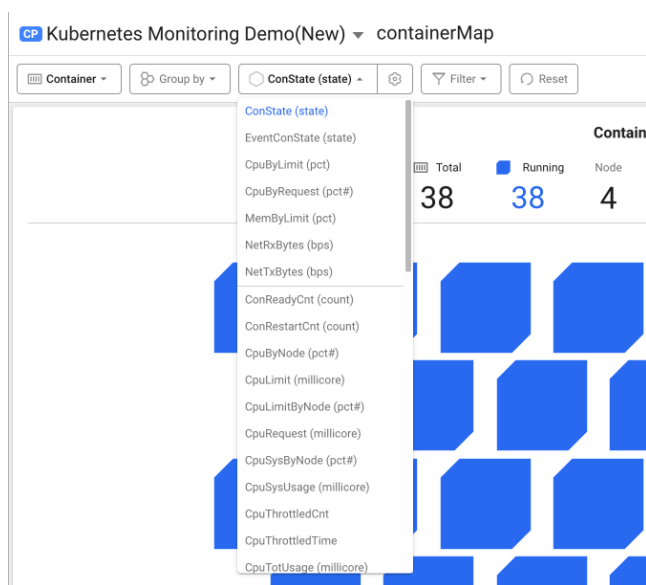


Container Map Metrics And Status

Customizing the chart metrics

The property of the default chart provided by the WhaTap Kubernetes monitoring service's [Container Map](#) menu is the state of container or pod. The metric name is [ConState](#) (Container State). However, it is required to construct the map based on the criteria other than the default state. For example, it may be required to determine the states of each target based on the CPU usage (millicore), memory usage (%), network usage (byte), and such rather than the default state.

WhaTap provides the function to change the container's chart metrics. Changeable chart metrics values are the same ones that WhaTap Kubernetes collects.

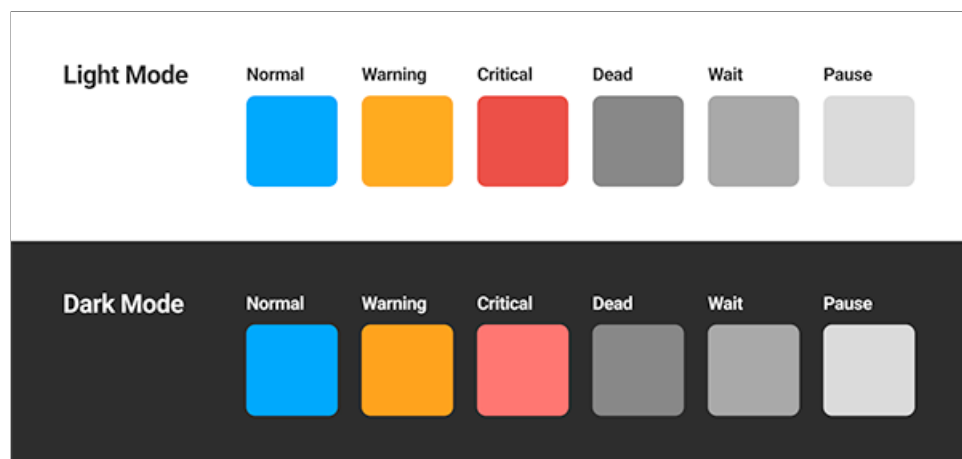


ⓘ The grouping settings (including the attribute fields and filtering options) on the screen are stored in the user browser. To initialize the stored settings, select [Reset](#).

Metrics threshold settings and state colors

The states of each target can be classified by color by setting the threshold values of the metrics given to the targets. Thresholds can be set when the units of metrics are for numbers (pct, bps, millicore).

The colors and meanings for states such as **Normal**, **Warning**, **Critical**, and such are as follows:



| Status | Description |
|-----------------|---|
| Normal | Displays the normal section (three divisions under Warning). |
| Warning | Indicates it is within the Warning threshold (more than Warning to Critical or more). |
| Critical | Indicates it is within the Critical threshold (more than Critical). |
| Dead | Container exit state |
| Wait | Container startup state |
| Pause | Container's suspended state |

Event-based container state

In [Container Map](#), the container view provides a state value called `EventConState`. `EventConState` is the event-based container state, which represents the container state where the event is reflected. The event types reflected to `EventConState` are as follows:

| Event | Level |
|------------------|-------------------|
| Kubernetes Event | Warning |
| WhaTap Event | Warning, Critical |

- `Kubernetes Event` is the event data provided by Kubernetes. It is the same as the event data that can be found in the user's Kubernetes environment through the `kubectrl get events` command.
- `WhaTap Event` is set through the WhaTap event function.

`EventConState` is determined by the following methods:

| Status | State value | Cause |
|----------|-------------|---|
| Critical | 50 | Critical event caused by the WhaTap microagent
Critical event caused by the WhaTap node agent
Toggle container state OOMKILLED |
| Warning | 30 | Warning event caused by Kubernetes
WhaTap microagent-caused Warning event
WhaTap node agent-caused Warning event
Container state RESTARTING transition |
| Dead | 23 | Transition of the container state to DEAD |
| Pause | 22 | Transition of the container state to PAUSE |
| Waiting | 21 | Transition of the container state to WAITING |

| Status | State value | Cause |
|--------|-------------|---|
| Normal | 10 | Transition of the container state to Normal |

Among the WhaTap events, there are events that have set **Additional notifications when the event state is resolved..** Internally, two types of events are distinguished as follows:

- Events that do not trace the state resolution: `stateless` events
- Events that set additional notifications when the state is resolved: `stateful` events

The `stateless` and `stateful` events generated by the container are used to determine the `EventConState` state as follows:

- `stateless` events: Reflects `stateless` events that occurred in the container within the last minute to the `EventConState` state.
- `stateful` events: Reflects `stateful` events from occurrence to termination to the `EventConState` state.

External Project Integration And Configuration

Home > Select Project > **Dashboard** > **Container Map**

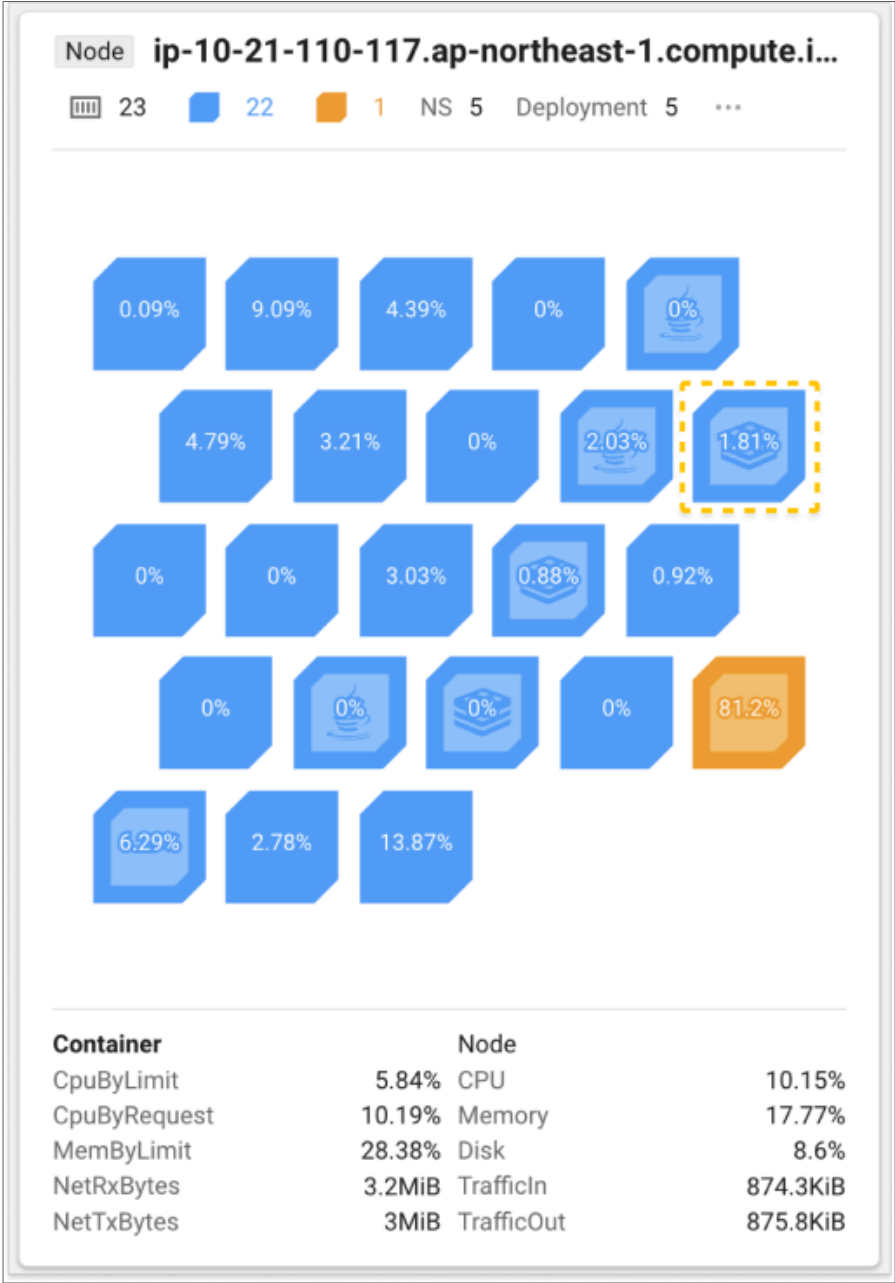
ⓘ The WhaTap Kubernetes agent 1.7.16 or later is required.

WhaTap provides various monitoring services on a project basis. The **External project integration** feature allows you to integrate external projects related to Pods in the Kubernetes projects.

To link to an external project, you need to configure the Kubernetes agent and a Pod to link to. Build an integrated monitoring scheme by linking the Kubernetes projects with external projects.

External project integration

When the external project integration feature is enabled, the project logo appears on the linked containers and Pods as follows. This makes it easy to visually see which external projects are linked. You can also navigate to the project via the button within the Summary View panel by clicking the block.



0.09%

9.09%

4.39%

0%

0%

4.79%

3.21%

0%

2.03%

1.81%

0%

0%

3.03%

0.88%

0.92%

0%

0%

0%

0%

81.2%

6.29%

2.78%

13.87%

Container

Node

CpuByLimit

5.84%

CPU

10.15%

CpuByRequest

10.19%

Memory

17.77%

MemByLimit

28.38%

Disk

8.6%

NetRxBytes

3.2MiB

TrafficIn

874.3KiB

NetTxBytes

3MiB

TrafficOut

875.8KiB

External project configuration

Agent CONFIG.

You must change the whatap-node-agent DaemonSet setting of the whatap-monitoring namespace in the [YAML](#) file for installation of the Kubernetes agent. In particular, you must add the following environment variables into the env section of the whatap-node-agent container.

```
- name: "collect_foreign_project"
  value: "true"
```

Example

Below is an example of the [YAML](#) file with environment variables added. Because there may be significant differences from the [YAML](#) file in the operation environment, see the example and then modify only the required parts before use. The highlighted lines are environment variables added.

```
apiVersion: "apps/v1"
kind: "DaemonSet"
metadata:
  labels:
    name: "whatap-node-agent"
    name: "whatap-node-agent"
    namespace: "whatap-monitoring"
spec:
  selector:
    matchLabels:
      name: "whatap-node-agent"
  template:
    metadata:
      labels:
        name: "whatap-node-agent"
    spec:
      containers:
        - command:
            - "/data/agent/node/cadvisor_helper"
```

```

- "-port"
- "6801"
env:
- name: "NODE_NAME"
  valueFrom:
    fieldRef:
      fieldPath: "spec.nodeName"
image: "whatap/kube_mon_dev:foreign-2"
name: "whatap-node-helper"
ports:
- containerPort: 6801
  name: "helperport"
resources:
  limits:
    cpu: "200m"
    memory: "350Mi"
  requests:
    cpu: "100m"
    memory: "100Mi"
volumeMounts:
- mountPath: "/rootfs"
  name: "rootfs"
  readOnly: true
- mountPath: "/sys"
  name: "hostsys"
  readOnly: true
- mountPath: "/dev/disk"
  name: "hostdiskdevice"
  readOnly: true
- mountPath: "/run/containerd/containerd.sock"
  name: "containerddomainsocket"
- command:
- "/bin/entrypoint.sh"
env:
- name: "NODE_IP"
  valueFrom:
    fieldRef:
      fieldPath: "status.hostIP"
- name: "NODE_NAME"
  valueFrom:
    fieldRef:

```

```

    fieldPath: "spec.nodeName"
  - name: "WHATAP_LICENSE"
    value: "x6050rs80f4eg-x1g0egm9rjb3mg-z7p9533a2ea87e"
  - name: "WHATAP_HOST"
    value: "13.124.11.223/13.209.172.35"
  - name: "WHATAP_PORT"
    value: "6600"
  - name: "HOST_PREFIX"
    value: "/rootfs"
  - name: "WHATP_MEM_LIMIT"
    valueFrom:
      resourceFieldRef:
        containerName: "whatap-node-agent"
        resource: "limits.memory"

```

```

- name: "collect_foreign_project"
  value: "true"

```

```

image: "whatap/kube_mon_dev:foreign-2"
name: "whatap-node-agent"
ports:
- containerPort: 6600
  name: "nodeport"
resources:
  limits:
    cpu: "200m"
    memory: "350Mi"
  requests:
    cpu: "100m"
    memory: "300Mi"
volumeMounts:
- mountPath: "/rootfs"
  name: "rootfs"
  readOnly: true
- mountPath: "/bin/entrypoint.sh"
  name: "start-script-volume"
  readOnly: true
  subPath: "entrypoint.sh"
- mountPath: "/whatap_conf"
  name: "whatap-config-volume"
initContainers:
- command:
  - "/data/agent/tools/whatap_debugger"

```

```

- "run"
image: "whatap/kube_mon_dev:foreign-2"
name: "whatap-node-debug"
volumeMounts:
- mountPath: "/rootfs"
  name: "rootfs"
  readOnly: true
serviceAccount: "whatap"
tolerations:
- effect: "NoSchedule"
  key: "node-role.kubernetes.io/master"
- effect: "NoSchedule"
  key: "node-role.kubernetes.io/control-plane"
volumes:
- hostPath:
  path: "/"
  name: "rootfs"
- hostPath:
  path: "/sys"
  name: "hostsys"
- hostPath:
  path: "/dev/disk"
  name: "hostdiskdevice"
- configMap:
  defaultMode: 448
  name: "node-start-script"
  name: "start-script-volume"
- emptyDir: {}
  name: "whatap-config-volume"
- hostPath:
  path: "/run/containerd/containerd.sock"
  name: "containerddomainsocket"

```

Configuring the Pod to be linked

Modify the settings of the Pod to link to. For example, in order to integrate the Redis Pod, add the following environment variables to the env section of the Redis container.

```

- name: "license"

```

```
value: "here_is_whatap_license"
- name: "whatap_platform"
value: "redis"
```

- `here_is_whatap_license` : Enter the license information of the external project.
- `whatap_platform` : Enter the platform of the external project to link to.

ⓘ The platforms that currently support integration with external projects are as follows: When entering on the platform, you can enter it without case sensitive.

Redis, Java, Node.js, Python, PHP

Example

Below is an example of the [YAML](#) file in which environment variables are added to the Redis Pod configuration. The highlighted lines are environment variables added.

```
apiVersion: apps/v1
kind: StatefulSet
metadata:
  name: redis-cluster
  namespace: backend
spec:
  serviceName: redis-cluster-svc
  replicas: 6
  selector:
    matchLabels:
      app: redis-cluster
  template:
    metadata:
      labels:
        app: redis-cluster
    spec:
      containers:
        - name: redis
          image: redis:5.0.1-alpine
          ports:
            - containerPort: 6379
```



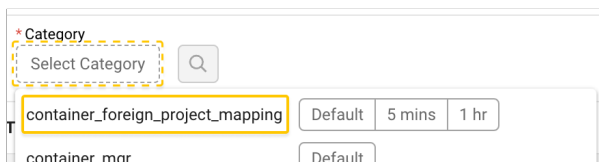
```

name: client
- containerPort: 16379
name: gossip
command: ["/conf/update-node.sh", "redis-server", "/conf/redis.conf"]
env:
- name: POD_IP
valueFrom:
fieldRef:
fieldPath: status.podIP
- name: license
value: x6050s081i709-z2lhh7gpi3bdr4-x7jovr61kumi5k
- name: whatap_platform
value: redis
volumeMounts:
- name: conf
mountPath: /conf
readOnly: false
- name: data
mountPath: /data
readOnly: false

```

Checking the external project to link to

Analysis > Metrics Search



After configuring an external project integration, go to the **Metrics Search** menu. Click [Select Category](#) as shown in the example image.

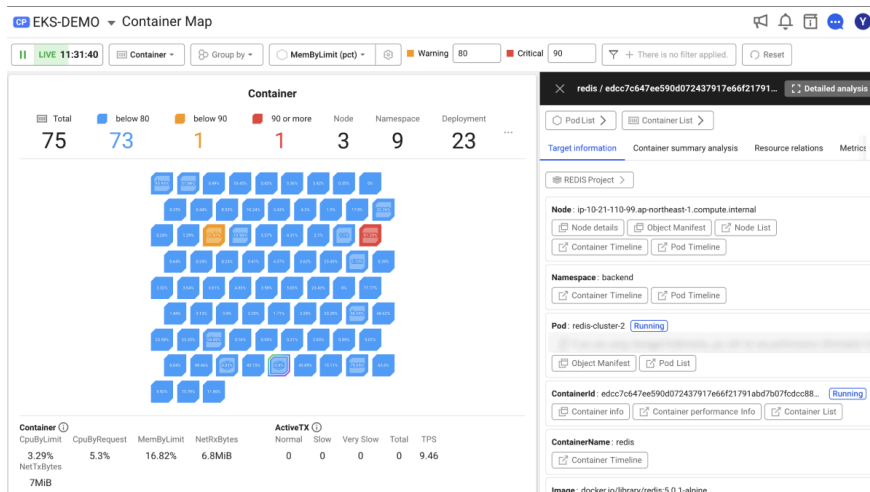
If the following category appears, it indicates that the integration has been made successfully.

```
container_foreign_project_mapping
```

Container Map Detailed Analysis

You can check detailed information about the target selected in the [Container Map](#) menu on the **Detailed analysis** screen.

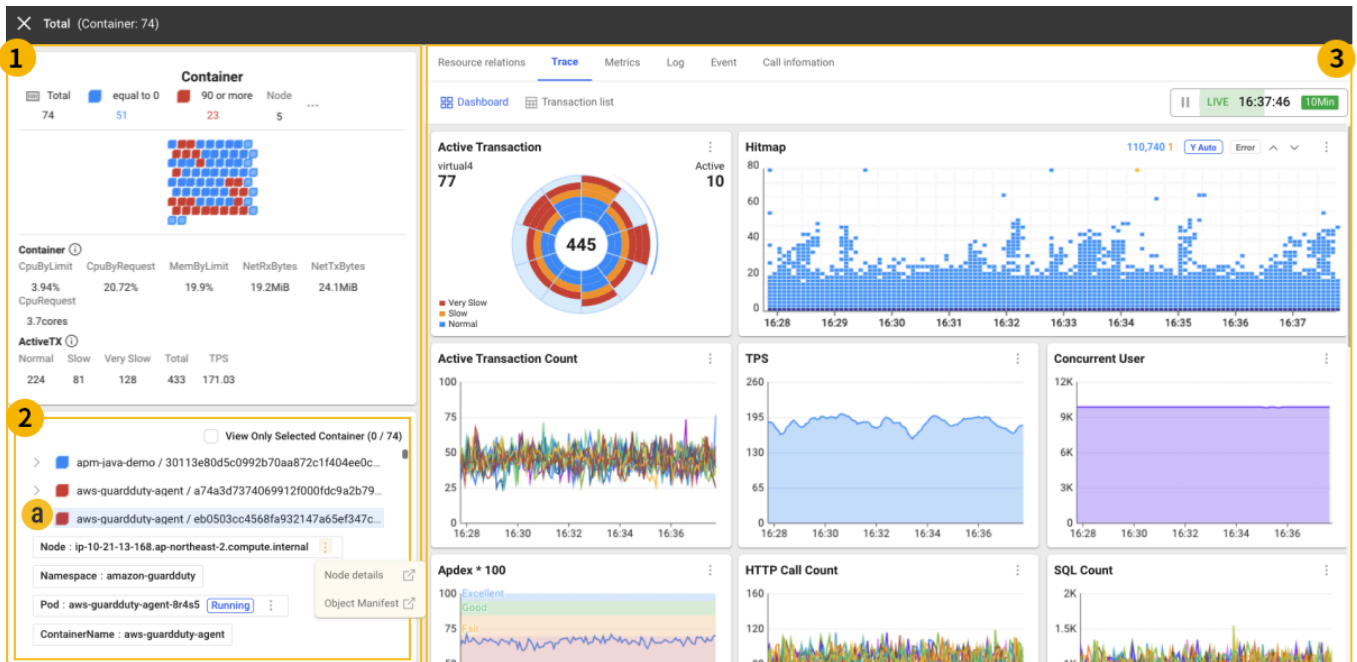
1. In the [Container map](#) menu, select a block (container or pod) to view.
2. On the right of the screen, the **View summarized** panel appears as follows.



3. If you select the **Detailed analysis** at the upper right of the **View Summary** panel, the furthermore detailed information appears.

Basic screen guide

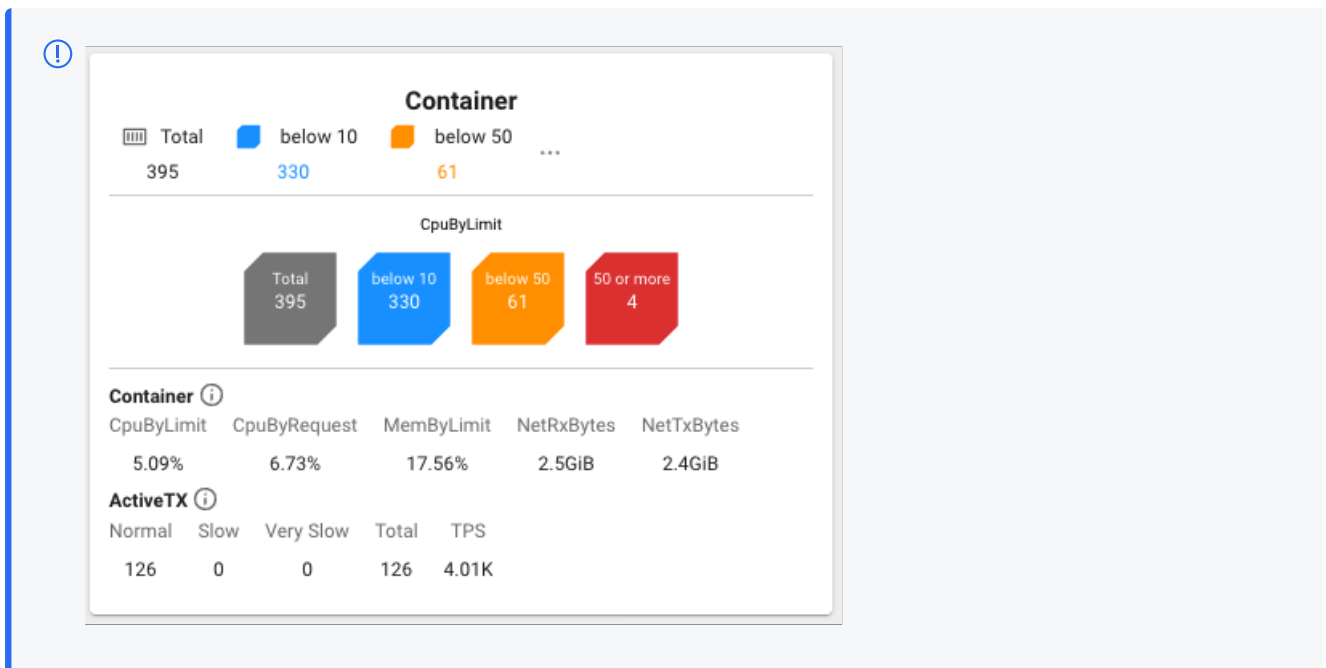
The **Detailed analysis** screen of the container map consists of the following. Basic information is available in the card and list formats. It provides [Trace](#), [Metrics](#), [Log](#), and [Event](#) tabs for key monitoring data.



- **1 Summary info card (container or pod)**

You can see the group of the selected target. When selecting a target additionally, you can check the metrics for the selected target in the **3** area. It has the same view as the card selected in the container map. For more information, see [the following](#).

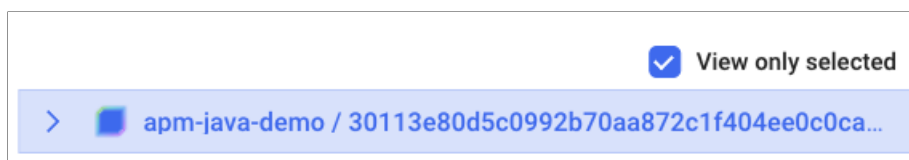
ⓘ In case of a large number of blocks when searching the summary cards, the notation method changes as follows. You can compare all blocks, **normal** blocks, **warning** blocks, and **fatal** blocks.



- **2 Target Info List**

You can see the group information of the selected target in a list format. When selecting a target additionally, you can see the metrics for the selected target in the **3** area.

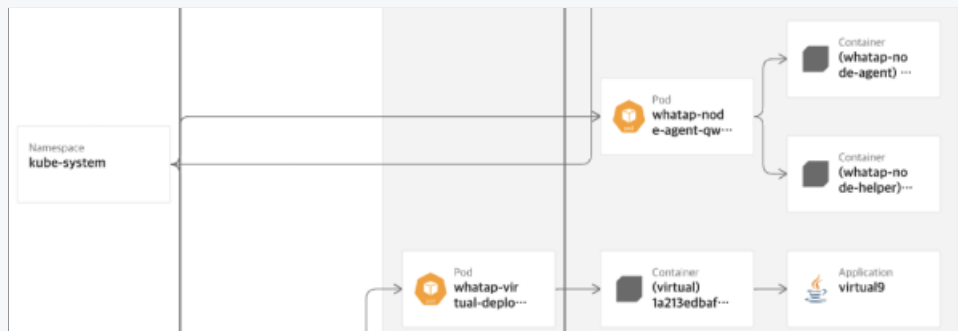
- If you check [View Only Selected](#) at the top of the **2** list, only the selected containers or Pods appear as follows:



- If you select [More](#) icon on the left of the list, the metrics for [Node](#), [Namespace](#), [Deployment](#), [ReplicaSet](#), [Pod](#), [ContainerName](#), [ContainerId](#), [Image](#), and [Agent](#) appear as shown in **a**. For more information, see [the following](#).
- **3 Resource relations, Trace, Metrics, Log, Event, Call Information**
 - [Resource relations](#)

The Resource relationships tab allows you to visually see the connections between Kubernetes resources.

- ❗ The relationship chart among [Namespace](#), [Deployment](#), [ReplicaSet\(DaemonSet\)](#), [Pod](#), [Container](#), and [Application](#) is provided as follows.



- [Trace](#)

In the [Trace](#) tab, you can monitor the transaction data for active applications on the Kubernetes container. It provides the views for [Dashboard](#) and [Transaction List](#). For more information, see [the following](#).

- [Metrics](#)

In the [Metrics](#) tab, you can check key metrics and resource usage patterns for the selected containers or pods. Through this, you can quickly identify resource issues and find optimization points. For more information, see [the following](#).

- [Log](#)

In the [Log](#) tab, you can see the logs generated from the Kubernetes containers and container applications. The logging functions are divided into the real-time container log searching and the past log searching. For more information, see [the following](#).

- [Event](#)

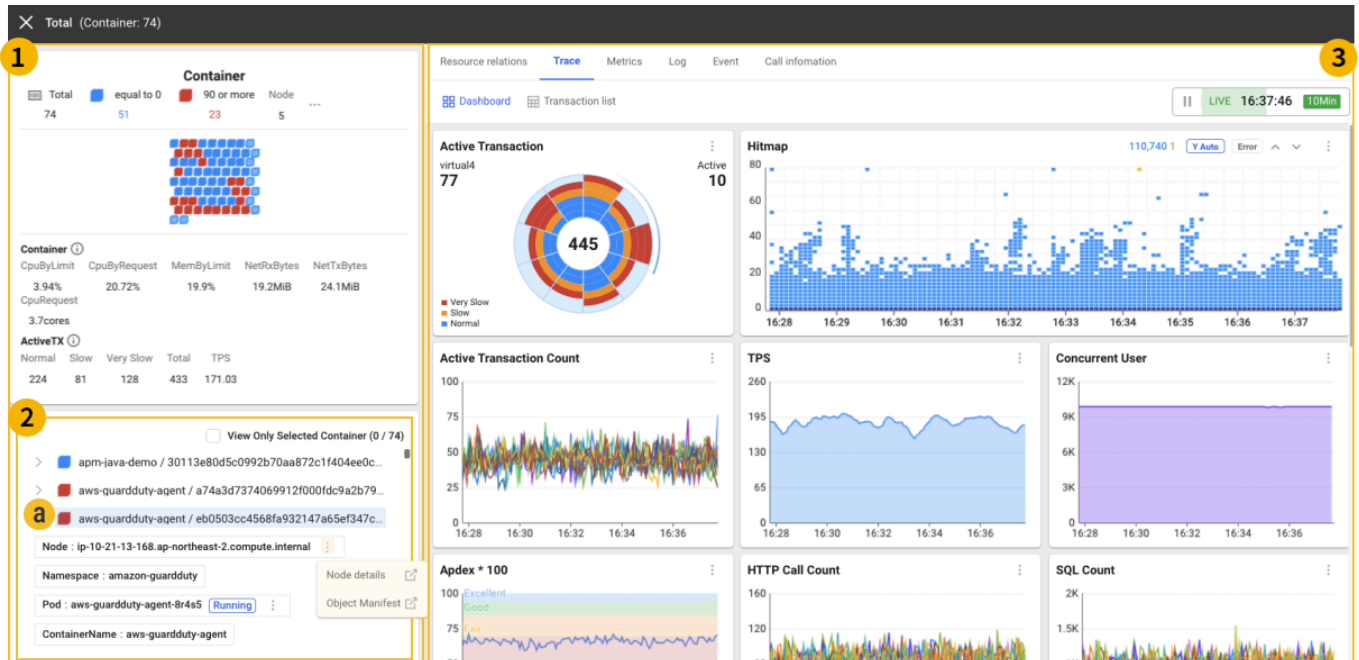
In the [Event](#) tab, you can see [Kubernetes](#) events that occur in a Kubernetes cluster and [WhaTap](#) events by the set conditions. For more information, see [the following](#).

- [Call Information](#)

In the [Call information](#) tab, you can see the call flow of the applications running on Kubernetes containers and the status of outbound call targets. For more information, see [the following](#).

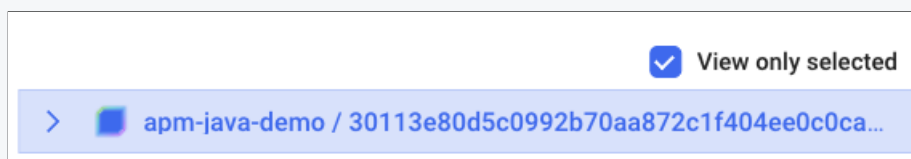
Target Information

Select Home screen > Select Project > **Dashboard** > **Container Map** > go to **Detailed analysis** > check the **target information list** on the left.



In the container view or pod view, a block represents each container or Pod. When you additionally select a target in the 1 **Summary information card**, you can see the information of the selected target in 2 **Target information list** and 3 through the Detail tab.

ⓘ If you select **View Only Selected** at the top of the 2 **Target information list**, only the selected containers or Pods appear as follows:



After selecting a block, you can check detailed target information such as the node on which the selected

container or pod is running, the affiliated namespace and object manifest through the **2 Target information** list within **Detailed analysis** or go to the list menu as shown under **a**.

- **Node:** You can check [Node details](#), [Object manifest](#), [Node summary analysis](#) and move to the [Node list](#) menu.
- **Deployment:** You can check the [Object manifest](#) and go to the [Deployment](#) list menu.
- **ReplicaSet:** You can check the [Object manifest](#).
- **Pod:** You can check [Pod start analysis](#), [Object manifest](#), [Pod summary analysis](#) and move to the [Pod list](#) menu.
- **ContainerId:** You can check the [Container info](#) and [Container summary analysis](#) and go to the [Container list](#) menu.
- **Image:** You can check the [container image](#) and go to the [Container image](#) menu.
- **ServiceName:** You can check the [Object manifest](#) and go to the [Service list](#) menu.
- **Volume:** You can see the volume and mount path used by the container.

- You can see the name of the **a** volume used in the Pod and the type of the **b** volume.

e.g. **Volume a** containerddomainsocket **b** HostPath

- When the [HostPath](#) volume type is used, you can see the file path of the **c** host system. This path connects the file system between the Pod and the host system.

e.g. **Host Path c** /run/containerd/containerd.sock

- ! • The volume type provided by WhaTap Kubernetes agent is as follows:

HostPath , ConfigMap , Projected , EmptyDir , PersistentVolumeClaim (PVC), Secret

- You can see the types of volumes supported by Kubernetes in the [Official Kubernetes documentation](#).

Kubernetes status summary analysis

It provides a summary analysis report on the performance and resources for each container, Pod, and node in the Kubernetes environment. You can monitor the resource allocation and usage for containers, Pods, and nodes in the clusters and namespaces in real time and respond effectively by referencing the suggestions of the

reports.

ⓘ The summary analysis of the current node is available only in Korean.

Container summary analysis

- **Basic information:** It provides information about the container's startup time, restart count, and the affiliated Pod and namespace.
- **Container image:** You can see how many containers are using the same container image as the one in use, and their total resource usage.
- **Node Information:** It provides hardware information about the node where the container is running, such as OS, IP address, number of CPU cores, and memory size.
- **Resource usage:** It provides the CPU and memory usage of the container and their utilization compared to the configured limit and request values.
- **Service networking info:** You can see the connected service and Ingress information.

Pod summary analysis

- **Basic Info:** You can check the Pod's cluster name, namespace, startup time, current state (Phase), and such.
- **Kubernetes Events:** You can check if any problem happened by looking at event logs related to the recent Pods.
- **Executor:** It identifies the controller (DaemonSet, Deployment, etc.) that created the Pod.
- **Node Info:** It provides detailed specifications of the node on which the Pod is running, including the operating system, kernel version, CPU, and memory.
- **Resource Usage:** It displays the CPU and memory usage for all containers in the Pod, and the utilization compared to the set resource limits.
- **Service networking info:** You can see the connected service and Ingress information.

Node summary analysis

- **Basic information:** It provides the cluster version of the node, startup time, current status, whether it is schedulable, internal IP address, region, and zone.

- Operating system and hardware information: It provides the operating system on which the node is running, OS image, CPU architecture, number of cores, and total memory size. You can check the allocatability and current usage of the memory and CPU of a node.
- Resource usage status: It displays the remaining resources through the difference between requests and allocatable resources (CPU, memory) for all containers running on the node. It summarizes the resource usage status, including the actual CPU and memory utilization.
- Disk usage status: It provides information about the disks attached to the node, file system types, device IDs, mount points, total and used capacity, and utilization.
- Pod status and resource availability: By comparing the total number of allocatable Pods on the node to the number of running Pods, you can see how many additional Pods can be run. In addition, the stability of the node is checked through the status of MemoryPressure, DiskPressure, PidPressure, and such.

Trace

Select Home > Select Project > [Dashboard](#) > [Container Map](#) > Target, and then **Detailed analysis** > [Trace](#).

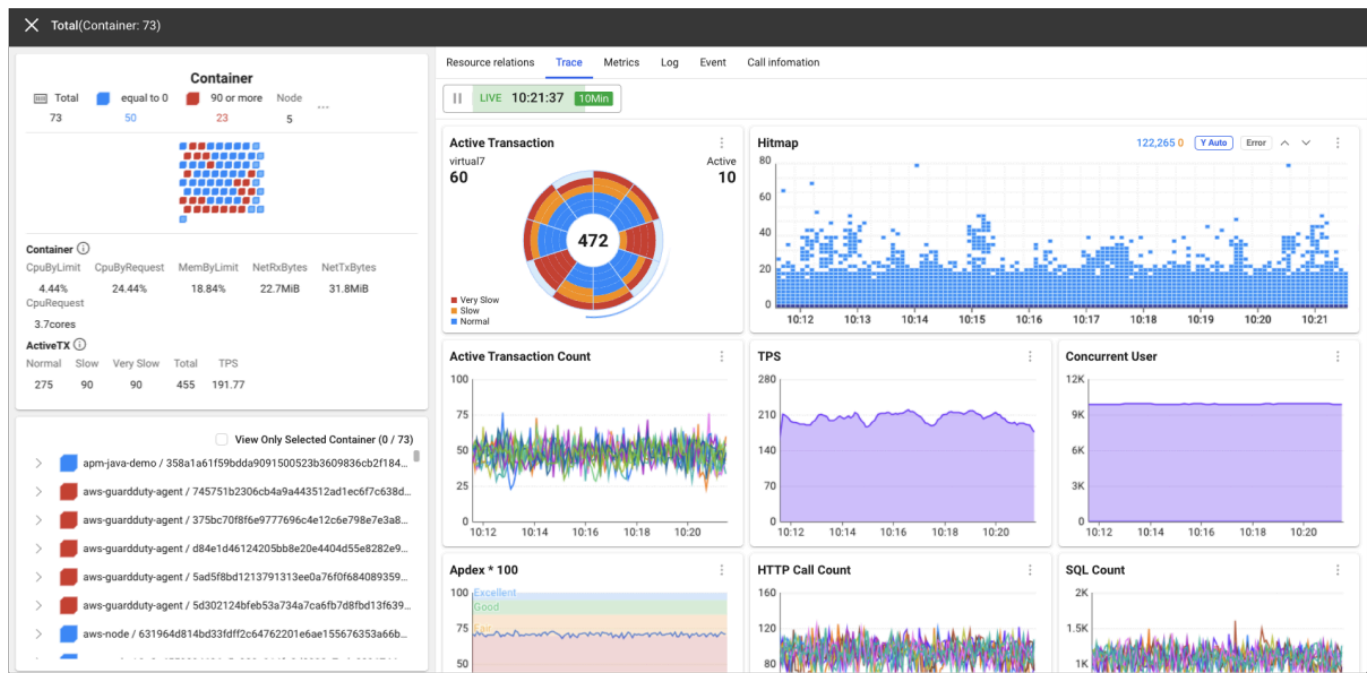
Containerized Application Trace

During Kubernetes monitoring, it is difficult to pinpoint the current status or cause of a failure by simply monitoring the cluster without tracing the applications in the Kubernetes environment.

When tracing the cause of a failure, it is often an application-level problem such as delay in SQL processing of a specific transaction or business logic error.

WhaTap Kubernetes trace function provides information about status monitoring for applications running on containers.

ⓘ To trace applications running on Kubernetes, the WhaTap monitoring agent must have been installed on each application container such as Java, Node.js, and Python. For more information about the agent installation, see [the following](#).



Dashboard

Dashboard provides key data for the following applications:

- **Active Transaction:** Current running transactions that are distinguished by color.
 - **Red (Very slow):** 8 seconds or more
 - **Orange (Slow):** 3 to 8 seconds
 - **Blue (Normal):** 0 to 3 seconds
- **Active Transaction Count**(Active Transaction Trend)
- **Hitmap:** Distribution of response times for completed transactions
 - Horizontal axis: Transaction end time
 - Vertical axis: Execution time
 - Transactions with longer execution times are at the top of the distribution.
 - Transaction data can be checked when dragging a specific area.
- **TPS** (Transaction Per Second): Number of transactions processed per second

- [Concurrent User](#): Concurrent users
- [Apdex * 100](#) (Application Performance Index): Measured user satisfaction for web applications
- [HTTP Call Count](#): Number of HTTP calls
- [HTTP Call Error Count](#): Number of HTTP call errors
- [SQL Count](#)
- [SQL Error Count](#)
- [Transaction Total Count](#): Number of all transactions
- [Transaction Error Count](#): Number of transaction errors
- [Response Time](#): Response time
- [Heap Memory Usage](#): Heap memory usage, available maximum/current memory
 - Real-time monitoring for changes in memory usage over time
 - Process total memory usage in case of Python
- [ThreadCount](#): Number of threads. Node.js and Python are excluded.
- [GC Count](#): Number of GCs. Node.js and Python are excluded.
- [GC Old Generation Count](#): Number of GCs in the Old area (Node.js and Python excluded)

Trace Analysis

You can check [Trace analysis](#) by dragging the desired section to view in [Hitmap](#) of the dashboard. After selecting a desired transaction from the transaction list, you can check the step information under each transaction at a glance.

ⓘ For more information about the trace analysis, see [the following](#).

Metrics

HomeProject > Select Project > **Dashboard** > **Container Map** > select a target and go to **Detailed analysis** > **Metrics**.

Kubernetes metrics

Kubernetes keeps an optimal state by continuously controlling and monitoring each container's resource usage. Accordingly, the process of checking resource usage and verifying the container resource usage setting based on that data is a basic pattern for Kubernetes monitoring.

CPU and memory metrics

Typical Kubernetes resource settings are the container's CPU request/limit and memory request/limit. The following is an example of configuration.

```
1  apiVersion: v1
2  kind: Pod
3  metadata:
4    name: <Pod name>
5  spec:
6    containers:
7      - name: <Container 1 person>
8        image: <Container 1 image>
9        resources:
10         requests:
11           memory: "64Mi"
12           cpu: "100m"
13         limits:
14           memory: "128Mi"
15           cpu: "200m"
16      - name: <Container 2 persons>
17        image: <Container 2 images>
18        resources:
19         requests:
```

```

20     memory: "128Mi"
21     cpu: "250m"
22     limits:
23     memory: "256Mi"
24     cpu: "500m"

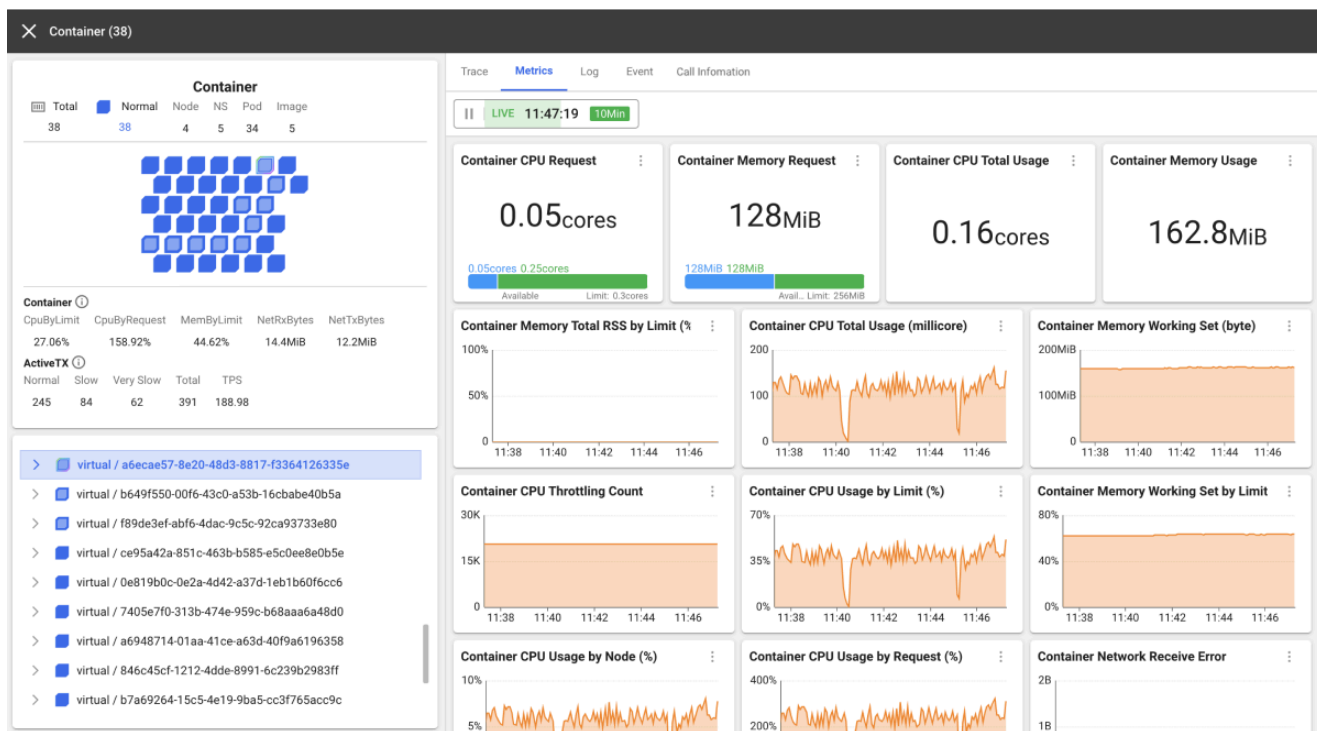
```

With the above configuration, there are 2 containers in the pod. The resource settings for each container are as follows:

| Target container | Resource type | Resource setting | Value |
|------------------|---------------|------------------|--------|
| Container 1 | CPU | Request | 100 m |
| | | Limit | 200 m |
| | Memory | Request | 64 Mi |
| | | Limit | 128 Mi |
| Container 2 | CPU | Request | 250 m |
| | | Limit | 500 m |
| | Memory | Request | 128 Mi |
| | | Limit | 256 Mi |

It is not easy to understand the resource configuration data and the current usage of various containers. Therefore, the WhaTap Kubernetes monitoring service provides the function to easily check the resource settings and current usage for a single container, selected multiple containers, or the overall deployment.

- **Detailed View** screen > **Metrics** - Resource usage monitoring and analysis



- You can select a specific container to check its status.



According to the above data, the CPU is set to Request 0.2 cores (200 m) and Limit 0.4 cores (400 m). The current CPU usage is 0.02 cores (20 m) (average for 5 seconds). If you check the CPU-related trend graph on the right, you can see that the CPU usage increases intermittently, although there is enough space.

The memory is set to Request 128 MiB and Limit 256 MiB. By the way, the current memory usage is 184 MiB. According to the memory-related trend graph on the right, it exceeded 100% on a request basis and is using more than 70% on a limit basis.

The main WhaTap metrics related to CPU and memory are as follows:

| Target resource | Shortname | Name | Description |
|-----------------|----------------|--|---|
| CPU | CpuTotUsage | Container CPU Usage (millicore) | Container CPU utilization |
| | CpuByRequest | Container CPU Usage by Request (%) | Container CPU utilization by request |
| | CpuByLimit | Container CPU Usage by Limit (%) | Container CPU utilization by limit |
| Memory | MemWs | Container Memory Working Set (byte) | Container memory working set |
| | MemWsByRequest | Container Memory Working Set by Request (%) | Container memory working set usage by request |
| | MemWsByLimit | Container Memory Working Set by Limit (%) | Container memory working set usage by quota |

⚠ If the node on which the pod is running has sufficient resources, it can use more resources than the request set in the container. Accordingly, among WhaTap metrics, the values for **Container CPU Usage by Request (%)** and **Container Memory Working Set by Request (%)** cannot exceed 100%.

For example, let's assume that the container's memory request has been set to 128 MiB. If the node of the active container (pod) has the total memory of 4 GiB and there has no pod on that node, this container can use more than 128 MiB of memory.

Other metrics

WhaTap provides various metrics other than CPU and memory.

- **Container Block I/O Read Byte** (IoReadBytes)
- **Container Block I/O Write Byte** (IoWriteBytes)
- **Container Network Receive Byte** (NetRxBytes)

- **Container Network Transmit Byte** (NetTxBytes)

For more information about the WhaTap Kubernetes metrics, see [the following](#).

Log

Home > Select Project > [Dashboard](#) > [Container Map](#) > Target selection, **Detailed analysis** > [Log](#)

Kubernetes logs

WhaTap Kubernetes monitoring service collects and stores the logs generated in Kubernetes. In Kubernetes, there are two different types of logs:

- Standard out logs for each container
- Logs output by applications running inside each container

These logs serve to compensate for monitoring blind spots that are difficult to check with the container status or metrics data. You can check these logs as follows:

- Standard Out logs of each container can be checked through the `kubectl logs` command provided by Kubernetes without entering into the container.
- Application logs in each container must be checked in the container or for that purpose, a separate log collection method must be devised.

Recent log messages (Container Stdout/Stderr)

This function allows you to immediately check the current container logs. You can fetch logs directly from your cluster environment to view real-time logs for the selected containers or applications within the selected pods. To check the container logs, the following command is used.

```
kubectl logs pod/<Pod name> -n <Namespace name>
```

If there are multiple containers in a pod, select each individual container to view logs.

```
kubectl logs pod/<Pod name> -c <container name> -n <Namespace name>
```

It is very inconvenient to access the terminal screen and run the command line tools every time to check logs. By using the [Log](#) function of WhaTap Kubernetes Monitoring, you can conveniently check the current logs.

The screenshot displays the 'Container (38)' interface. On the left, there's a 'Container' summary section with a grid of blue squares representing pods. Below it, a table shows resource usage for 'Container 1':

| CpuByLimit | CpuByRequest | MemByLimit | NetRxBytes | NetTxBytes |
|------------|--------------|------------|------------|------------|
| 25.17% | 147.01% | 44.31% | 15.4MiB | 13.9MiB |

Below the table, there's an 'ActiveTX' section with a table showing transaction status:

| Normal | Slow | Very Slow | Total | TPS |
|--------|------|-----------|-------|--------|
| 261 | 95 | 47 | 403 | 184.42 |

On the right, the 'Log' tab is active, showing a list of containers and their logs. The selected container is 'kube-flannel / 6a48da30-8602-4170-8cde-562d0fe25d53'. The log content shows various system messages and network configurations.

However, there are things to think about. If you exit the container, you cannot see logs any longer. Even if the container is still running, you can check only the logs for a specific period or amount. The WhaTap's Recent Log Messages ([Container Stdout/Stderr](#)) function allows you to see the logs in the Kubernetes environment. The disappeared logs from the user environment are difficult to check even through WhaTap.

Therefore, a separate method is required for stable and continuous management of log data. In particular, as the microservice architecture spreads and the number of containers for businesses increases, the log checking and management methods are becoming more complex.

Any good idea?

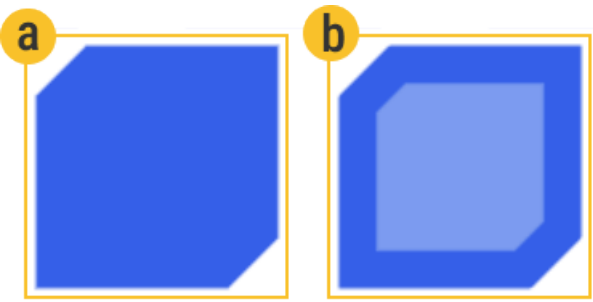
Log Search

All the related logs disappear when the container is shut down, but you can check the logs that have occurred on the selected container or pod through [Log search](#). When storing the container Standard Out logs, you can set the storage period and check information for specific targets or perform correlation analysis.

The screenshot displays the 'Log' tab of a container management interface. On the left, a 'Container' summary shows resource usage for 80 containers. The main area features a 'Container Stdout/Stderr' search bar with a time selector (1) set to '2025/01/20 19:39 ~ 2025/01/21 19:39 1 day'. Below the search bar, there are toggle switches for 'Display Tag' (2) and 'Display personal information'. A dropdown menu for 'containerStdout' is also visible. The log table (3) lists entries with columns for Timestamp, Log, and Log. The log entries show Redis cluster status updates and connection attempts.

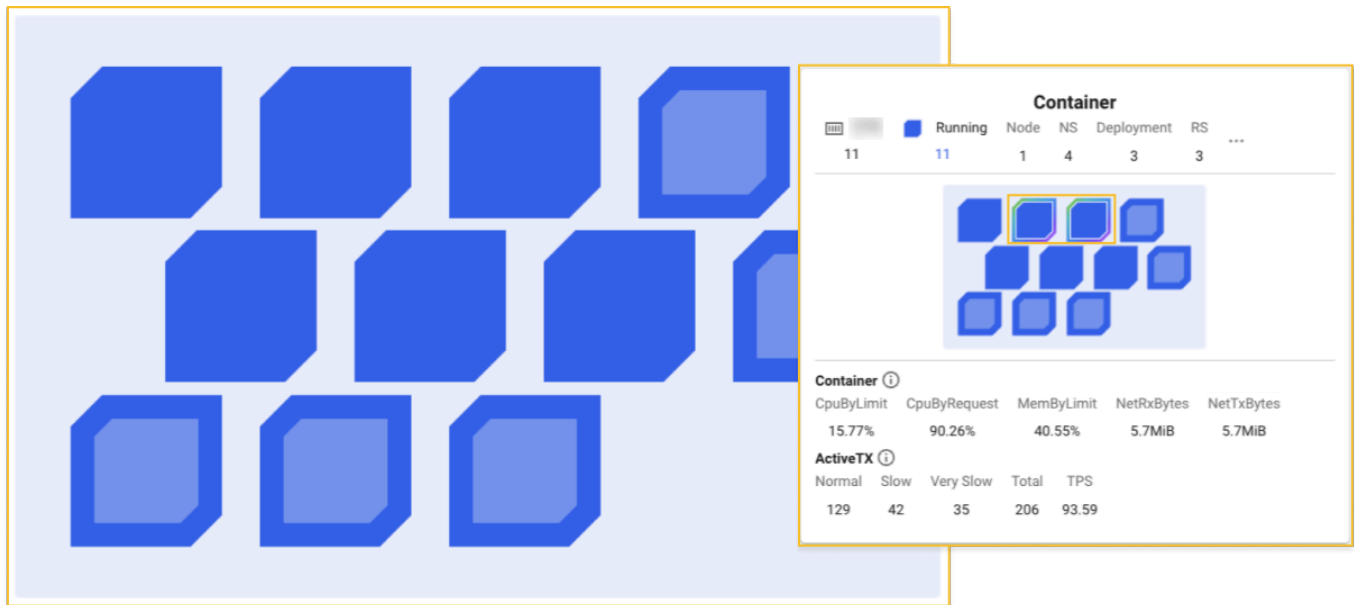
- Through the ① time selector, you can select a desired time for searching data.
- By specifying the ② category through the options, you can search data. For more information about the categories to view, see the following guide to [Bock](#) and [Group](#).
- If you disable the ② **Display tag** toggle, the **timestamp** and **message** can be checked excluding tags from the log records.
- If you enable the ② **Display personal information** toggle, you can see the original version of the masked data. For more information about the **De-identification of personal information** feature, see [the following](#).
- Click ② **Log search**. Then **Log > Log search** appears where you can search in detail through the functions such as Filtering.
- If you select ③ ▶ **More**, the parsed key values and log messages can be displayed. For more information, see [the following](#).

General blocks and micro blocks



- a General block**
A single polygon in the container map is called a block. In the `containerStdout` category of the log, you can see the information whose `containerid` matches.
- b Micro block**
Blocks with agents installed are called microblocks. In addition to the `containerStdout` category, you can also see `App*` related categories such as `AppLog` , `AppStdOut` , and `AppStdErr` . If you select the category, you can see the logs that match `microOid` (agent ID in the block).

Group



When all blocks are selected on the [Container map](#) screen or multiple blocks are selected from the card on the **View Detailed** screen, the multiple blocks are called a group. When selecting a group, you can see all the logs in the `containerStdout` and `App*` categories in the [Log search](#) tab.

Adding the node agent settings to save Standard Out

To save the container's standard outputs through the WhaTap log service, set the `logsink_enabled=true` option in the WhaTap Kubernetes node agent. For more information about the settings and the WhaTap log monitoring service, see [the following](#).

Adding the application agent settings for collecting application logs inside containers

To collect the logs of applications running on containers through the WhaTap log service, see [the following](#). Java, Python, and Go are supported.

⚠ If you enable the WhaTap log service and use the container's standard output log storage function, an additional fee is incurred. For more information about pricing, see [the following](#).

ⓘ Some older versions of the agent do not provide the function to save container standard output logs. The agent upgrade method is as follows:

Agent update

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

The latest version of the WhaTap Kubernetes agent can be checked in [the following](#).

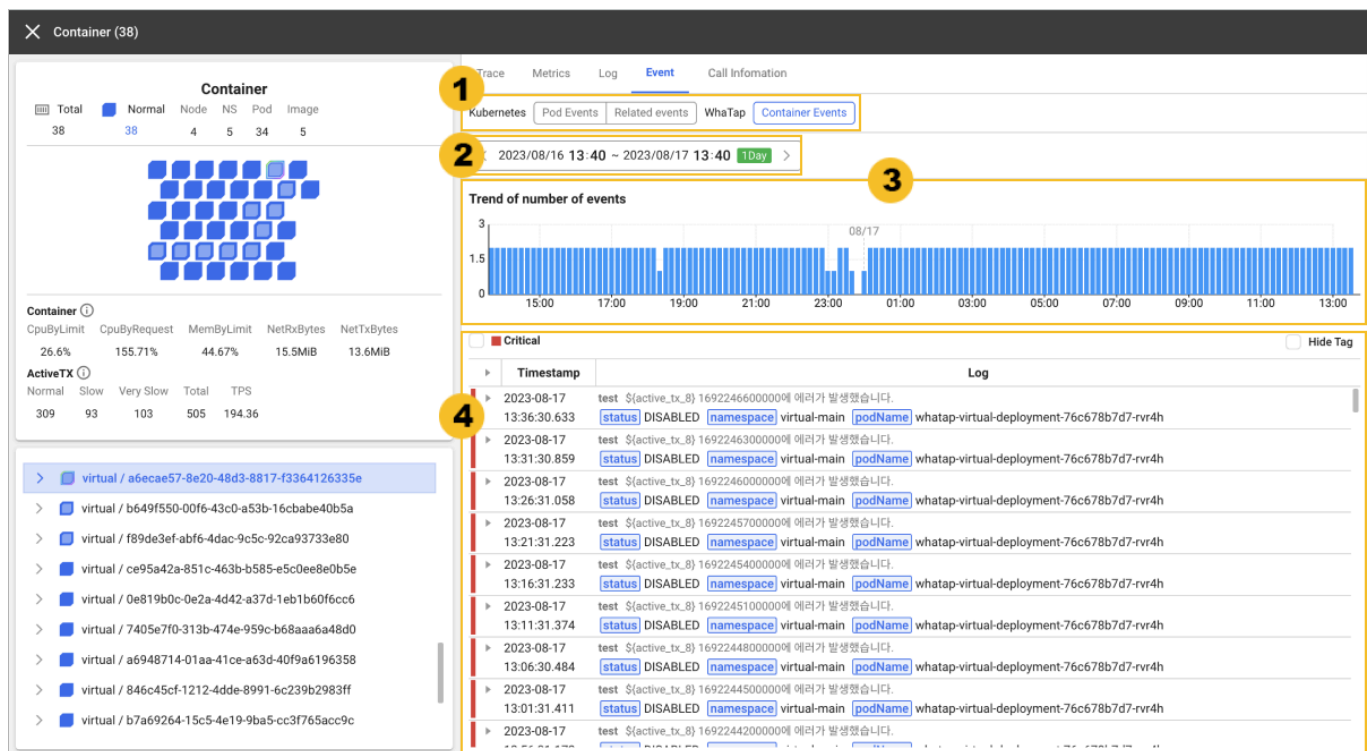
Event

Select Home > Select Project > [Dashboard](#) > [Container Map](#) > Target, and then in **Detailed analysis**, select the [Event](#) tab.

It provides records for all events that occur during the resource lifecycle. The events provided in WhaTap Kubernetes Monitoring are as follows.

- Kubernetes Event ([Kubernetes](#))
- WhaTap event ([WhaTap](#))

Basic screen guide



- You can select a ① Kubernetes event ([Kubernetes](#)) or WhaTap event ([WhatTap](#)) for query.
 - Selecting all or multiple blocks

You can search [All Kubernetes events](#) or [All WhatTap events](#) when all or multiple blocks are selected from [Container Map](#) and the card on the left.

- Selecting clustering or individual block

When selecting [Step 2](#) grouping or individual block at the top of [Container Map](#), you can view the event. For more information about **grouping** (clustering), see [the following](#).

❗ Before entering the **Display Detail** screen, depending on the view or grouping selected at the top of [Container Map](#), the events that can be viewed may differ.

- Using the [2](#) time selector, you can set the time zone to view event records in. The default value is [1 day](#).

❗ For more information on how to use the time selector.

- The [event number trend](#) in the [3](#) selected time zone can be viewed with a bar chart.
- In the [4](#) list format, you can see the event records.
 - If you select [Hide Tag](#) on the upper right of the event record list, you can see the [timestamp](#) and [message](#) excluding the tags while recording events.
 - [Kubernetes](#) events
 - You can see the data by selecting a desired event type: [Normal](#) or [Warning](#). These event types are identical to the event types provided by Kubernetes.
 - The bars on the left of each event indicate the event types. Gray means [Normal](#), and orange means [Warning](#).
 - [WhaTap](#) events
 - Select a desired event level to search among [Critical](#), [Info](#), and [Warning](#).
 - The bars on the left of each event indicate the event levels. Gray indicates [Info](#), orange [Warning](#), and red [Critical](#).

Kubernetes event

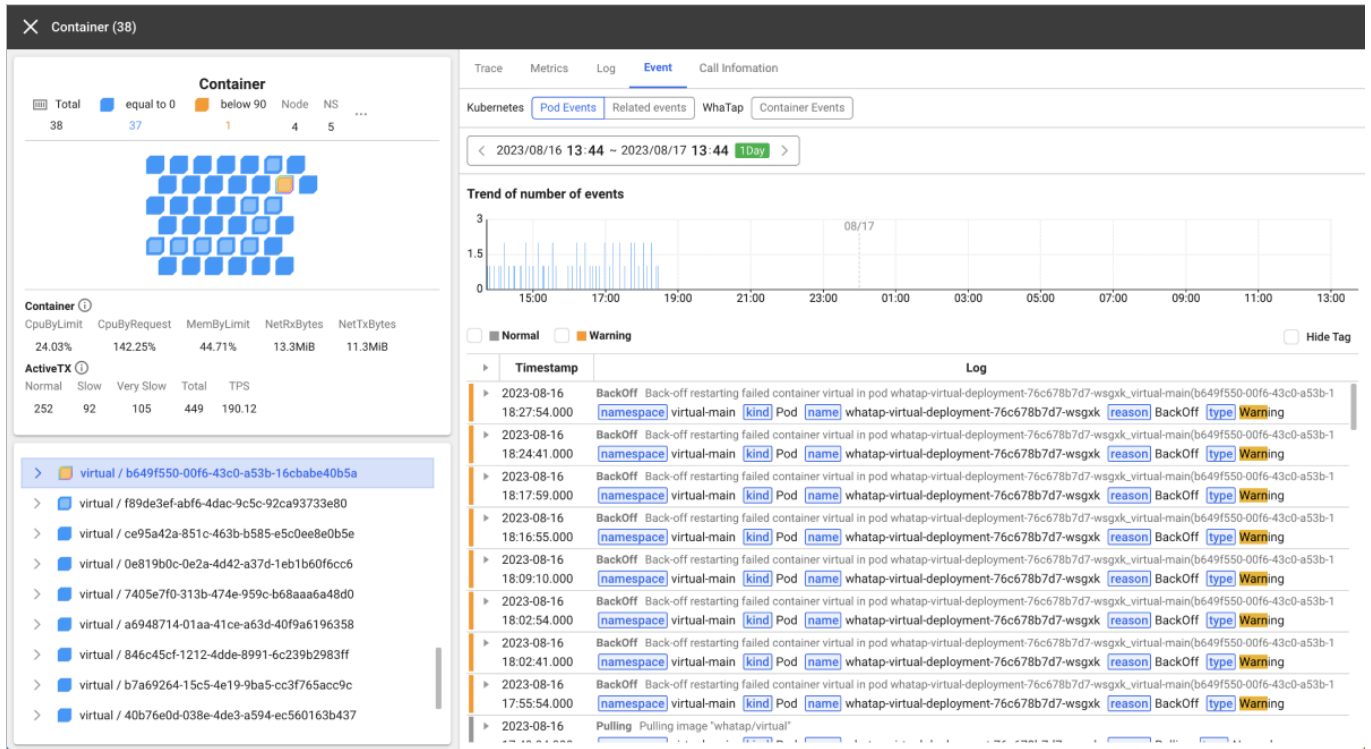
Kubernetes events are provided by Kubernetes itself. In the Kubernetes environment, you can use the `kubectl get events` command to get the event data of the Kubernetes cluster.

```
$ kubectl get events -A
NAMESPACE LAST SEEN TYPE REASON OBJECT MESSAGE
community 15m Normal ScalingReplicaSet deployment/baekdusan Scaled up replica set
baekdusan-59554d4859 to 5
community 12m Normal ScalingReplicaSet deployment/suraksan Scaled down replica set
suraksan-66fc4bf889 to 3
community 5m18s Normal ScalingReplicaSet deployment/baekdusan Scaled down replica set
baekdusan-59554d4859 to 3
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 7m19s Normal ScalingReplicaSet deployment/mountain Scaled up replica set
mountain-6795bfbc54 to 5
community 27m Normal ScalingReplicaSet deployment/mountain Scaled down replica set
mountain-6795bfbc54 to 4
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 5m18s Warning FailedToUpdateEndpoint endpoints/baekdusan-svc Failed to update endpoint
community/baekdusan-svc: Operation cannot be fulfilled on endpoints "baekdusan-svc": the object has been modified;
please apply your changes to the latest version and try again
```

However, the event data is not stored continuously. The events that occurred after a specified time are no longer

available.

WhaTap Kubernetes collects and stores all Kubernetes events that occurred in the user cluster environment. Event data such as pod, ReplicaSet, deployment, and namespace is provided according to the target to check.



The bars on the left of each event indicate the event types. Gray means Normal, and orange means Warning. The Kubernetes events provided by user selection are as follows.

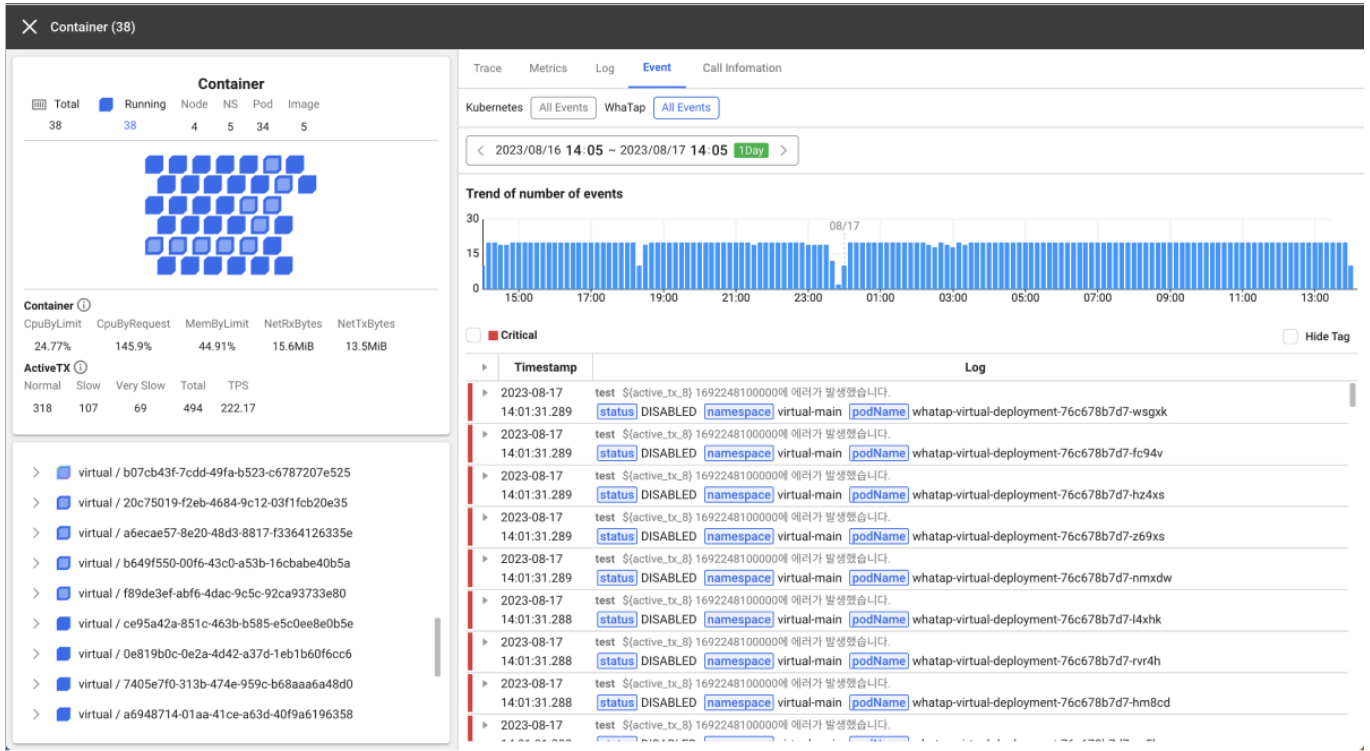
| Selected target | Provided event | Provision-related event |
|-----------------|--|--|
| All | All project events | |
| Namespace | All events under the namespace | |
| Node | Node events or all pod (container) events in the node | |
| Deployment | ① kind is Deployment②
name is identical to the user | name starts with the user deployment name. |

| Selected target | Provided event | Provision-related event |
|--------------------|---|--|
| | deployment name | |
| ReplicaSet | ① kind is ReplicaSet.②
name is identical to the user's ReplicaSet name | name starts with the user ReplicaSet name. |
| Pod
(container) | ① kind is Pod②
name is identical to the user pod name | The pod starts with the name of the deployment to which the pod belongs or with the ReplicaSet name. |

Kubernetes events stored in WhaTap consists of message , namespace , kind , name , reason and type .

WhaTap event

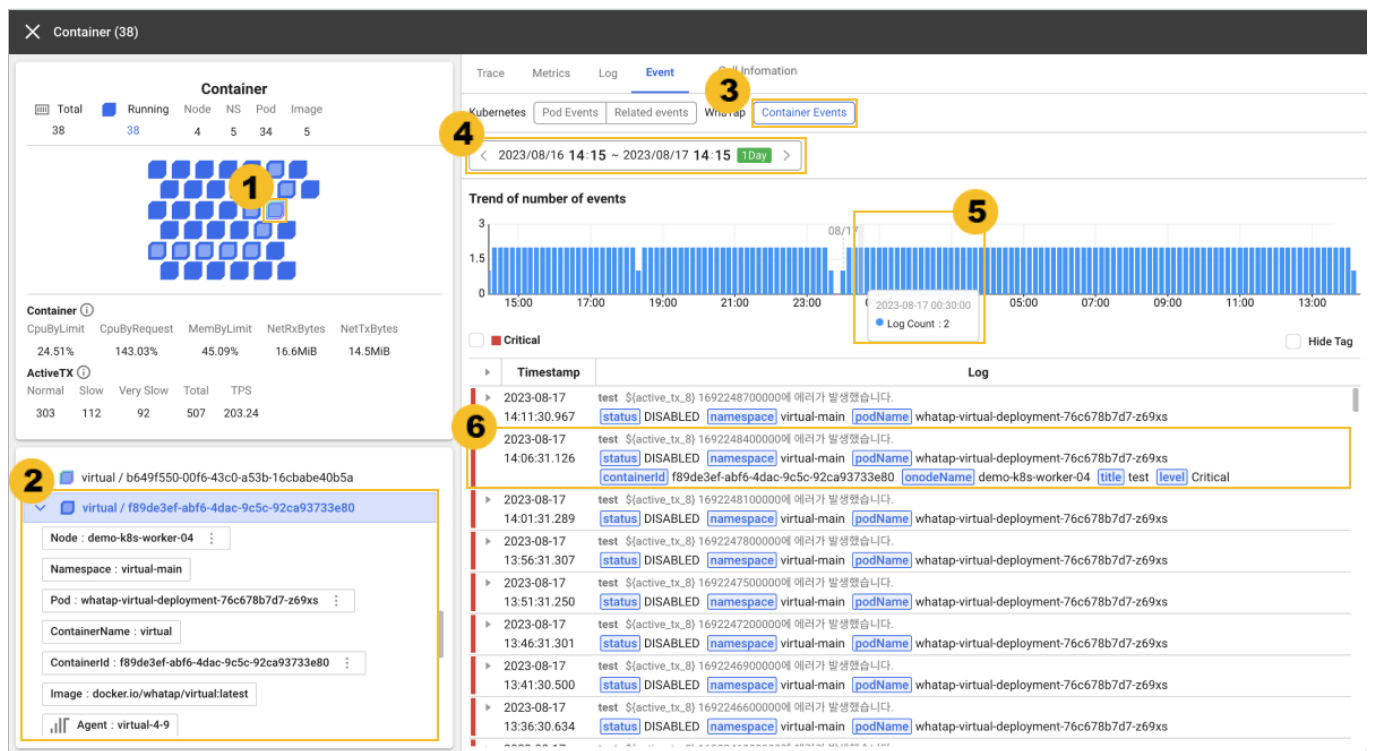
WhaTap events occur depending on the event conditions registered by the event configuration. Under the Event tab on the detailed view screen of Container Map, you can see the list of events based on the WhaTap alert settings.



The bars on the left of each event indicate the event levels. Gray means Info, orange Warning, and red Critical. Available Whatap events according to the selected target are as follows.

| Selected target | Provided event |
|-----------------|--|
| All | All project events |
| Namespace | All events under the namespace |
| Node | <code>onodeName</code> is identical to the user node name |
| Deployment | <code>deployment</code> is identical to the Deployment name |
| ReplicaSet | <code>replicaSetName</code> is identical to the user ReplicaSet name |
| Pod | <code>podName</code> is identical to the user pod name |
| Container | <code>containerId</code> is identical to the user Container ID |

WhatTap events consist of the items such as `message`, `status`, `title`, and `level`. The following example screen displays the [WhatTap Container Event](#) records.



- From the summary information card on the left of **1 Display Detail**, select a target to view.
- From the **2** summary information list, check the information of the selected one.
- In the **3 Event** tab, select a [WhaTap](#) event.

Before entering the **Display Detail** screen, you can view [container events](#) by setting the option at the top of [Container map](#) to Container view.

- Through the **4** time selector, select a time to view. The default value is **1 day**.
- Through the **5** bar chart, see the [Trend of number of events](#) in the specified time zone.

If you look at the trend for event counts for the specified time ranges, you can see that two events have been counted.

- From the **6** event history, check the information of event records. You can see the levels for each event through the bars on the left of the list. To check all event messages and tags, select the **▼** icon.

In the event record list, you can view for two **Warning** level events:

CPU utilization of 85% or more by the container limit and GC count or more .

Select Home > Select Project > [Dashboard](#) > [Container Map](#) > Target, and then in **Detailed analysis**, select the [Call Information](#) tab.

- App Call Flow
- Remote Access Status

App Call Flow



connection relationship between three nodes: custom group (Okind), Pod, and DB connection (DBC) and outbound call (HTTTPC). Each node provides the following data.

1 Custom group (okind)

It is the name of the group specified by a user. If a group (okind) is not specified, the node does not appear.

2 Pod

It indicates the name of the pod to which the container belongs. You can see the total number of active transactions and processing speed within the container application. After selection is made, [Active transaction](#) appears.

- [Active Tx](#): Indicates an active transaction withing the container application.
 - [Total](#): Indicates the number of all active transactions withing the container application.
 - [Normal](#): Indicates the number of transactions processed within 0 to 3 seconds.
 - [Slow](#): Indicates the number of transaction processed within 3 to 8 seconds.
 - [Very Slow](#): Indicates the number of transactions processed for over 8 seconds.
- [Application](#): Indicates the name of the application within the Pod.
- [Node](#): Indicates the name of the node that the Pod belongs to.
- [Namespace](#): Indicates the name of the namespace that the Pod belongs to.
- [ReplicaSet](#): Indicates the name of the ReplicaSet that the Pod belongs to.
- [Deployment](#): Indicates the name of the deployment that the Pod belongs to.

3 DB connection and outbound call

- [DB connection \(dbc\)](#): Indicates the DB connection called by the container application within the Pod. When selected, it goes to the [Remote Access Status](#) view tab.
 - [Active Tx](#): Indicates an active transaction within a container application that is being connected to the database.
 - [Total](#): Indicates the number of all active transactions withing the container application.
 - [Normal](#): Indicates the number of transactions processed within 0 to 3 seconds.

- **Slow**: Indicates the number of transaction processed within 3 to 8 seconds.
- **Very Slow**: Indicates the number of transactions processed for over 8 seconds.
- **Outbound call (httpc)**: Indicates an external API called by the container application within the Pod. When selected, it goes to the **Remote Access Status** view tab.
 - **Active Tx**: Indicates an active transaction within the container application that is calling an external API.
 - **Total**: Indicates the number of all active transactions withing the container application.
 - **Normal**: Indicates the number of transactions processed within 0 to 3 seconds.
 - **Slow**: Indicates the number of transaction processed within 3 to 8 seconds.
 - **Very Slow**: Indicates the number of transactions processed for over 8 seconds.

Remote Access Status

Using the outbound call logs, the information on whether the outbound call target is normal is provided based on the connection status of the destination called by the monitored target. Through the **Remote Access Status** view function, the visibility to inaccessible targets is provided. Through this, you can easily identify the problem point when a service issue occurs. Additionally, any resource encroachment such as increased threads and connections due to continuous connection failures can be detected and responded in advance.

❗ Remote Access Logs

This log records the success or failure events for outbound calls by monitoring target through the application agent configuration.

- Outbound call types are as follows.

httpcall , db , socket

- To collect remote access logs, add the following options in **Agent CONFIG..**

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

The screenshot displays the 'Call Information' interface. At the top, there are tabs for 'Trace', 'Metrics', 'Log', 'Event', and 'Call Information'. Below these, there are links for 'App Call Flow' and 'Remote Access Status'. The main area is divided into two panels. The left panel, labeled '2', is titled 'Remote Access Logs' and contains a table with columns 'Time', 'Agent', and 'Log'. The table lists various log entries with timestamps, agent names, and log details. The right panel, labeled '3', is titled 'Destination Status' and shows a list of destinations with their status. A top bar, labeled '1', contains controls for 'Agent' and 'Okind' selection, a search field, and playback/pause buttons.

| Time | Agent | Log |
|--------------------|---------------|--|
| 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 |

In the **Remote Access Logs** tab, remote access logs can be viewed in a list format. Up to 500 logs are loaded and displayed for the last 24 hours. When viewing in real time, data is updated every 3 seconds.

1 Options at the top bar

- You can determine whether to display the **Agent** or **BIZ** column.
- Through the search term input field, you can filter columns containing agents, tasks, and logs to which the search term belongs.
- ▶ **Play**: If the logs have been updated, you can go to the bottom to check the most recent logs.
- || **Pause**: Automatic scrolling is stopped temporarily.
- ▲ ▼ : Moves to the top or bottom.

2 Remote Access Logs

You can easily see the key content of the remote access logs through icons and tags. In case of an error log, it is displayed in **red**.

- Outbound call types are guided through the icon (db), (httpcall), and (socket).
- **target** : Indicates an outbound call and connection target.
- **content** : Indicates the log content.
- **exception** : Indicates the content of the log recorded when an error occurs.

3 Destination status

Check the list of outbound call targets for all viewed logs. Depending on the last connection status, **Success** or **Fail** appears.

- View mode
 - **List**: Default list view. Scrolling occurs when the entire list exceeds the screen height.
 - **Map**: If the list goes beyond the screen height, the length of the list is shortened and outbound call targets are listed horizontally.
- Filtering

When you select an outbound call target, you can see the list of remote access logs by filtering only the logs for the call targets. It is released when selected again.

In case there is no data

Check whether remote access logs have been collected for the last 24 hours for real-time view. If no remote access log has been collected for the last 24 hours, the following guide message is provided.



Currently, we cannot find any logs related to remote calls within the last 24 hours on this project.

Remote access status visualizes the log of success/failure of external (remote) calls by monitored target through application agent settings.

- This function is **supported only when the application is Java**.
- To use this feature,
Add the agent settings below in the [Management] > [Agent Settings] menu.

- logsink_rt_enabled=true
- logsink_enabled=true

[Go to settings →](#)

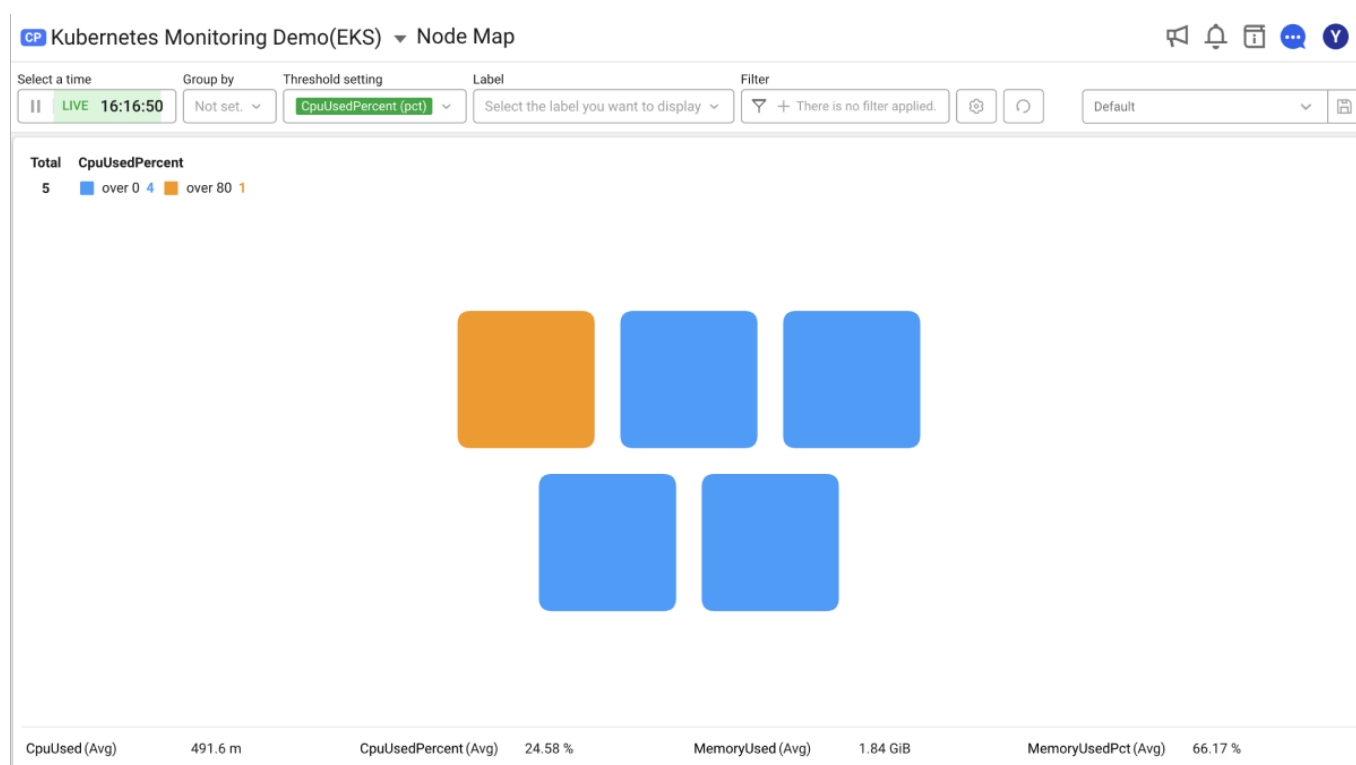
Node Map

Home > Select Project > **Dashboard** > **Node Map**

ⓘ Requirements

- The WhaTap Kubernetes agent 1.4.1 or later is required.
- If it is a Kubernetes cluster project (**CP**), you can view the corresponding menu.

The **Node Map** menu provides the feature to monitor the status of nodes from a cluster perspective. It visualizes the node status, resource usage, and agent status to help users understand intuitively. Additionally, it supports a feature that allows you to easily check the previous status by viewing the node status at a previous time.



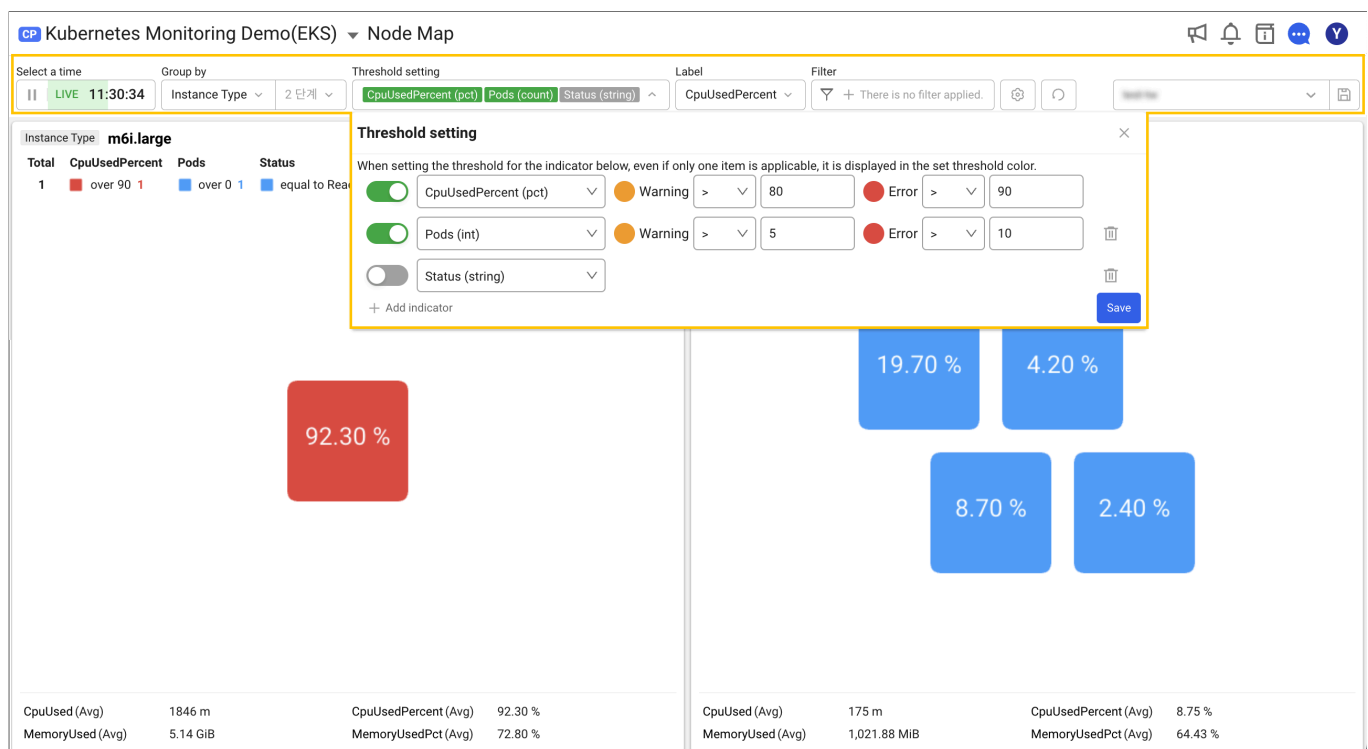
Basic screen guide

The **Node Map** menu largely consists of the **User Configuration** area and the **Node Map Chart** area.

! For more information on how to use the time selector.

User Configuration

In the **User Configuration** area at the top of **Node Map**, you can configure additional settings for **Group by**, **Threshold settings**, **Label**, **Filter**, presets, and ⚙️. The following example image has **Group by**, **Threshold settings**, and **Label** applied.



Grouping

Users can display nodes by **Group by** based on various conditions. This makes it easy to see the relationships

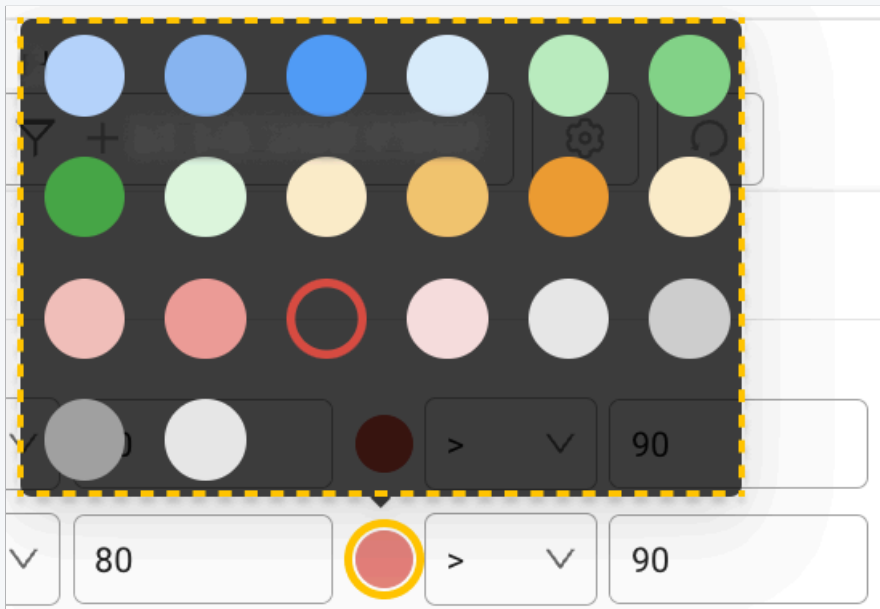
between nodes within each group. The **Group by** feature supports up to 2 levels. **Node Map Group by** allows users to select nodes based on specific criteria to create a customized monitoring environment.

Setting the threshold

Users can represent the node status with colors via **Threshold settings**. You can set one or more metrics, and you can specify warning (**Warning**) and error (**Error**) sections for each metric. The normal (**Normal**) section is displayed in **blue**. Depending on the set range, an appropriate color to the status is applied.

1. Click the **Threshold settings** option.
2. Select a desired metric and then set **Warning** and **Error** sections.
3. To add a metric, click + **Add Indicator** at the bottom.
4. Once you have finished setting the thresholds for each desired metric, click **Save** at the bottom.

ⓘ The abnormal sections such as **Warning** and **Error** are set to **orange** and **red** by default. You can customize the colors for each state in **Threshold settings**.



String type metrics

If a string type (**String**) metric has a specific string, it is designated as **Error**. For example, if the **Status (string)**

metric contains a text, `NotReady` , the node is marked as **Error** and displayed in **red**.

| Metric name | Value | Error status |
|----------------|--|-------------------------|
| Status | <code>Ready *</code> / <code>NotReady *</code> | <code>NotReady *</code> |
| Ready | <code>true</code> / <code>false</code> | <code>false</code> |
| PIDPressure | <code>true</code> / <code>false</code> | <code>true</code> |
| DiskPressure | <code>true</code> / <code>false</code> | <code>true</code> |
| MemoryPressure | <code>true</code> / <code>false</code> | <code>true</code> |
| Unschedulable | <code>true</code> / <code>false</code> | <code>true</code> |

Node status display priority

Its highest priority is the **Error** section. If any of the metrics falls into the **Error** section, the node appears in **red** at first. In the following example, both **a** and **b** nodes are marked as **Error**.



Label



Users can set text labels inside a node block to view the set key data at a glance. If you set **Label** based on the [CpuUsedPercent](#) metric, you can see the information inside the node block as in the example image on the left.

Filtering

Users can use filters by selecting nodes that meet specific conditions for intensive monitoring.

Preset

Users can save the set options as a preset and load them if needed. Presets are saved for each project. This allows project members to monitor the cluster with the same settings. A default preset ([Default](#)) is provided, and the default preset cannot be deleted.

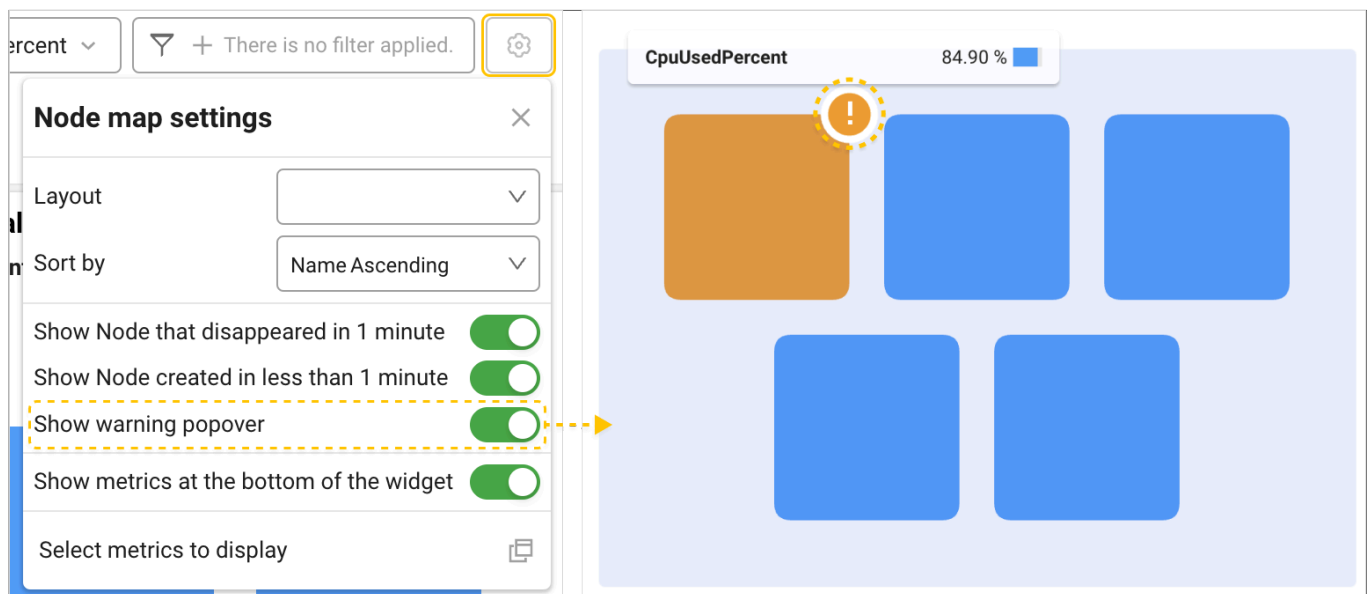
! Default preset

The default preset for **Node Map** is set to the threshold for the [CpuUsedPercent \(pct\)](#) metric as follows:

- **Normal**: The value is 80 or less.
- **Warning**: The value is greater than 80 and 90 or less.
- **Error**: The value is greater than 90 .

Additional function

Through the ⚙️ on the upper right of **Node Map**, you can adjust additional detailed settings as follows:



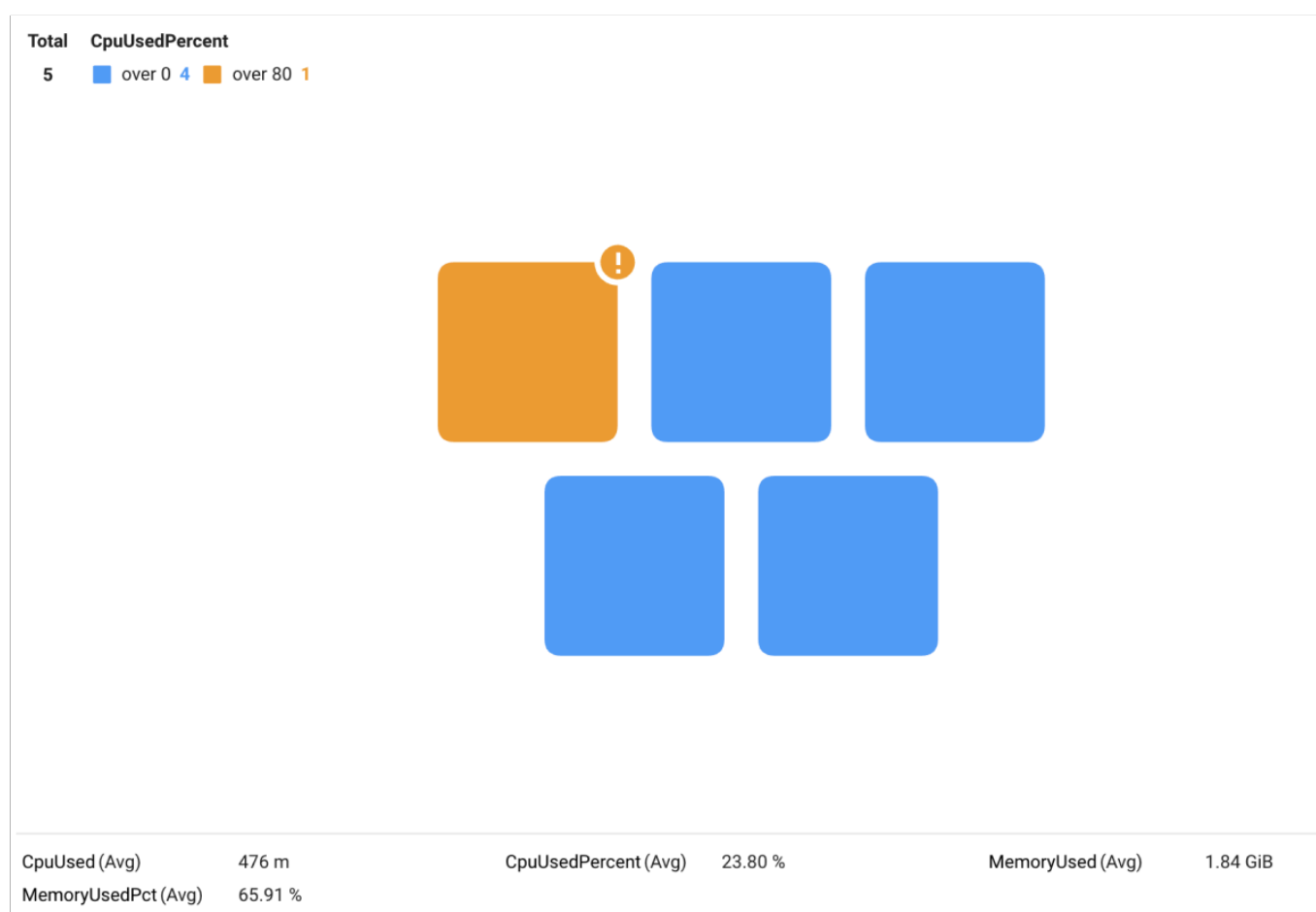
- **Layout**: It fixes columns of the grouped widgets.
- **Sort by**: It sorts the widgets grouped by node count or name.
- **Show Node that disappeared in 1 minute**: It displays the nodes that have disappeared within the last

minute.

- **Show Node created in less than 1 minute:** It displays the nodes created within the last minute.
- **Show Warning Popover:** It displays a warning popover when a specific node is in a **Warning** or **Error** status depending on **Threshold settings**.
- **Show Metrics at the Bottom:** Set whether to display the average, sum, maximum, and minimum values at the bottom of the widget.
- **Metrics to Display:** Select a metric to display at the bottom of the widget. Select a metric and set it to display data for the metric as an average, sum, maximum, or minimum for all grouped nodes.

Node map chart

The **Node Map** chart consists of the following three areas:



1 Top area of the widget

It displays the total number of nodes, or the total number of grouped items, and the numbers of states for the selected metric.

2 Visualization area in the center

It provides a visual dashboard that allows you to intuitively check the overall node status and count.

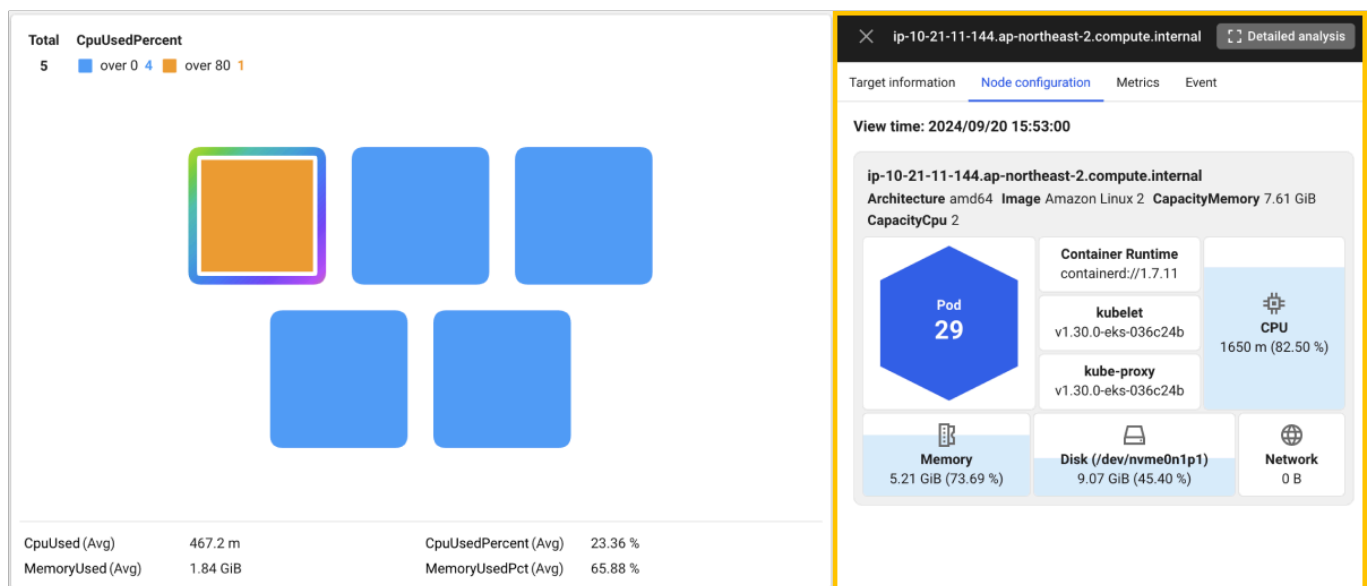
3 Bottom area of the widget

It displays the statistics for a selected metric. It displays the data in a desired manner among sum, maximum, and minimum values as set through **Metrics to Display** in Additional Settings.

! **Node Map** only displays the nodes that have node agents installed.

Summary view and detailed analysis

In **Node Map**, select a node to view. You can check the summary data of the object through the summary view screen that appears on the right. You can check the detailed analysis screen by clicking **Detailed analysis** on the summary view screen.



Target information

In the **Target information** tab, you can check the key properties of a selected node. You can also move to the [Node list](#) and [Container map](#) menus.

ⓘ The [Labels](#) metric is supported in Kubernetes agent 1.7.9 or later.

Node summary analysis

You can check the summary information of node resource allocation and usage status through **Node summary analysis**.

- **Basic information:** It provides the cluster version of the node, startup time, current status, whether it is schedulable, internal IP address, region, and zone.
- **Operating system and hardware information:** It provides the operating system on which the node is running, OS image, CPU architecture, number of cores, and total memory size. You can check the allocatability and current usage of the memory and CPU of a node.
- **Resource usage status:** It displays the remaining resources through the difference between requests and allocatable resources (CPU, memory) for all containers running on the node. It summarizes the resource usage status, including the actual CPU and memory utilization.
- **Disk usage status:** It provides information about the disks attached to the node, file system types, device IDs, mount points, total and used capacity, and utilization.
- **Pod status and resource availability:** By comparing the total number of allocatable Pods on the node to the number of running Pods, you can see how many additional Pods can be run. In addition, the stability of the node is checked through the status of MemoryPressure, DiskPressure, PidPressure, and such.

Node configuration

It provides a summary of the node configuration and overall resource usage selected in the **Node configuration** tab. You can comprehensively see the hardware and software components of the node. Users can monitor resource usage of nodes and the status of each component, as well as performance and operational stability.

View time: 2024/09/20 14:40:30 ~ 2024/09/20 18:11:25

 This is a summary analysis based on statistical data that can be used as reference material for detailed analysis or system improvement.

ip-10-21-140-100.ap-northeast-1.compute.internal is an amd64 linux node with 2 CPUs and 3.77 GiB memory. This node maintains light usage most of the time, but **occasionally experiences moderate load on CPU, Memory**. If you want stable operation, we recommend periodic monitoring in preparation for when load occurs.

✓ **Network usage** is up to 1.31 MiB, which is **24.69 times higher than the average usage, 54.44 kiB, so you need to keep an eye on the volatility of traffic patterns.**

AVG

MAX

Architecture amd64 **Image** Amazon Linux 2 **CapacityMemory** 3.77 GiB **CapacityCpu** 2



Container Runtime
containerd://1.7.11

kubelet
v1.30.2-eks-1552ad0

kube-proxy
v1.30.2-eks-1552ad0



CPU
1380 m (69.00 %)



Memory
1.04 GiB (30.50 %)



Disk (/dev/nvme0n1p1)
4.51 GiB (5.64 %)



Network
1.31 MiB

- Node system specifications

- [Architecture](#): Hardware architecture used by the node.
- [Image](#): Information about the operating system image that the node is running on.
- [CapacityMemory](#): Total memory capacity of the node.
- [CapacityCpu](#): Total CPU capacity of the node.
- **Software configuration**
 - [Container Runtime](#): Version of the container runtime used on the node.
 - [kubelet](#): Version of the Kubernetes kubelet running on the node.
 - [kube-proxy](#): Version of the kube-proxy that manages the network features.
- **Resource usage status**
 - [CPU](#): CPU usage of the node.
 - [Memory](#): Memory usage of the node.
 - [Disk](#): Disk capacity and usage rate on the node.
 - [Network](#): Network throughput

- ! • If you click CPU, memory, disk, or network, the **Metrics** tab appears that can display the usage trend for the query period.
- By clicking [AVG](#) and [MAX](#), you can see the average usage and maximum usage for the query period.

Process

The **Process** tab provides information about the major Kubernetes management processes inside the selected node. It can prevent cluster failures caused by process issues and minimize the downtime.

< 2024/10/29 14:27 >

name

Column Settings

CSV

| De... | Nodelp | Pid | Name | CreateTime | CpuUsedPercent | MemoryUsedPercent | Rssl |
|-------------|--------------------------|--------|---------------|---------------------|--------------------|--------------------|------|
| <div></div> | ip-10-21-110-99.ap-no... | 2833 | containerd | 2024/10/29 09:20:07 | 0.40 % <div></div> | 1.09 % <div></div> | |
| <div></div> | ip-10-21-110-99.ap-no... | 2960 | kubelet | 2024/10/29 09:20:12 | 1.39 % <div></div> | 1.35 % <div></div> | |
| <div></div> | ip-10-21-110-99.ap-no... | 3237 | kube-proxy | 2024/10/29 09:20:22 | 0.10 % <div></div> | 0.60 % <div></div> | |
| <div></div> | ip-10-21-110-99.ap-no... | 3442 | aws-k8s-agent | 2024/10/29 09:20:25 | 0.10 % <div></div> | 0.68 % <div></div> | |
| <div></div> | ip-10-21-110-99.ap-no... | 122157 | coredns | 2024/10/29 13:50:56 | 0.20 % <div></div> | 0.64 % <div></div> | |
| <div></div> | ip-10-21-110-99.ap-no... | 122564 | coredns | 2024/10/29 13:51:03 | 0.20 % <div></div> | 0.63 % <div></div> | |

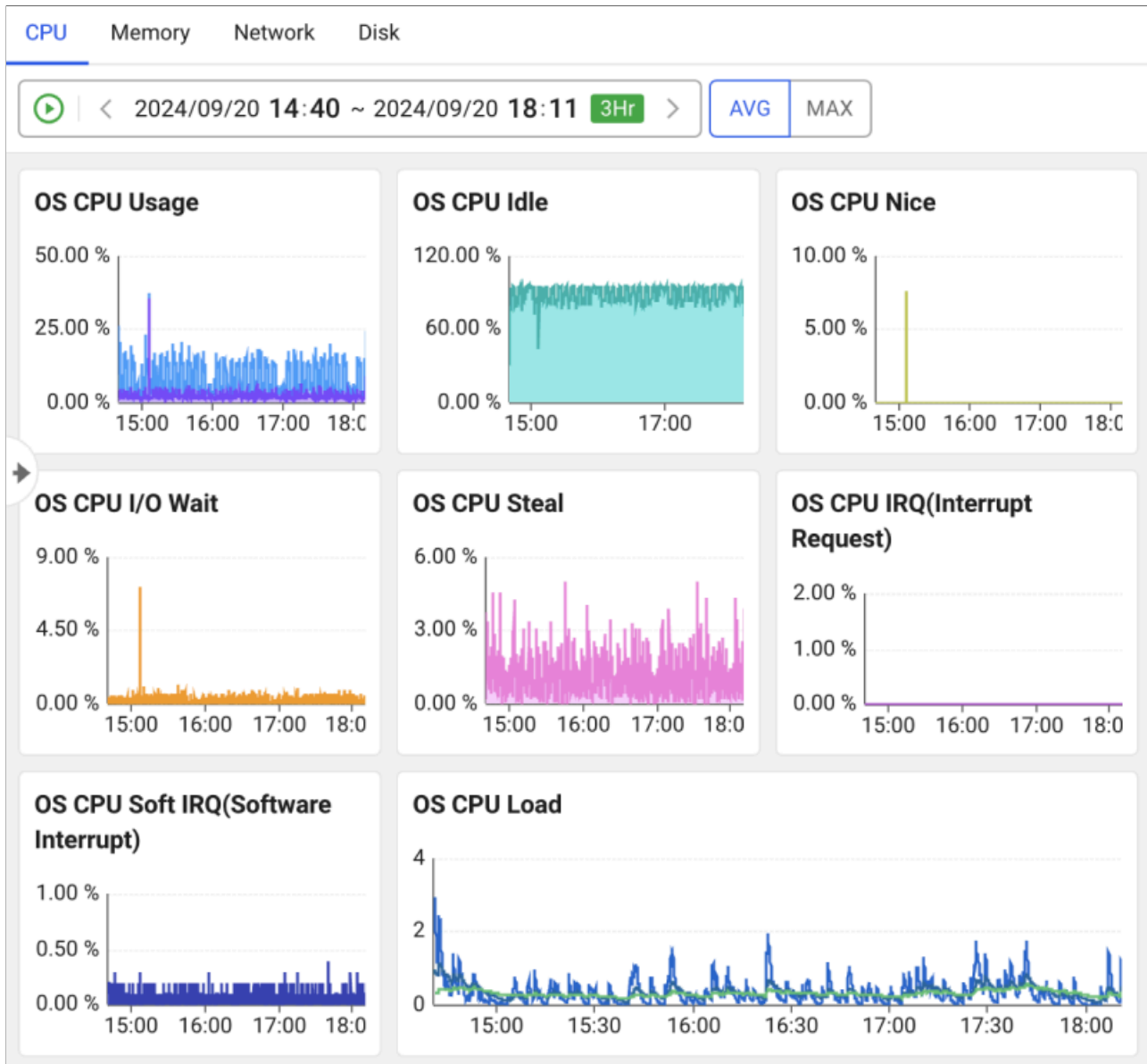
Process details

- You can check information details about a process by selecting the  icon in the **Detail** column on the left of the process list.
- You can select a method how to aggregate the statistical data during retrieval while querying long-term data of 3 days or more by clicking [AVG](#) or [MAX](#).
- When you click a process key data value, the value is copied.

 The **Process** tab is supported in Kubernetes agent 1.7.14 or later.

Metrics

You can check the resource usage trend of the node select in the **Metrics** tab. You can check the usage changes in [CPU](#), [memory](#), [network](#), and [disk](#) based on the average ([AVG](#)) and maximum ([MAX](#)) values during the query period.

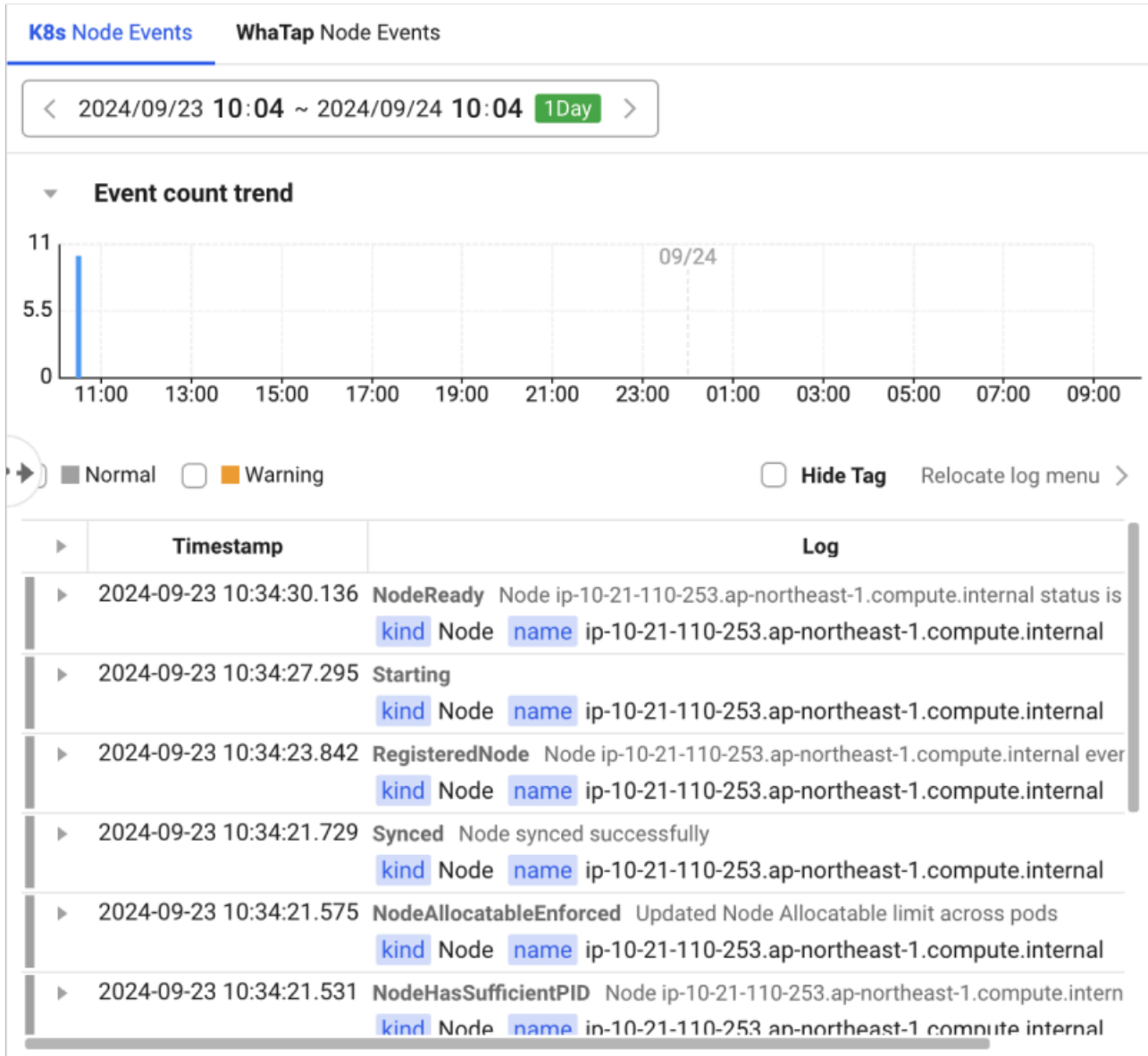


- **AVG**: Average usage for the query period.
- **MAX**: Maximum usage for the query period.
- **Total**: Total traffic usage across multiple network interfaces.
- **In/Out**: Inbound (receiving) and outbound (sending) traffic usage for a specific network interface.

ⓘ The **K8s CPU Request / Allocatable CPU (%)** widget is supported in Kubernetes agent 1.7.10 or later.

Event

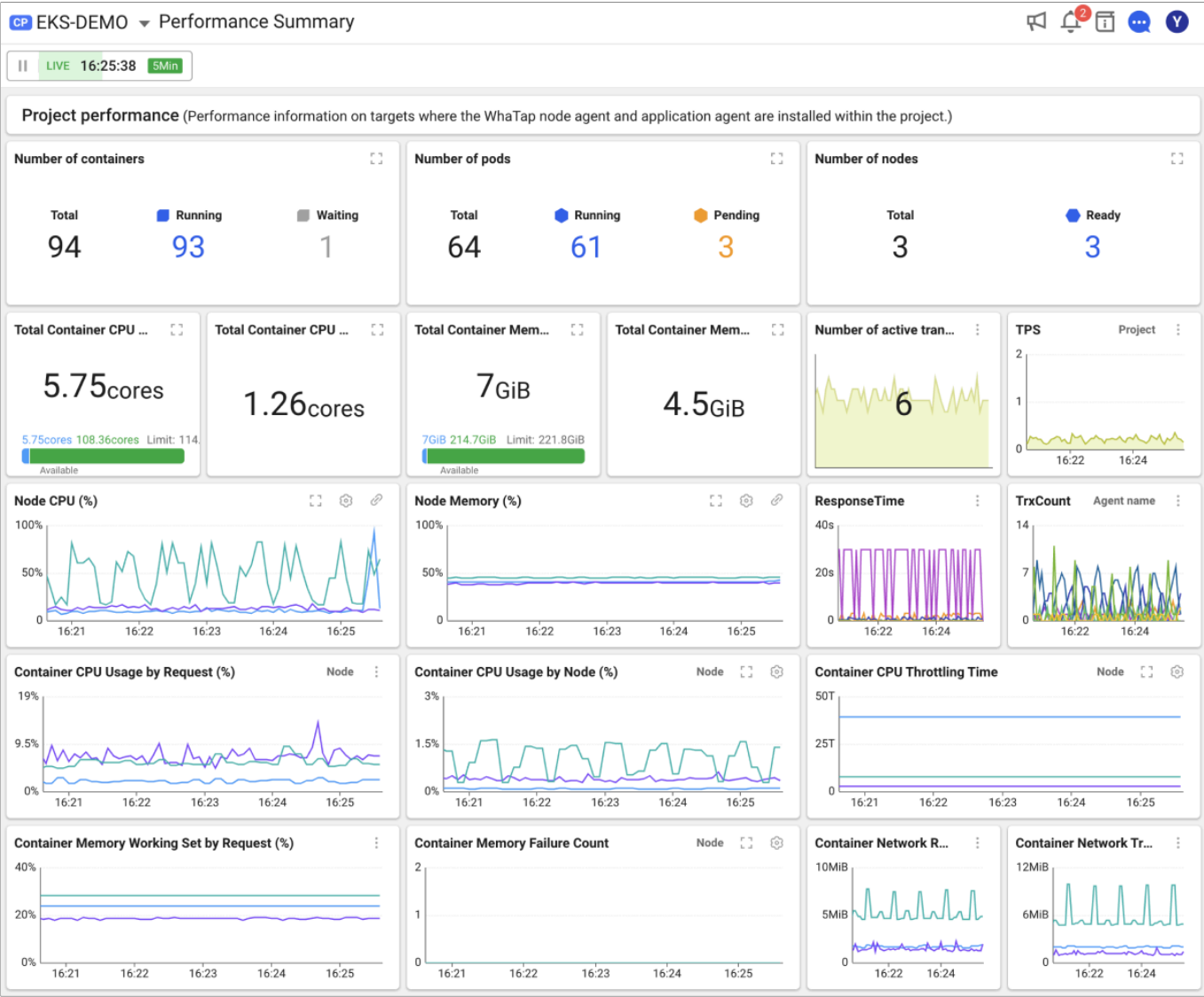
You can check [K8s Node events](#) and [WhaTap Node events](#) that occurred on the node selected in the **Event** tab. [K8s node events](#) include important activity logs within the cluster such as changes in node status, while [WhaTap node events](#) provide information about the events triggered based on the user-defined alert criteria.



Performance Summary

Home > Select Project > [Dashboard](#) > [Performance Summary](#)

The charts for each late time zone display information of the target where the WhaTap node agent and application agent within the project. Information separated by the namespace project can be found in the namespace project.



- **Shortcut option**

By selecting  [More](#) icon at the upper right of each widget, you can see the  [Shortcut](#) option as follows:

- **Line chart configuration option**

You can select the  [More](#) icon at the upper right of each line chart widget, or can directly check the  configuration options. For line charts, you can change the ID through the configuration option. The setting is not saved.

For the [Node CPU \(%\)](#) widget, you can view the data by [Node](#) for example. You can change the metric to view by [ContainerName](#).

 If the agent installation is not completed, data may not appear.

Node info

The widgets for node data are as follows:

- Recent 30 minutes chart
 - [Node CPU \(%\)](#), [Node Memory \(%\)](#), etc.

-  • On the upper right of the node count widget, if you select  [More](#) >  [Shortcut](#), [Cluster](#) > [Node List](#) appears.
- If you select  [More](#) >  [Shortcut](#) on the upper right of the [Node CPU \(%\)](#) widget or [Node Memory \(%\)](#) widget, [Analysis](#) > [Metrics Chart](#) appears. Using this menu, you can see the chart with detailed conditions.

Container Info

The widgets for container data are as follows:

- Recent 10 minutes chart
 - [Container Status](#), [Container CPU \(%\)](#), [Container Memory \(%\)](#), etc.

Application information

The widgets for application data are as follows:

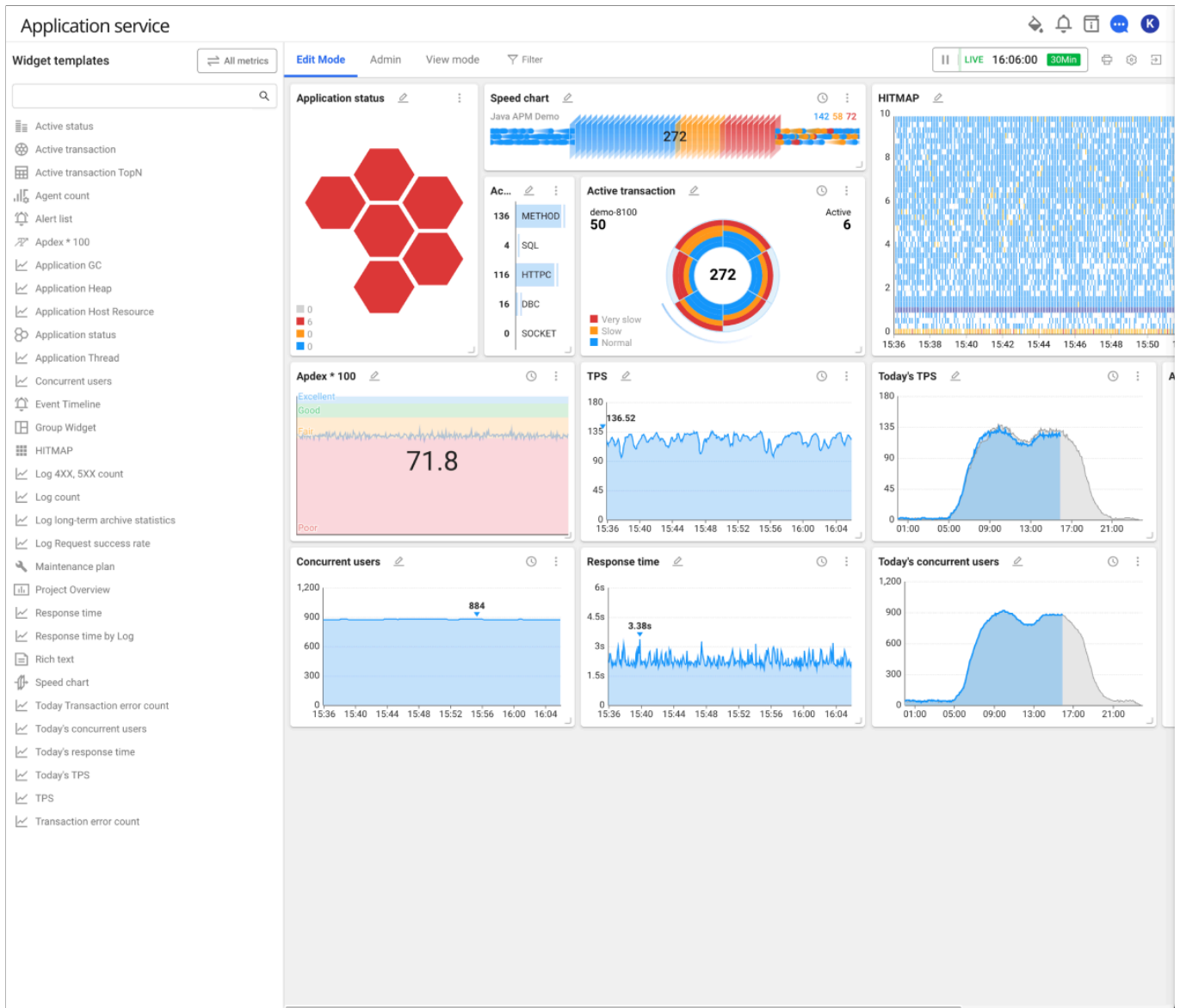
- Recent 10 minutes chart
 - [Active TX](#), [TPS](#), [Response Time](#), etc.

ⓘ The application data is displayed when the target application agent to be installed exists.

Flex Board

Flex Board is a custom integrated dashboard. You can create a real-time dashboard with the project data such as applications, servers, databases, and containers.

You can quickly complete the initial setup with preconfigured templates and build dashboards in your preferred format. By adding data widgets or modifying properties, you can display information in the form you need, and with filtering functions, you can narrow down monitoring targets. When you set a time range, you can check data from a specific point in time, and by using auxiliary charts, you can analyze data in various ways. Dashboards can be added to favorites for easy access, and personalized dashboards can be copied to other accounts for reuse.



The usage scope of the Flex board differs depending on the following entry:

Home > Integrated Flex Board

- When creating a widget, all projects that can be viewed are provided as options.
- The dashboard is saved in your account and can be shared with other users with the "To copy" feature.
- Individual account dashboards are not affected by the role, but the shared dashboards are read-only and cannot be edited.

Home > Select Project > Dashboard > Flex Board

- When creating a widget, the related project data is automatically entered.
- The created Flex board is automatically shared with other project members.
- The users with the project **Edit** role or **Project flexboard update** role, and **site admin** role can modify the board.

✓ Users with edit permissions for the Flex Board can use the following features:

- Export and import dashboards as JSON files
- Check data requests and responses within the dashboard
- View and edit widget configuration options in JSON format

ⓘ In the **Flex Board** menu within the project, only the users with the dashboard editing role can access **Edit Mode**, **Admin Mode**, and **Filter**.

- Accessible member permissions
 - Project **Edit** role
 - **Project flexboard update** role
 - **Site Admin** role
- For more information about the **View mode** and **Filter**, see [the following](#).
- For more information about the member roles, see [the following](#).

Using Flex Board

- ✓ You can build dashboards using various widget templates, including metrics widgets.
For details on the features and usage of each template, see the [Widget template](#) document.

Apply a preset template

Use a preset template to speed up the initial setup, then modify it as needed.

1. Select the **Flex Board** menu.
2. From the **Template** list on the left, select the template you want.
3. When the **Flex board management** window appears, enter a name in **Rename**.
 - In **Layout**, select a widget and drag it to the position you want.
4. Click the **Save** button. A Flex board is created based on the selected template.
 - You can check the created board in the board list of **Flex Board**.

Create a Flex board

You can create a Flex board and place widgets to build your own dashboard.

1. In **Flex Board**, click the **Create Dashboard** button at the top right.
2. When the **Create Dashboard** window appears, enter the dashboard **Name** in the input box.
3. Select a widget placement method, then click the **Create Dashboard** button.
 - **Fixed layout flexboard**: free widget placement with pixel basis
 - **Responsive flexboard**: responsive grid layout based on browser size (configure the grid, then click the **Create Dashboard** button)
4. In the **Widget templates** selection screen, choose the widgets to place on the dashboard.

- To add metrics widgets, click the **All Metrics** button on the right of **Widget templates**.

- ! • **Widget templates:** choose from a predefined widget list that summarizes key indicators commonly used in monitoring.
- **All metrics:** create widgets based on all metrics data collected in your project. For details about metrics widgets, see [the following document](#).

5. Move the placed widgets or adjust their size.
 - When a crosshair cursor appears at the top of a widget, click and drag the widget to move it.
 - When an arrow cursor appears at the bottom right of a widget, click and drag to resize it.
6. After you finish placing widgets, select **View mode** to review the layout. To edit the layout again, select **Edit Mode** and adjust widget placement.
7. When all steps are complete, click the  (**Go to list**) icon at the top right.
 - You can check the created board in **Flex Board** under **Dashboard list**. Select the created board to view the new Flex board.

- ! We plan to add dashboard templates or widgets based on customer feedback. If you need a specific dashboard template or widget, contact us at support@whatap.io.

Edit a Flex board

! Flex board edit permissions

- **Home > Integrated Flex Board:** this is a personal account dashboard, so permissions do not apply; however, dashboards shared as read only cannot be edited.
- **Home > Select project > Dashboard > Flex Board:** users with project edit permission, project Flex board edit permission, or Site Admin permission can edit.

For details on member permissions, see [the following document](#).

1. In **Flex Board** under **Dashboard list**, click the edit icon . The **Flex board management** window appears on the right.
2. In the **Flex board management** window, configure Name, Project, Layout, widget.json, and option.json.
 - **Rename:** edit the dashboard name.
 - **Project:** select the project from which widgets will fetch data.

 You can configure the **Project** option only in **Home > Integrated Flex Board**.
If you entered from **Integrated Flex Board**, you can select the data source project for each widget.

1. Click the **+** button at the top right of a widget, then search and select one or multiple projects.
2. After selecting projects, click the **Apply** button.

- **Layout**
 - Drag widgets to move them.
 - Click and drag the bottom right of a widget to resize it.
 - To delete a widget, click the  icon at the top right of the widget.
- **Widget.json:** import dashboard widget settings in JSON format. Click the  icon to copy the JSON.
- **Option.json:** import dashboard option settings in JSON format. Click the  icon to copy the JSON.

Delete a Flex board

1. In **Flex Board > Dashboard list**, click the  icon for the item to delete.
2. In the confirmation message, click the **Delete** button.

 Dashboard deletion behavior differs for owners and shared users.

- If the dashboard owner deletes a dashboard, it is deleted for all user accounts with whom it was shared.



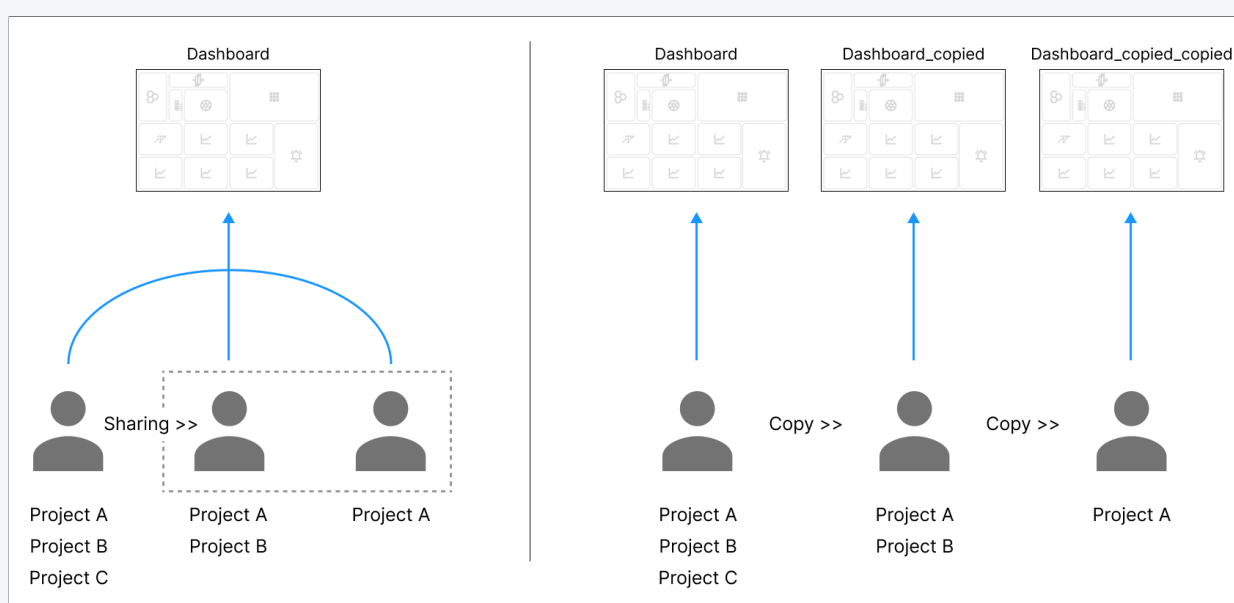
- If a user who received a shared dashboard deletes it, it is deleted only from that user's **Dashboard list**. The original dashboard remains.

Share a Flex board



- The Flex board sharing feature is available only in **Home > Integrated Flex Board**. It is not provided in a project's **Flex board** menu.

! Difference between share and copy



- Share: all members you share with view the same dashboard.
- Copy: if you copy a dashboard to another member or to yourself, you view a separate copied dashboard.
- Dashboards shared or copied as **Read only** cannot be edited. They can be edited only if shared as **Edit mode**.
- Members who receive a dashboard as **Read only** can also **re-share as read only** or **copy** that dashboard.



- You can share a dashboard only with members who belong to at least **one** of the projects included in the dashboard. Check whether the member to share or copy belongs to the relevant project.

☆ test

K8S-JINS-OKD-TEST, K8S-JINS-TEST-MICROK8S, DB-M...

Updated: 2023/12/22 11:10:24 AM



You can share a Flex board created in **Integrated Flex Board** with other members. When the owner edits a shared dashboard, the changes apply to all members it was shared with.

- In **Integrated Flex Board** under **Dashboard list**, click the icon for the dashboard to share.
- When the **Sharing a dashboard** window appears, select one or multiple members to share with, or enter accounts directly.
 - If you select **Read only**, members cannot edit the shared dashboard.
 - Share targets must belong to at least one of the projects included in the dashboard.
- Click the **Share** button.
 - Shared items display the **Shared** tag. Hover over the **Shared** tag to view information about the members you shared with.
 - Items shared as read only display the **Read only** tag.

| When you receive a shared dashboard | When you share a dashboard (dashboard owner) |
|-------------------------------------|--|
| | |



Dashboard sharing conditions

- The dashboard sharing feature is available only in **Home > Integrated Flex Board**.



- Anyone can share their own dashboard, and you can re-share or copy a dashboard you received.
- Dashboards received as **Read only** cannot be edited; dashboards received as **Edit mode** can be edited.
- If the **owner** deletes a dashboard, it is deleted for all accounts that received it.
- If a **recipient** deletes a dashboard, it is removed only from that user's **Dashboard list** and the original remains.
- For projects in a received dashboard where you do not have **View** permission, you cannot view data.

Copy a Flex board

Users can copy a Flex board and send it to other members.

1. In **Integrated Flex Board** under **Dashboard list**, click the  icon.
2. Choose a copy option.
 - **Copy to me:** copy to your own account.
 - a. An item with the `_copied` suffix is added to **Dashboard list**.
 - **Copy to others:** copy to another member's account.
 - a. Click **+ Account**.
 - b. Select the email of the member to copy to and click the **Copy dashboard** button.
 - c. An item with the `_copied` suffix is added to that member's **Dashboard list**.

Share as a JSON file

You can export Flex board settings to a JSON file to share with other members, or import settings from other members.

Export

1. Go to **Integrated Flex Board** or a project's **Flex board** menu.
2. In **Dashboard list**, click the  button.
3. Choose an export option.
 - **Include Project Information:** include dashboard project information in the JSON file when sharing.
 - **Except Project Information:** share without including dashboard project information in the JSON file.
4. Download the JSON file and send it to the member you want to share with.

Import

1. Go to **Integrated Flex Board** or a project's **Flex board** menu.
2. Click the  **Import** button at the top right.
3. Select the downloaded JSON file.

 Only members with **Edit** permission for the project can use this feature.

Flex board screen modes

Features vary by Flex board screen mode.

Accessible member permissions

Only users with **Edit** permission in a project's **Flex Board** menu can access **Edit Mode**, **Admin**, and **Filter**. For details on member permissions, see [the following document](#).

- **Project Edit** permission
- **Project flexboard update** permission
- **Site Admin** permission

Modes

Available features differ by **Flex Board** screen mode.

- **Edit Mode:** place widgets from the widget template list on the Flex board layout; edit, delete, or reposition widgets.
- **Admin:** manage widget properties in JSON format. Click the  button at the top right of a widget on the layout.
- **View mode:** after finishing layout, sizing, and detailed options, monitor each widget's data in real time like a dashboard.

Common features

- Data filtering
- Real time data time range settings
- Print mode
- Flex board management
- Data merge options

Data filtering

Use data filtering to narrow monitoring targets and build an efficient Flex board.

1. At the top of the screen, click the  **Filter** tab.
2. In the **Filter** window on the right, select filter targets. To select all filter targets, choose **Select all**.
 - **Agent** option: select agent types or an application list, with **Search** available
 - **Metrics** option: select a project or agent type; add **+ And** conditions, or auto select by string with **Contain** or **Exclude**
3. After selecting filter targets, click the **Apply** button.
4. Save frequently used conditions with the **Save filter** button.
 - a. Enter a filter name and select **Save**. **Saved filters** appears at the bottom of the **Filter** window.

Filtering conditions

Filtering conditions are divided into agent type and metrics type.

Agents provide only basic monitoring targets as selectable options, while metrics provide selectable tags for all metrics data collected in the project. The metrics type is useful when you collect data in a non agent form and want to apply filtering. For details on metrics, see [the following document](#).

| Type | Agent | Metrics |
|-------------------|---|-----------------------------------|
| Selection options | WhaTap project monitoring targets
(applications, servers, databases,
containers, Kubernetes PODs and NODEs) | All metrics tags including agents |
| Scope | All widgets | All widgets that use metrics data |
| Filtering method | Checkboxes | Equal, Contain, Exclude |

Set the lookup time

Use the time selector at the top right of **Flex Board** to set data lookup time for widgets.

- : stop real time mode and set a specific lookup time
- : switch back to real time mode

Print

You can save the Flex board screen as a PDF and print it.

1. On the Flex board screen, click the  icon at the top right to switch to print mode.
2. Configure options.
 - **Viewing filter options:** toggle filter options on the screen
 - **Convert to {mode} orientation / Convert to {mode} orientation:** change orientation to horizontal or vertical
3. After finishing, select **PDFDownload**.

4. To exit print mode, click the  **Exit print mode** at the top right.

Flex board management

You can change the Flex board layout; move, resize, or rename widgets; and use other management features. You can also open **Flex board management** by clicking the  icon in **Dashboard > Flex Board** under **Dashboard list**.

1. On the Flex board screen, click the  icon at the top right to open the **Flex board management** window.
 - **Rename:** change the Flex board name
 - **Layout:** move, resize, and delete widgets
 - **Don't Allow Other Users to Modify.:** restrict modifications
 - Members with edit permission can edit, delete, and modify; if you enable the option, it changes to **Don't Allow Other Users to Modify.: On**
 - **Widget.json:** edit and manage widgets on the Flex board in JSON
 - **Option.json:** edit and manage Flex board settings in JSON (project info, Flex board filters, time selection, and other settings)

 The **Widget.json** and **Option.json** features are available only to dashboard administrators.

2. After completing all settings, click the **Save** button.

Add a dashboard to favorites

You can add a Flex board you created to favorites for quick access. In **Dashboard > Flex Board** under **Dashboard list**, click the  button for the dashboard to add. It changes to .

Managing The Widgets

You can customize your dashboard by handling the widgets placed on the Flex board.

- ✓ You can build dashboards using various widget templates, including metrics widgets.
For details on the features and usage of each template, see the [Widget template](#) document.

Adding a widget

You can add any widget to the created Flex board.

1. In **Dashboard list** of **Flex Board**, select a dashboard to add the widget.
 2. On the upper left of the screen, select **Edit Mode**.
 3. Select a desired widget from the **Widget templates** list on the left. To add a metrics widget, select **All metrics** and then click a desired widget.
- If you have entered **Integrated Flex Board** on the Home screen and added a widget, click **Project selection** and then select target projects to collect.
 - If you have more widgets to add, repeat the same procedure.
4. After all steps are finished, select **View mode** to check the layout and then select  (**Go to list**).

Changing the widget attributes

In **Dashboard list** of **Flex Board**, select a board to change the widget properties. It goes to **Edit Mode**.

Changing the widget name

1. On the right of the widget name to change, select .
2. Set the widget values for **Title**, **Font size**, and **Color**, and then select **Save**.

- If the **Apply to the entire widget** toggle button is enabled, the font size and color are applied to other widgets.
- To lock the widget title, select [Configuration icon](#) and then click **Save**.

Time Setting

1. On the upper right of the widget to set the time, select ⌚.
2. Set the time in the **Select a time** window.
 - To apply the set time to the dashboard, check **Use dashboard settings**. The time setting on the dashboard is located on the upper right.
 - To apply a desired time setting, uncheck **Use dashboard settings**. Use < or > to set the desired time.
3. After setting the time, select ✕.

ⓘ Depending on the data type of the widget, the time setting may not be available.

Setting the output data options

By setting the output data options of the chart widget, you can select the desired data.

1. At the top right of the widget, click the ⋮ icon to configure the output data options.
 - You can filter data by selecting types for project, agent, application, etc.
 - You can change the type of the graph displayed on the chart by selecting individual or merged data.
 - Select ⋮ > ⚙️. You can change the option details through the **Chart settings** window. The options that can be set are as follows.
 - **Chart type:** This option allows you to display the current aggregated figures on the chart.
 - **y axis Chart max value:** The chart maximum value on the Y axis can be selected with one of **Auto** and **Fix**.
 - **Set horizontal line:** Select **Add** to display the threshold or range as a horizontal line. You can

apply multiple items added. After adding the horizontal line, select **Apply** to apply it to the chart.

- **Show chart max value:** The maximum value of the graph is displayed on the chart.
 - **Sub-chart:** You can add a sub-chart to the main charts of the widget. Select a type of sub-chart and then select its layout, size, and whether or not to merge data. To apply to other widgets, select **Apply to the entire widget**. Sub-charts are useful to view data in various ways.
2. After setting all options, on the left of the **Chart settings** window, click **×** or click the outside of the **Chart settings** window.

ⓘ Depending on the widget's data type, available options may differ.

Changing the metrics widget data

You can change the data query conditions of the metrics widget.

1. In **Flex Board > Dashboard list**, select the dashboard where you want to change the widget data.
2. In **Edit Mode**, click the **⋮** icon at the top right of the widget you want to modify.
3. Click the **<>** icon to open the data settings edit window.
4. Change the settings referencing the following, and then select **< Apply to the widget**.
 - Metrics options that are created by default when creating a widget appear on the tab.
 - Metrics options are **Category**, **Field**, and **Tag**.
 - The **Category** currently restricts the editing function.
 - In **Field** and **Tag**, multiple items can be selected. The chart data is grouped using the selected tag options.
 - Select **🔍** to query the original data.
 - For more information about **Object Merge** and **Time merge**, see the [Data merge options](#).

Data merge options

The metrics widget placed on the **All metrics** list of the Flex board provides the data merge options. Data merge provides the following methods: **Object Merge** and **Time merge**.

- **Object Merge**

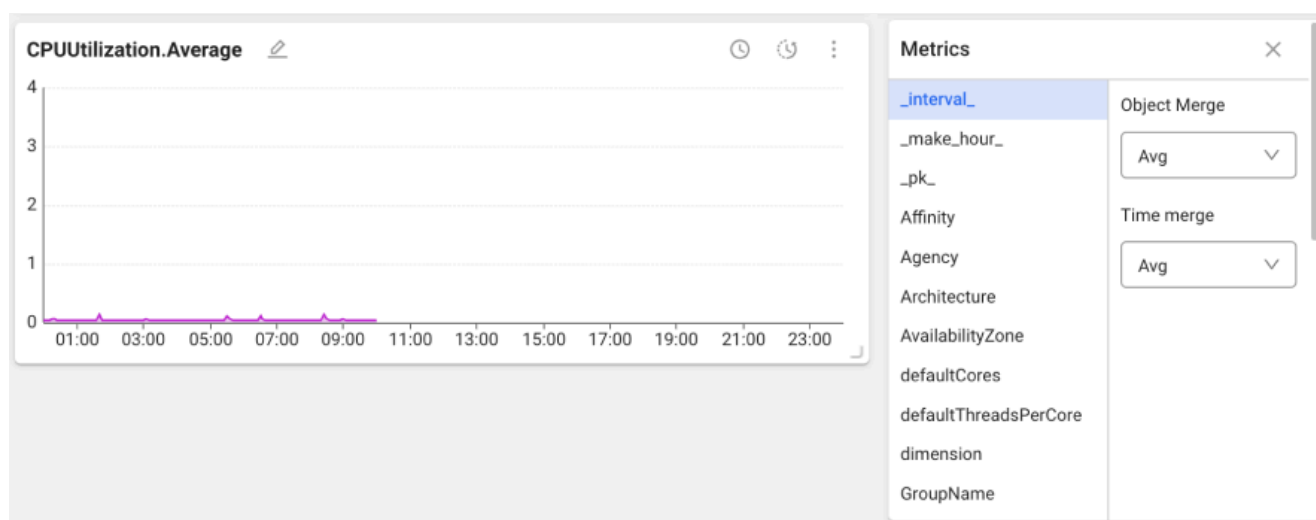
It is used to group the chart data based on the tag (e.g. agent name) selected on the left. At this time, if the tags match among the data with different fields (e.g. CPU), the corresponding data is merged.

- **Time merge**

It is mainly used if the lookup time is too long. Data with the same fields in the original data are merged at regular time intervals (5 minutes or 1 hour) depending on the time range. This method is called "Time Merge."

- **AWS ColudWatch Metrics**

In case of AWS monitoring metrics, the statistical method is added after the field name in the form of suffix. Data merging defaults reference this suffix.



1. In **Edit Mode**, select **All metrics** and then change the **Widget templates** list to the **All metrics** list.
2. Place the metrics widget on the layout.
3. On the upper right of the metrics widget, select **:**.
4. If a tag (e.g. agent name) option is selected, the data merge options appear. Depending on the widget size, the metrics option's location may differ.

Copying a widget

1. Click  on the upper right of the widget.
2. Click .
3. If the **Copy** window appears, click **+ Add widget**.
4. Change the widget name in the list.
5. Click **Apply**.
6. In **Edit Mode**, you can see the copied widget.

Delete Widget

1. Click  on the upper right of the widget.
2. Click .
3. If the confirmation message appears, click **Delete**. To cancel the deletion, select **Cancel**.

OpenMetrics Template List

We provide default templates and predefined Templates so that users who are new to **OpenMetrics** can quickly collect and visualize metrics without complex configuration. This lets you easily apply commonly used open source metrics, lowers the initial adoption barrier, and helps operators build a scalable monitoring environment.

Template list

Template List provides default open-source template cards.

Each card includes a brief description, a link to the template, and collection guidance.

- → **Template:** Click to navigate directly to the corresponding Template.
- ⓘ **Collection guide:** Click to see how to configure data collection.

Nginx Template

Monitor Apache Nginx to track and analyze web server performance in real time.

Istio Template

Visualize service relationships and traffic flow in an Istio service mesh, and provide an integrated monitoring environment by linking with WhaTap APM data. Use the button at the top right to zoom in or out of the Template.

OpenMetrics Template istio

Legend

Click the ⓘ button at the top right to view the legend.

Node shapes

- Triangle: Kubernetes service
- Circle: Workload

- Rectangle: Application
- Small circle: Pod

Node status

Error ratio within all HTTP requests and responses (4xx, 5xx)

- **Green**: error rate below 5%
- **Orange**: error rate 5% or higher
- **Red**: error rate 10% or higher
- Gray: no requests

Pod node status

Kubernetes Pod phase

- **Green**: Pods in Running state
- **Orange**: Pods in Pending state
- **Red**: Pods in Failed state

Edge colors

Error ratio of HTTP requests between two nodes (4xx, 5xx)

- **Green**: error rate below 10%
- **Orange**: error rate 10% or higher
- **Red**: error rate 20% or higher
- **Blue**: TCP communication
- Gray: unknown status

OpenMetrics Explorer

OpenMetrics Explorer allows you to directly explore custom Prometheus-based metrics using PromQL, visualize them in various formats, and check the status of metrics or gain insights.

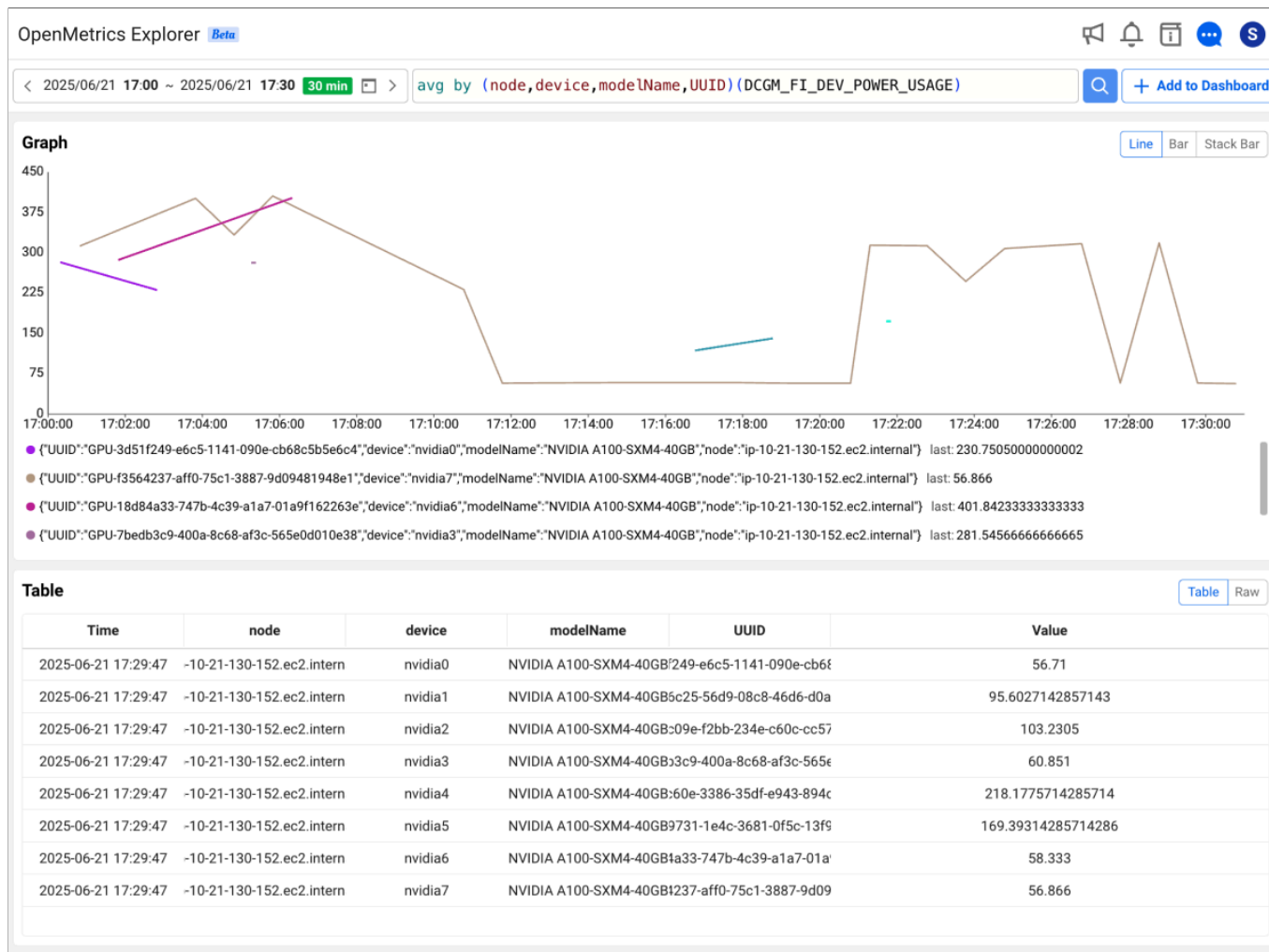
With this feature, you can freely query the desired metrics, extract meaningful data, or link them to dashboards for continuous monitoring.

ⓘ To use the **OpenMetrics Explorer**, the Open Agent must be installed.

- Refer to the [Install WhaTap Operator](#) documentation.

Explore with PromQL

In **OpenMetrics Explorer**, you can run PromQL queries and view the results with various visualization options.



Graph

- **Line:** Basic time series line graph
- **Bar:** Visualized as a bar chart
- **StackBar:** Stacked bar chart showing cumulative values

Table / Raw

- **Table:** Displays values of each time series in chronological order as a table
- **Raw:** Shows the Prometheus API response (JSON format) directly

PromQL Syntax Support and Autocomplete

When entering metrics such as `node_cpu_seconds_total` or `container_memory_usage_bytes`, autocomplete provides suggestions for metrics and operators.

Label filtering in the form of `{pod="example"}` is also supported.

Add to Dashboard

You can save the queried PromQL as a dashboard widget to continuously monitor desired metrics.

How to Add

1. Click the **+ Add to Dashboard** button in the upper-right corner of the **OpenMetrics Explorer** screen to open the **Add to Dashboard** modal.
2. In the **Add to Dashboard** modal, select an existing dashboard or create a new one.
3. Click the **Apply** button to add the widget to the selected dashboard.
 - You can enter the widget name and description in the added dashboard.
 - The added widget can be freely repositioned in the Flexboard grid layout, and the PromQL can be modified.

Cluster Summary

Home > Project Select > **Cluster** > **Cluster Summary**

ⓘ Requirements

You can see the **Cluster Summary** menu only for the Kubernetes cluster project (**CP**).

You can check summary information about the cluster. You can integrate the data separated by the namespace project.

Cluster object

You can check versions related to the cluster and the number of Kubernetes objects such as nodes, Pods, and containers within the cluster.

- ⓘ • You can check the [cluster version](#) when the Kubernetes agent version is 1.4.1 or later.
- If the Kubernetes agent version is 1.3.6 or later, you can check the following:
 - [kubelet Version](#)
 - [Container Runtime](#)



Cluster Version

- Data at an interval of 5 seconds. When viewing real-time data, the data is called once every 5 seconds.

kubelet Version

- Data at an interval of 5 seconds. When viewing real-time data, the data is called once every 5 seconds.

Container Runtime

- Data at an interval of 5 seconds. When viewing real-time data, the data is called once every 5 seconds.

Number of Containers

- You can check the number of containers by status.
- Data at an interval of 5 seconds. When viewing real-time data, the data is called once every 5 seconds.

Number of Pods

- You can check the number of Pods by status.
- Data at an interval of 5 seconds. When viewing real-time data, the data is called once every 5 seconds.

Number of nodes

- You can check the number of nodes by status.
- Data at an interval of 5 seconds. When viewing real-time data, the data is called once every 5 seconds.

Kubernetes Object list

- You can see the list of Kubernetes objects.
- Data at an interval of 5 minutes. When viewing real-time data, the data is called once every 3 minutes.

Object count trend

- You can see how the number of Kubernetes objects has changed over the past 24 hours.

- Data at an interval of 5 minutes. When viewing real-time data, the data is called once every 3 minutes.

Cluster Integration Information

You can see cluster project information and the Pods, containers, logs, event count, and event list separated by the namespace project.

ⓘ To view the data, the **Read** role is required for the namespace project.

CP

Kubernetes Monitoring Demo(EKS) ▾ Cluster Summary

Cluster Object

Cluster Integration Information

II

LIVE 12:28:47

Number of Pods/Containers

| Project | Namespace | Number of Containers | | Number of Pods | |
|---------------------------------------|-------------------|----------------------|--------------|----------------|--------------|
| | | Total | By State | Total | By Phase |
| 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 | | | | |

Number of Logs/Events

| Project | Namespace | Number of Logs | | Kubernetes Number of Events | | WhaTap Number of Alarms | |
|---------------------------------|-------------------|----------------|---------------|-----------------------------|---------------|-------------------------|---------------|
| | | Last hour | Last 24 hours | Last hour | Last 24 hours | Last hour | Last 24 hours |
| K8s Namespace Project DEH | virtual | 15,089,594 | 186,103,131 | 0 | 0 | 0 | 0 |
| Kubernetes Monitoring Dem
CP | amazon-guardduty | 7,567,270 | 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 Event 5

| Time | Project | Name | Type | Kind | Reason | Message |
|---------------------|---------------------------|----------------------------|---------|------|------------------|---|
| 2023/11/22 12:21:49 | Kubernetes Monitoring ... | whatap-virtual-deploym... | 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-deploym... | 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-deploym... | Warning | Pod | FailedScheduling | 0/3 nodes are available: persistentvolumeclaim "premdata-pvc" not f |

Number of Pods/Containers

You can check the container count and Pod count on a project basis.

- [Number of Containers](#)
 - [Total](#): Displays the total number of containers belonging to the cluster.
 - [By State](#): Displays the number of containers in progress by status.
- [Number of Pods](#)
 - [Total](#): Displays the total number of Pods belonging to the cluster.
 - [By Phase](#): Displays the number of Pods in progress by status.

Number of Logs/Events

You can check the numbers of logs, Kubernetes events, and WhaTap alarms on a project basis.

- [Number of Logs](#)
 - [Last 1 hour](#): Displays the number of log lines accumulated in the last hour.
 - [Last 24 hours](#): Displays the number of log lines accumulated for the day.
- [Kubernetes Number of Events](#)
 - [Last 1 hour](#): Displays the number of event lines accumulated in the last hour.
 - [Last 24 hours](#): Displays the number of event lines accumulated for the day.
- [WhaTap Number of Alarms](#)
 - [Last 1 hour](#): Displays the number of alarms delivered in the last hour.
 - [Last 24 hours](#): Displays the number of alarms sent for the day.

Kubernetes Event 5

You can see a list of 5 most recent Kubernetes events that occurred in the cluster project and namespace project.

Namespace Status

Home > Select Project > **Cluster** > [Namespace Status](#)

! If it is a Kubernetes cluster project (CP), you can view the corresponding menu.

Through [Namespace Status](#), you can see the WhaTap Kubernetes cluster project and sub namespace projects. This menu cannot be accessed by the namespace project.

| K8s Namespace Project DEMO | | | | | | | |
|----------------------------|-------|-------------|----|------------|---|--|--|
| Node | 3 | Container | 57 | Namespace | 6 | | |
| Node CPU core | 6 | Pod | 39 | Deployment | 9 | | |
| Node me... | 24.5G | Application | 10 | ReplicaSet | 9 | | |

Kubernetes Demo Project 33194

The [Namespace Status](#) menu is similar to the project list screen, but has the following differences:

- **Project list**

After creating a namespace project, you can move the project to another group, but the namespace is difficult to see at a glance.

- **Namespace status**

- Namespaces that do not belong to the same group as the cluster project can also be checked.
- You can see all related projects at a glance.

In case of the cluster project, the blue **Kubernetes** tag appears that indicates the Kubernetes platform. In the namespace project, the **Namspace** tag appears. The title on the upper left of the widget is the project name, and the namespace name appears next to the **Namspace** tag at the bottom.

Node List

Home > Project Select > **Node** > **Node list**

 If the data for specific columns does not appear, your Kubernetes node agent may be outdated. In this case, update the Kubernetes agent to the 1.3.7 or later so that enough data can be collected.

You can check the node information of the current project in a list.

- Real-time data is updated every 10 seconds.
- It provides the functions for node status summary, comparison, filtering, and [CSV](#) file download.

CSV file name, `kubernetes_node_list_{pcode}{timestamp}.csv_`

Basic screen guide

CP EKS-DEMO ▾ Node List

II LIVE 10:28:24 All 5 Ready 5

⌵ + There is no filter applied.

Select Column

Download CSV

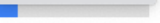
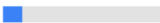
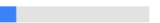
| Node info | | | | | | | | | cluster project c | |
|--------------------------|-------|-------|-------------------------|-------------------|------------------------|----------------|---------------------|------|---------------------|---------|
| Co... | De... | St... | OnodeName | Uptime | Age | Image | ContainerRuntime | Pods | TotalContainerCount | TotalCo |
| ☐ | | | ip-10-21-110-132.ap-... | 49d 19h 23min 09s | 49d 19h 22min 31s 7... | Amazon Linux 2 | containerd://1.7.11 | 16 | 31 | |
| ☐ | | | ip-10-21-110-178.ap-... | 7h 05min 26s | 7h 04min 48s 76ms | Amazon Linux 2 | containerd://1.7.11 | 5 | 26 | |
| ☐ | | | ip-10-21-110-253.ap-... | - | 1d 23h 53min 39s 76... | Amazon Linux 2 | containerd://1.7.11 | 8 | - | |
| ☐ | | | ip-10-21-110-36.ap-n... | 22h 51min 34s | 22h 50min 52s 76ms | Amazon Linux 2 | containerd://1.7.11 | 14 | 21 | |
| ☐ | | | ip-10-21-140-131.ap-... | 15h 22min 40s | 15h 22min 07s 76ms | Amazon Linux 2 | containerd://1.7.11 | 11 | 25 | |

You can see data by selecting a desired option at the top of **Node list**.

- You can see the number of all nodes and the numbers of nodes for each state.
- If you select a desired column, you can search by sorting **Node list** based on the selected column.
- By clicking the expand button on the left of the **Node list** column, you can see the disk information.

Metric information

- If you place your mouse cursor over the desired column, the metrics provided by the column are shown as follows:

| | 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/ |

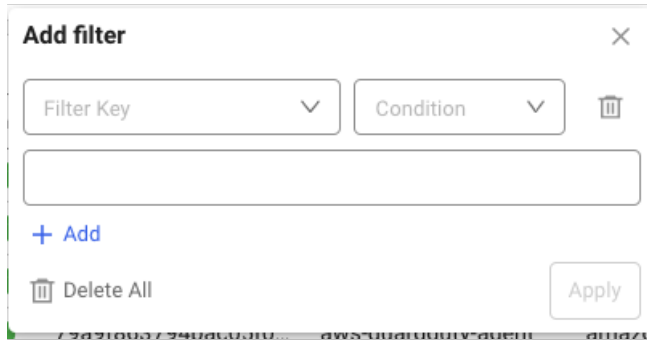
- In the column list of the [Select Column](#) option, you can see the metrics for each column in the column list of the option screen. When you place the mouse cursor over a desired column, the  **INFO icon** appears to the right of the column name. If the mouse cursor is placed over the  **INFO icon**, you can see the metric as follows:

| | | | |
|--|--|---|--|
| ☐ OS | ☐ CpuLoad15 | ☐ MemorySwapTot... | ☐ UsedInodePercent |
| ☒ Image | ☐ CpuNice | ☐ MemorySwapUs... | ☐ IoPercent |
| ☐ Kernel | ☐ CpuSoftIrq | ☐ MemoryTotal | ☒ IoPercentMax |
| ☐ KubeProxy | ☐ CpuSteal | ☒ MemoryUsedPct | ☐ QueueLength |
| ☒ Kubelet | ☒ CpuUsr | | ☐ ReadBps |
| ☒ ContainerRuntime | ☒ Field Unit pct | | ☐ ReadIops |
| | ☒ CpuUsr  | | ☐ TotalSpace |

Filtering

By using the [Filter](#) option, you can select a desired Node list for view. Through the [and](#) condition, multiple filters can be applied.

1. If you select + [Add icon](#) in the [Filter](#) field, the [Add filter](#) window appears as follows:



- Select [Filter Key](#) and [Condition](#).
 - Select a value that meets the conditions.
2. Configure a filter and then select [Apply](#) to apply the filter.
 3. To add multiple filters, select + [Add icon](#) in the [Filter](#) field or [Add filter](#) window.

Select Column

When using the [Select column](#) option, you can select the desired Node data to view the list.

Select Column

CSV Download

→ View previous version

Deselect all(10) Initialize(default)

☐ Node Info

☐ CapacityCpu

☒ Uptime

☒ Age

☐ Architecture

☐ CapacityMemory

☐ OS

☒ Image

☐ Kernel

☐ KubeProxy

☒ Kubelet

☒ ContainerRuntime

☐ CPU

☐ Cpuidle

☐ CpuIOWait

☐ CpuIrq

☐ CpuLoad1

☐ CpuLoad5

☐ CpuLoad15

☐ CpuNice

☐ CpuSoftIrq

☐ CpuSteal

☐ CpuSys

☒ Cpu...

☐ Memory

☐ MemoryBuffers

☐ MemoryCached

☐ MemoryFree

☐ MemoryPageFau...

☐ MemoryShared

☐ MemorySwapTot...

☐ MemorySwapUs...

☐ MemoryTotal

☒ MemoryUsedPct

☐ Disk

☐ BlkSize

☐ FreePercent

☐ FreeSpace

☐ TotalInode

☐ UsedInode

☐ UsedInodePercent

☐ IoPercent

☒ IoPercentMax

☐ QueueLength

☐ ReadBps

☐ ReadIops

☐ Network

☐ DroppedIn

☐ DroppedOut

☐ ErrorIn

☐ ErrorOut

☐ HwAddr

☐ PacketIn

☐ PacketOut

☐ TrafficUsedSize

☒ TrafficUsedSize...

☐ TrafficIn

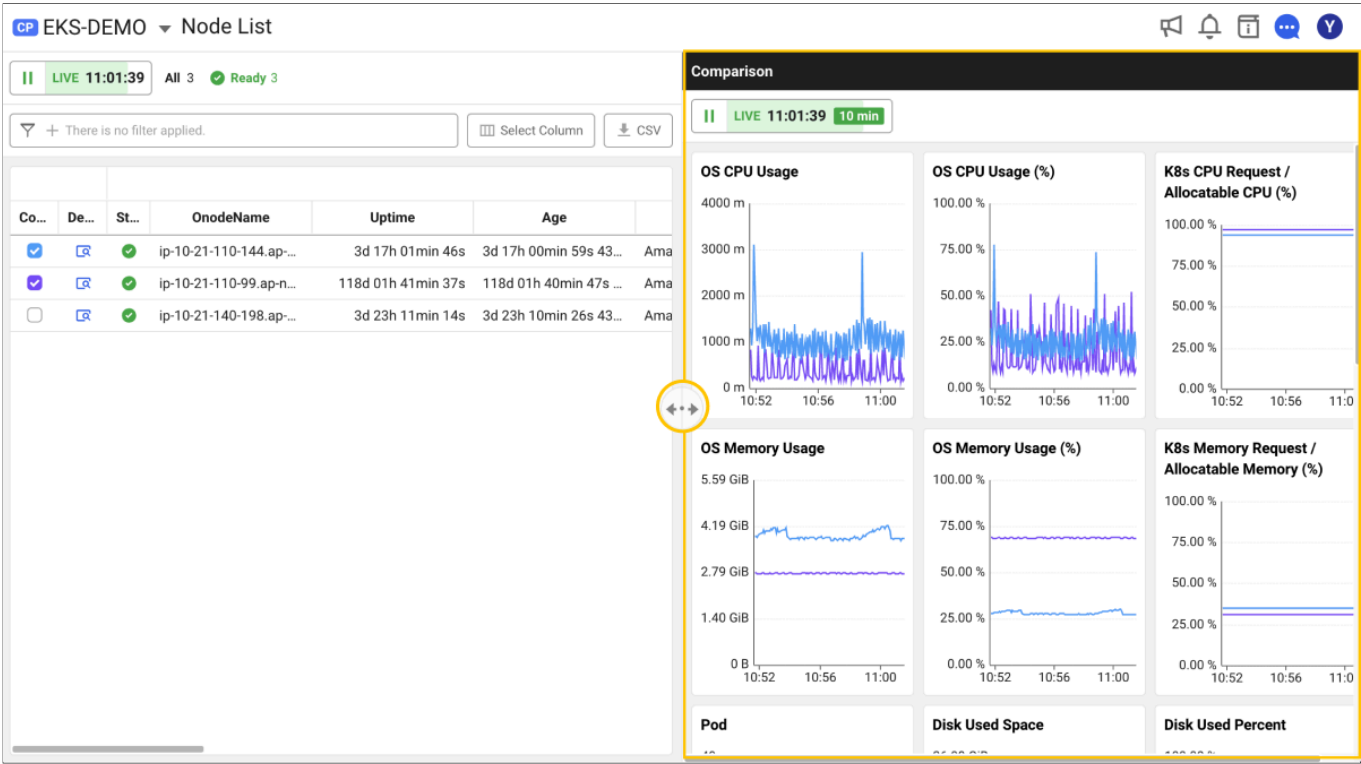
☐ TrafficOut

Apply

- The selected column list is kept even if you re-enter the menu.
- If you select [Reset](#), the predefined default key metric columns are selected. In this case, the column selection history is deleted.
- After selecting a column, you can click [Apply](#) at the lower right to apply the selected values.

Comparing

The [Compare](#) option allows you to see a comparison chart of key metrics for multiple nodes. When you select the [Compare](#) check boxes of the nodes to view in the list, a window comparing details for the nodes appears on the right as shown below.



View Detailed

Node details can be checked through the [Detail](#) option for details on the selected node. If you select the icon in the [Detail](#) column of the node to view in the list, the detailed information window for the node appears on the right.

Node configuration

It provides a summary of the node configuration and overall resource usage selected in the **Node configuration** tab. You can comprehensively see the hardware and software components of the node. Users can monitor resource usage of nodes and the status of each component, as well as performance and operational stability.

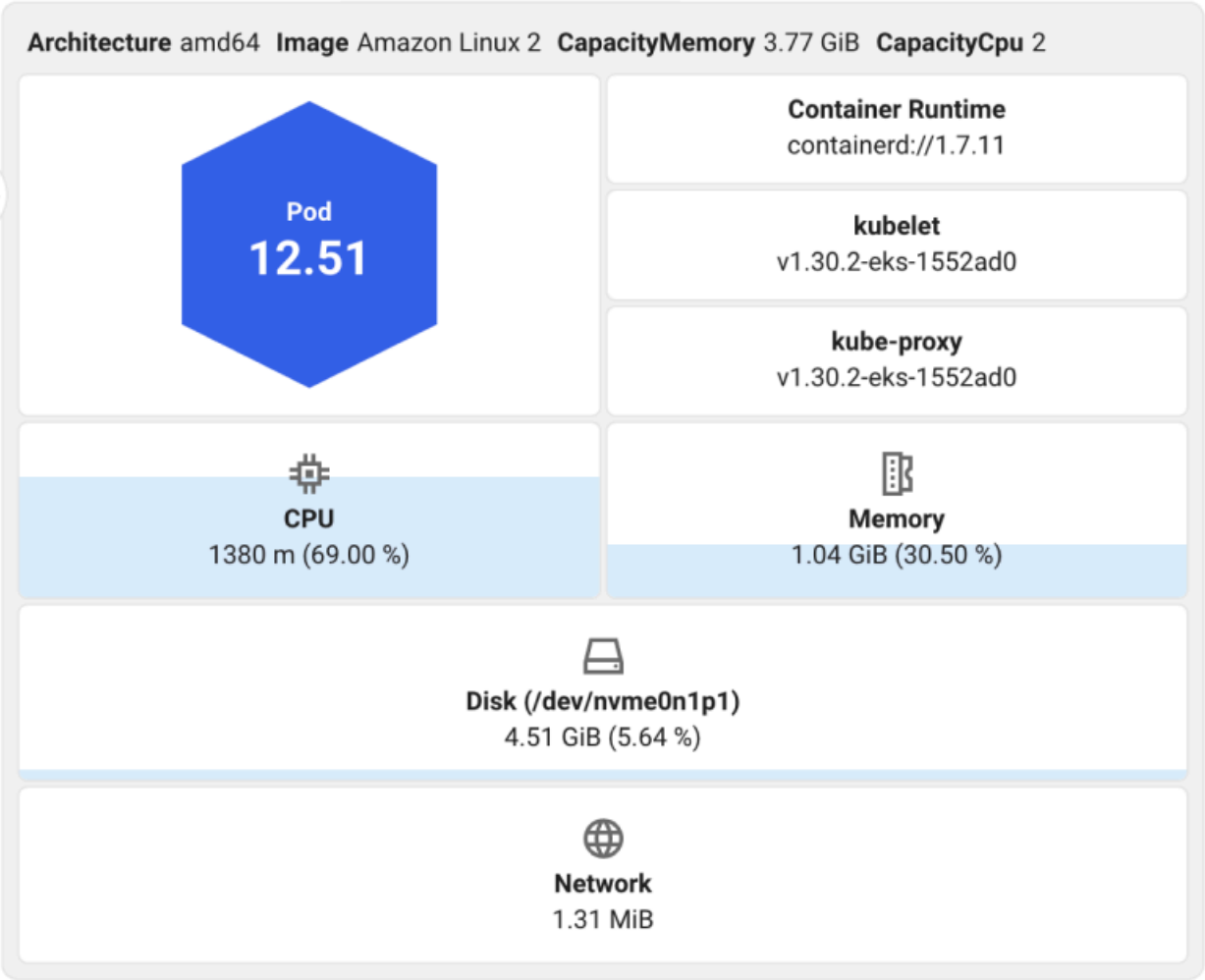
View time: 2024/09/20 14:40:30 ~ 2024/09/20 18:11:25

 This is a summary analysis based on statistical data that can be used as reference material for detailed analysis or system improvement.

ip-10-21-140-100.ap-northeast-1.compute.internal is an amd64 linux node with 2 CPUs and 3.77 GiB memory. This node maintains light usage most of the time, but **occasionally experiences moderate load on CPU, Memory**. If you want stable operation, we recommend periodic monitoring in preparation for when load occurs.

✓ **Network usage** is up to 1.31 MiB, which is **24.69 times higher than the average usage, 54.44 kiB, so you need to keep an eye on the volatility of traffic patterns.**

AVG **MAX**



- Node system specifications

- [Architecture](#): Hardware architecture used by the node.
- [Image](#): Information about the operating system image that the node is running on.
- [CapacityMemory](#): Total memory capacity of the node.
- [CapacityCpu](#): Total CPU capacity of the node.
- **Software configuration**
 - [Container Runtime](#): Version of the container runtime used on the node.
 - [kubelet](#): Version of the Kubernetes kubelet running on the node.
 - [kube-proxy](#): Version of the kube-proxy that manages the network features.
- **Resource usage status**
 - [CPU](#): CPU usage of the node.
 - [Memory](#): Memory usage of the node.
 - [Disk](#): Disk capacity and usage rate on the node.
 - [Network](#): Network throughput

- ⓘ • If you click CPU, memory, disk, or network, the **Metrics** tab appears that can display the usage trend for the query period.
- By clicking [AVG](#) and [MAX](#), you can see the average usage and maximum usage for the query period.

- ⓘ The [Node configuration](#) tab is available in case of a cluster project ([CP](#)).

Resource

Resource In the tab, you can see the list of [Pods](#) scheduled on the selected node and the [containers](#) and [applications](#) running in those Pods. The information appears reflecting 1 minute of resource data as of the end of the selected time zone. This allows you to more clearly identify the node status at the time point and the operation status of each Pod, container, and application.

| コンテナ Pod アプリケーション | | | | |
|--|-----------------------|-----------------------|---------------------------|-----------------|
| ① 他のプロジェクトで監視しているコンテナは表示されません。 | | | | |
| <div> <div>< 2024/09/20 18:11 ></div> <div>カラム選択</div> <div>全体 16</div> <div>🟢 RUNNING 16</div> </div> | | | | |
| CpuTotUsage 264.12 m CpuRequest 1705 m CpuLimit 20865 m MemWs 466.14 MiB
MemRequest 2.86 GiB MemLimit 28.59 GiB | | | | |
| コンテナ情報 | | | | |
| Co... | ContainerId | ContainerName | Image | ApplicationName |
| 🟢 | 893c0db6b6b2ce6ae... | aws-node | 602401143452.dkr.e... | - |
| 🟢 | 9b290ccdd32993641... | busybox-container | docker.io/library/bus... | - |
| 🟢 | ccb25d34326b94371... | busybox-container | docker.io/library/bus... | - |
| 🟢 | ce508e4f25a0efc51b... | ebs-plugin | 602401143452.dkr.e... | - |
| 🟢 | 520422821f75c78c3... | kube-proxy | 602401143452.dkr.e... | - |
| 🟢 | c29b3a9f711f962885... | kubecon2023 | 224983703333.dkr.e... | - |
| 🟢 | dcc500fbf6c9c9ef4b... | kubecon2023-logger | docker.io/library/bus... | - |
| 🟢 | 717fe5230f19455684... | liveness-probe | 602401143452.dkr.e... | - |
| 🟢 | 73868ad01bd28f336... | nginx | docker.io/library/ngin... | - |
| 🟢 | 3dd9c9268a2d291a2... | nginx-container | docker.io/library/ngin... | - |
| 🟢 | 3e2c44e0ba5f85743... | nginx-container | docker.io/library/ngin... | - |
| 🟢 | 590cf0e747483edfb1... | node-driver-registrar | 602401143452.dkr.e... | - |

① When the **Overview** column is selected, you can see the summary analysis for container and Pod.

Process

The **Process** tab provides information about the major Kubernetes management processes inside the selected node. It can prevent cluster failures caused by process issues and minimize the downtime.

< 2024/10/29 14:27 >

name

Column Settings

CSV

| De... | Nodelp | Pid | Name | CreateTime | CpuUsedPercent | MemoryUsedPercent | Rssl |
|-----------------------------------|--------------------------|--------|---------------|---------------------|--------------------|--------------------|------|
| <div><div></div><div></div></div> | ip-10-21-110-99.ap-no... | 2833 | containerd | 2024/10/29 09:20:07 | 0.40 % <div></div> | 1.09 % <div></div> | |
| <div><div></div><div></div></div> | ip-10-21-110-99.ap-no... | 2960 | kubelet | 2024/10/29 09:20:12 | 1.39 % <div></div> | 1.35 % <div></div> | |
| <div><div></div><div></div></div> | ip-10-21-110-99.ap-no... | 3237 | kube-proxy | 2024/10/29 09:20:22 | 0.10 % <div></div> | 0.60 % <div></div> | |
| <div><div></div><div></div></div> | ip-10-21-110-99.ap-no... | 3442 | aws-k8s-agent | 2024/10/29 09:20:25 | 0.10 % <div></div> | 0.68 % <div></div> | |
| <div><div></div><div></div></div> | ip-10-21-110-99.ap-no... | 122157 | coredns | 2024/10/29 13:50:56 | 0.20 % <div></div> | 0.64 % <div></div> | |
| <div><div></div><div></div></div> | ip-10-21-110-99.ap-no... | 122564 | coredns | 2024/10/29 13:51:03 | 0.20 % <div></div> | 0.63 % <div></div> | |

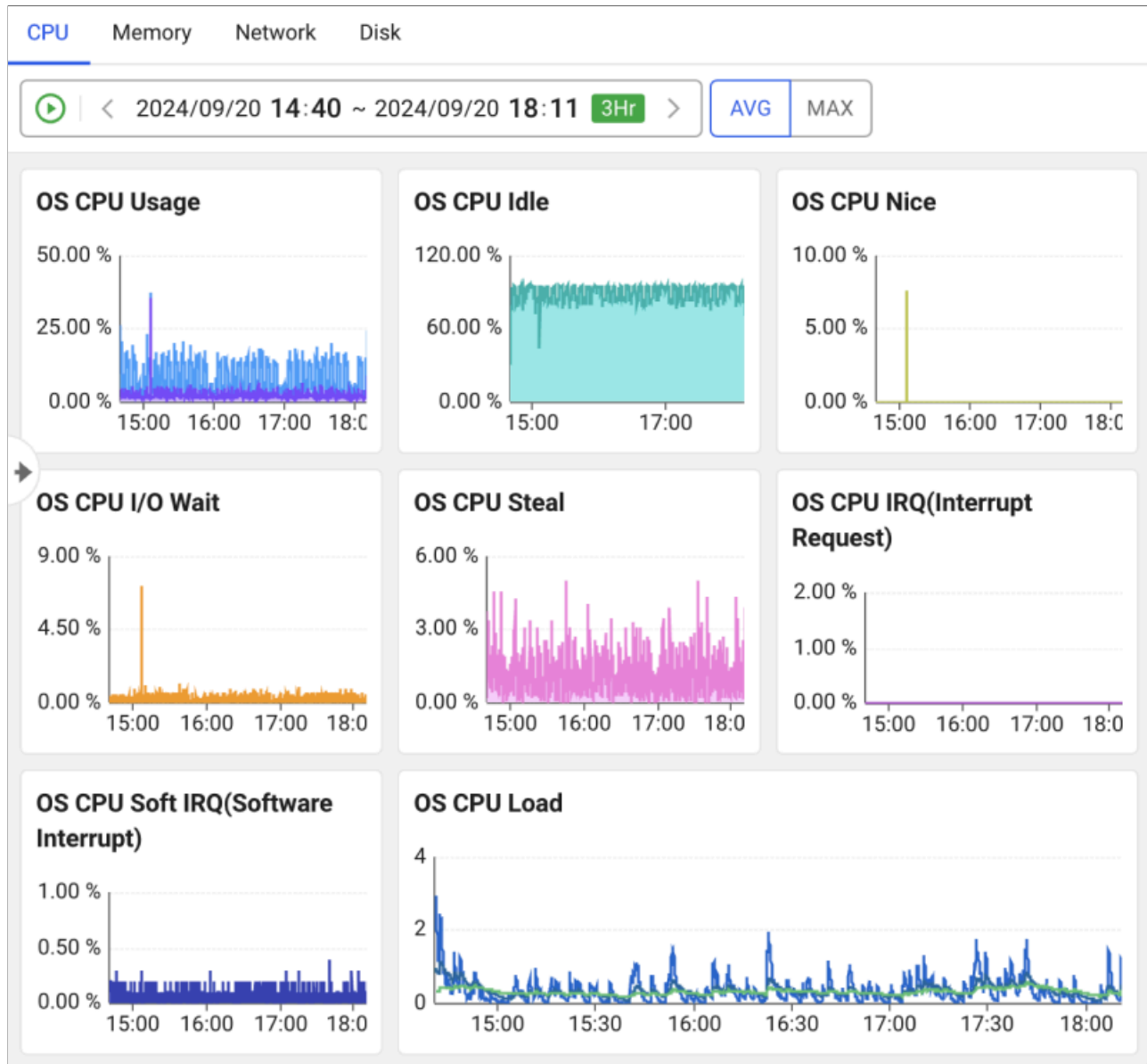
Process details

- You can check information details about a process by selecting the  icon in the **Detail** column on the left of the process list.
- You can select a method how to aggregate the statistical data during retrieval while querying long-term data of 3 days or more by clicking [AVG](#) or [MAX](#).
- When you click a process key data value, the value is copied.

 The **Process** tab is supported in Kubernetes agent 1.7.14 or later.

Metrics

You can check the resource usage trend of the node select in the **Metrics** tab. You can check the usage changes in [CPU](#), [memory](#), [network](#), and [disk](#) based on the average ([AVG](#)) and maximum ([MAX](#)) values during the query period.

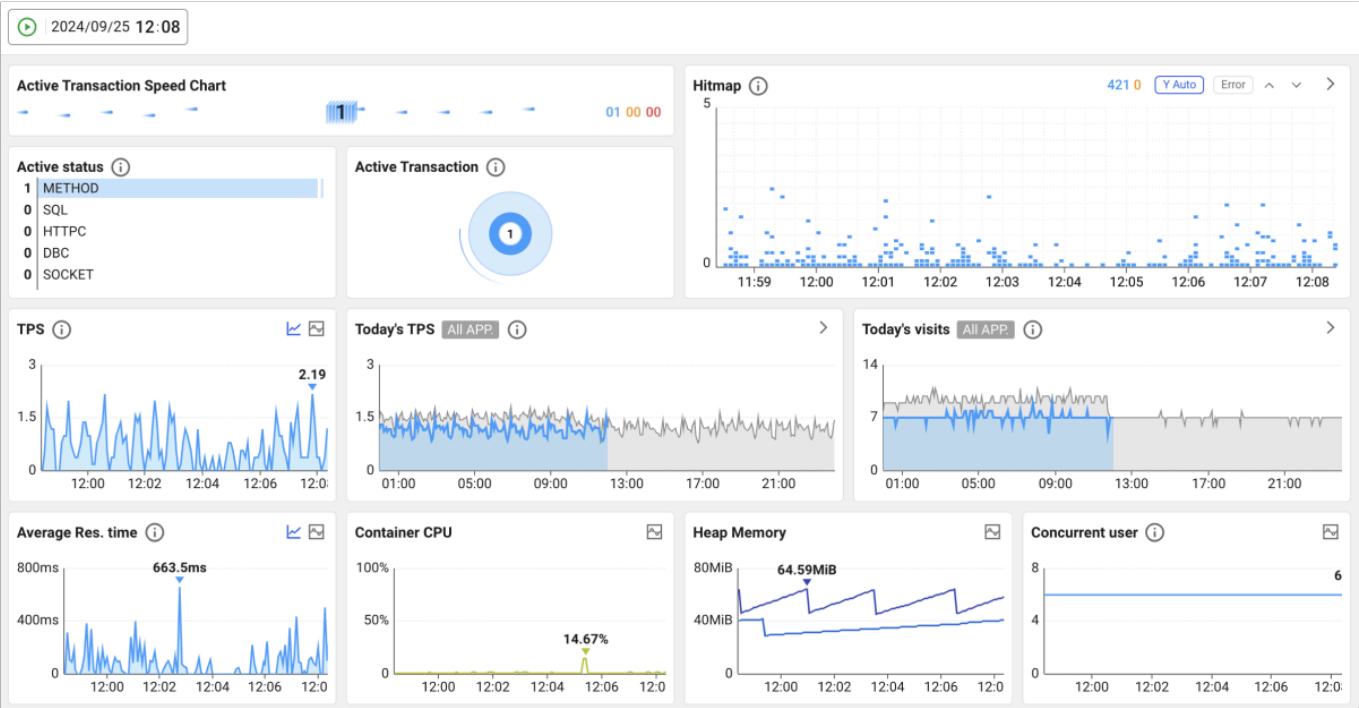


- **AVG**: Average usage for the query period.
- **MAX**: Maximum usage for the query period.
- **Total**: Total traffic usage across multiple network interfaces.
- **In/Out**: Inbound (receiving) and outbound (sending) traffic usage for a specific network interface.

ⓘ The **K8s CPU Request / Allocatable CPU (%)** widget is supported in Kubernetes agent 1.7.10 or later.

Trace

You can see the trace information of the applications running on the node selected in the **Trace** tab.



Event

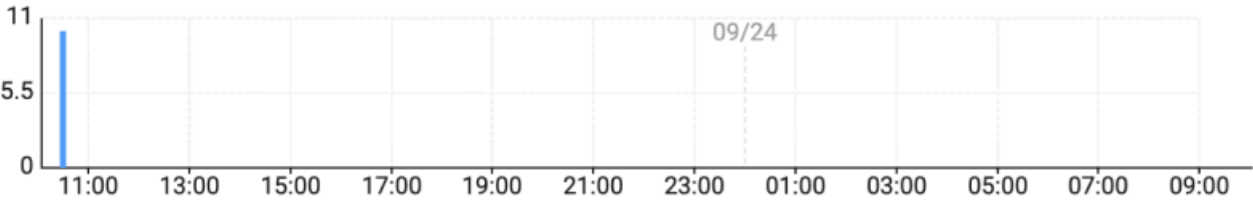
You can check [K8s Node events](#) and [WhaTap Node events](#) that occurred on the node selected in the **Event** tab. [K8s node events](#) include important activity logs within the cluster such as changes in node status, while [WhaTap node events](#) provide information about the events triggered based on the user-defined alert criteria.

K8s Node Events

WhaTap Node Events

< 2024/09/23 10:04 ~ 2024/09/24 10:04 1Day >

▼ Event count trend



▶

Normal

Warning

☐ Hide Tag

Relocate log menu >

| ▶ | Timestamp | Log |
|---|-------------------------|--|
| ▶ | 2024-09-23 10:34:30.136 | NodeReady Node ip-10-21-110-253.ap-northeast-1.compute.internal status is
<div>kind Node name ip-10-21-110-253.ap-northeast-1.compute.internal</div> |
| ▶ | 2024-09-23 10:34:27.295 | Starting
<div>kind Node name ip-10-21-110-253.ap-northeast-1.compute.internal</div> |
| ▶ | 2024-09-23 10:34:23.842 | RegisteredNode Node ip-10-21-110-253.ap-northeast-1.compute.internal ever
<div>kind Node name ip-10-21-110-253.ap-northeast-1.compute.internal</div> |
| ▶ | 2024-09-23 10:34:21.729 | Synced Node synced successfully
<div>kind Node name ip-10-21-110-253.ap-northeast-1.compute.internal</div> |
| ▶ | 2024-09-23 10:34:21.575 | NodeAllocatableEnforced Updated Node Allocatable limit across pods
<div>kind Node name ip-10-21-110-253.ap-northeast-1.compute.internal</div> |
| ▶ | 2024-09-23 10:34:21.531 | NodeHasSufficientPID Node ip-10-21-110-253.ap-northeast-1.compute.intern
<div>kind Node name in-10-21-110-253 an-northeast-1 compute internal</div> |

Node Timeline

Home > Project Select > **Node** > **Node Timeline**

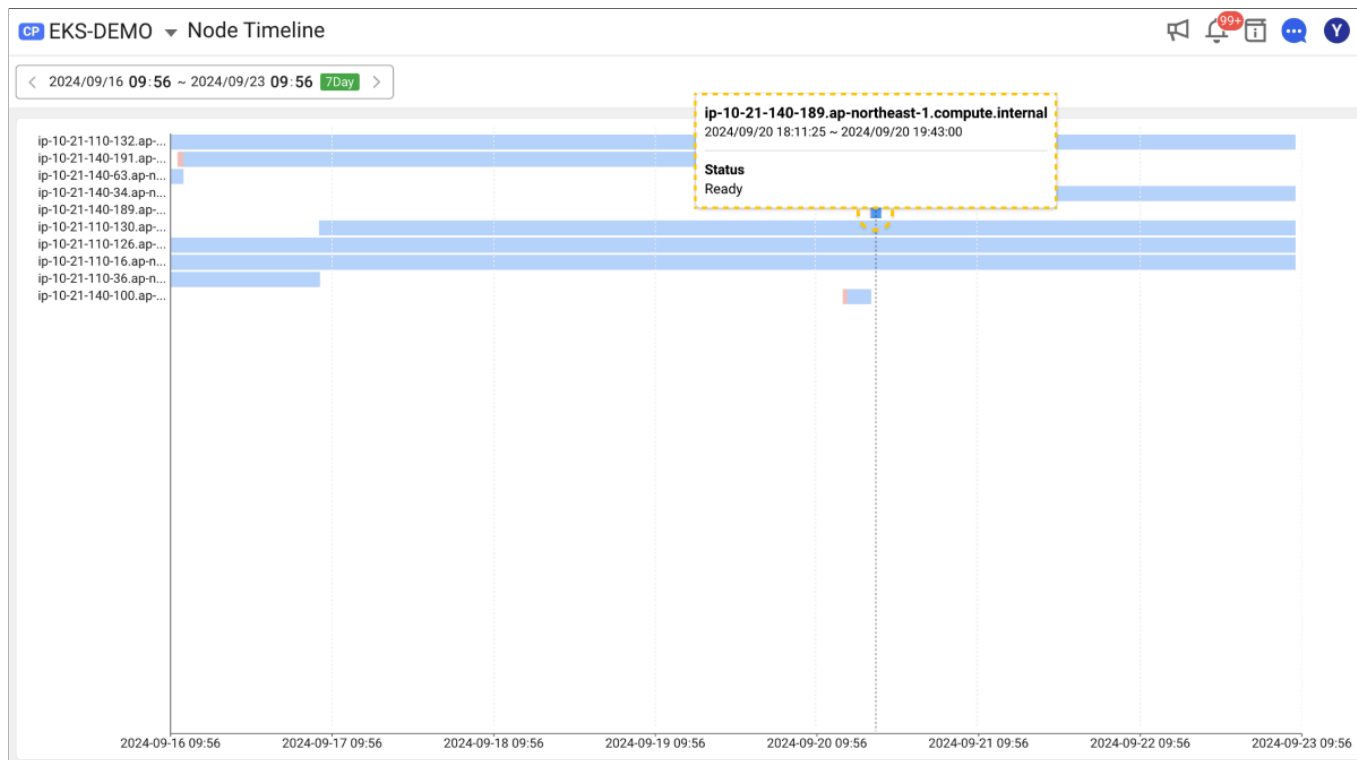
ⓘ Requirements

- The WhaTap Kubernetes agent 1.7.11 or later is required. Timeline data is provided from the time point collected in version 1.7.11 or later. It reflects the data collected after update of the agent.
- If it is a Kubernetes cluster project (**CP**), you can view the corresponding menu.

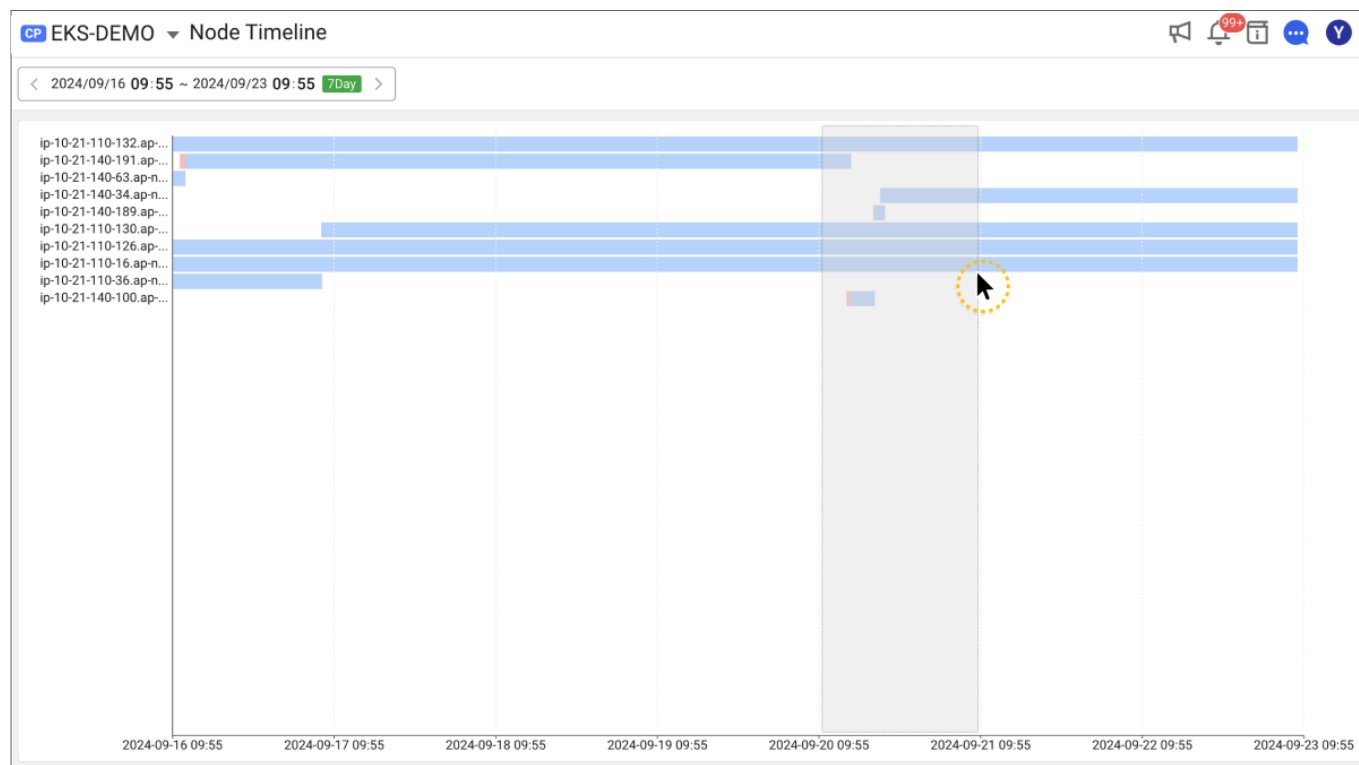
The **Node Timeline** menu provides the feature to trace changes in node state over time. Through this, you can visually identify when a disappeared node was removed and what state the node was in at that time. It is useful for analyzing issues that occurred during cluster operation or checking the past situations.

Basic screen guide

When you enter the **Node Timeline** menu, a list of nodes collected over the past 7 days is displayed as follows. You can visually see the state changes of nodes along the time axis. If you hover your mouse over each time zone, the start and end times for the zone appears as tooltips together with the node's status (**Ready**, **NotReady**, etc.).



You can adjust the time range by dragging the zone on the timeline. Set a desired period to check the past data.

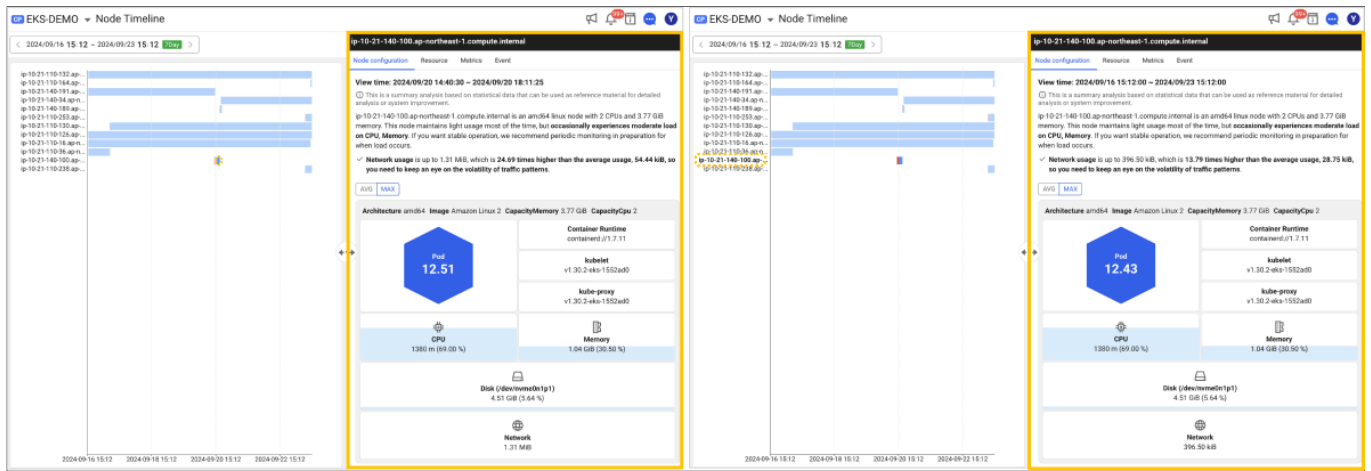


Additionally, when you select a specific period, you can check the node status during the period in detail through **Display Detail**. For more information, see the [Node timeline detail](#) guidance.

- For more information about the node status, see [the following](#).

Node timeline detail

Node timeline details can be checked by selecting a specific zone or node as follows:



- You can view detailed information of a node at the time point through the **Display Detail** window on the right by selecting a zone to view.
- You can select a node name to view from the list on the left of the chart and view the overall status of the node during the query period through the **Display Detail** window on the right.

Node configuration

It provides a summary of the node configuration and overall resource usage selected in the **Node configuration** tab. You can comprehensively see the hardware and software components of the node. Users can monitor resource usage of nodes and the status of each component, as well as performance and operational stability.

View time: 2024/09/20 14:40:30 ~ 2024/09/20 18:11:25

 This is a summary analysis based on statistical data that can be used as reference material for detailed analysis or system improvement.

ip-10-21-140-100.ap-northeast-1.compute.internal is an amd64 linux node with 2 CPUs and 3.77 GiB memory. This node maintains light usage most of the time, but **occasionally experiences moderate load on CPU, Memory**. If you want stable operation, we recommend periodic monitoring in preparation for when load occurs.

✓ **Network usage** is up to 1.31 MiB, which is **24.69 times higher than the average usage, 54.44 kiB, so you need to keep an eye on the volatility of traffic patterns.**

AVG

MAX

Architecture amd64 **Image** Amazon Linux 2 **CapacityMemory** 3.77 GiB **CapacityCpu** 2

Pod
12.51

Container Runtime
containerd://1.7.11

kubelet
v1.30.2-eks-1552ad0

kube-proxy
v1.30.2-eks-1552ad0


CPU
1380 m (69.00 %)


Memory
1.04 GiB (30.50 %)


Disk (/dev/nvme0n1p1)
4.51 GiB (5.64 %)


Network
1.31 MiB

- Node system specifications

- [Architecture](#): Hardware architecture used by the node.
- [Image](#): Information about the operating system image that the node is running on.
- [CapacityMemory](#): Total memory capacity of the node.
- [CapacityCpu](#): Total CPU capacity of the node.
- **Software configuration**
 - [Container Runtime](#): Version of the container runtime used on the node.
 - [kubelet](#): Version of the Kubernetes kubelet running on the node.
 - [kube-proxy](#): Version of the kube-proxy that manages the network features.
- **Resource usage status**
 - [CPU](#): CPU usage of the node.
 - [Memory](#): Memory usage of the node.
 - [Disk](#): Disk capacity and usage rate on the node.
 - [Network](#): Network throughput

! By clicking [AVG](#) and [MAX](#), you can see the average usage and maximum usage for the query period.

Resource

Resource In the tab, you can see the list of [Pods](#) scheduled on the selected node and the [containers](#) and [applications](#) running in those Pods. The information appears reflecting 1 minute of resource data as of the end of the selected time zone. This allows you to more clearly identify the node status at the time point and the operation status of each Pod, container, and application.

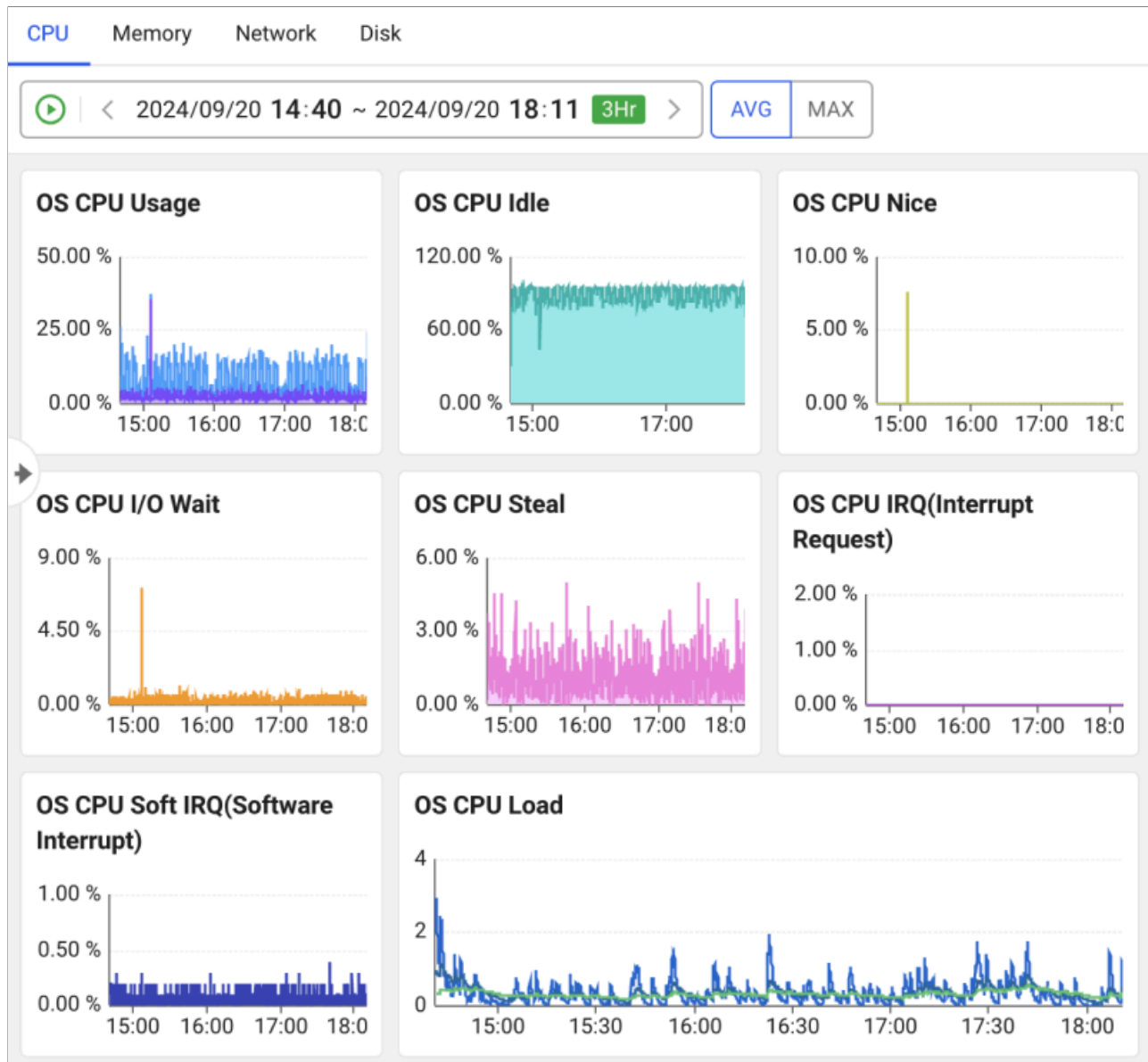
| コンテナ Pod アプリケーション | | | | |
|--|-----------------------|-----------------------|---------------------------|-----------------|
| ⓘ 他のプロジェクトで監視しているコンテナは表示されません。 | | | | |
| <div> <div>< 2024/09/20 18:11 ></div> <div>カラム選択</div> <div>全体 16</div> <div>🟢 RUNNING 16</div> </div> | | | | |
| CpuTotUsage 264.12 m CpuRequest 1705 m CpuLimit 20865 m MemWs 466.14 MiB
MemRequest 2.86 GiB MemLimit 28.59 GiB | | | | |
| コンテナ情報 | | | | |
| Co... | ContainerId | ContainerName | Image | ApplicationName |
| 🟢 | 893c0db6b6b2ce6ae... | aws-node | 602401143452.dkr.e... | - |
| 🟢 | 9b290ccdd32993641... | busybox-container | docker.io/library/bus... | - |
| 🟢 | ccb25d34326b94371... | busybox-container | docker.io/library/bus... | - |
| 🟢 | ce508e4f25a0efc51b... | ebs-plugin | 602401143452.dkr.e... | - |
| 🟢 | 520422821f75c78c3... | kube-proxy | 602401143452.dkr.e... | - |
| 🟢 | c29b3a9f711f962885... | kubecon2023 | 224983703333.dkr.e... | - |
| 🟢 | dcc500fbf6c9c9ef4b... | kubecon2023-logger | docker.io/library/bus... | - |
| 🟢 | 717fe5230f19455684... | liveness-probe | 602401143452.dkr.e... | - |
| 🟢 | 73868ad01bd28f336... | nginx | docker.io/library/ngin... | - |
| 🟢 | 3dd9c9268a2d291a2... | nginx-container | docker.io/library/ngin... | - |
| 🟢 | 3e2c44e0ba5f85743... | nginx-container | docker.io/library/ngin... | - |
| 🟢 | 590cf0e747483edfb1... | node-driver-registrar | 602401143452.dkr.e... | - |

ⓘ When the **Overview** column is selected, you can see the summary analysis for container and Pod.

Metrics

You can check the resource usage trend of the node select in the **Metrics** tab. You can see the usage changes in **CPU**, **memory**, **network**, and **disk** based on the average (**AVG**) and maximum (**MAX**) values during the query

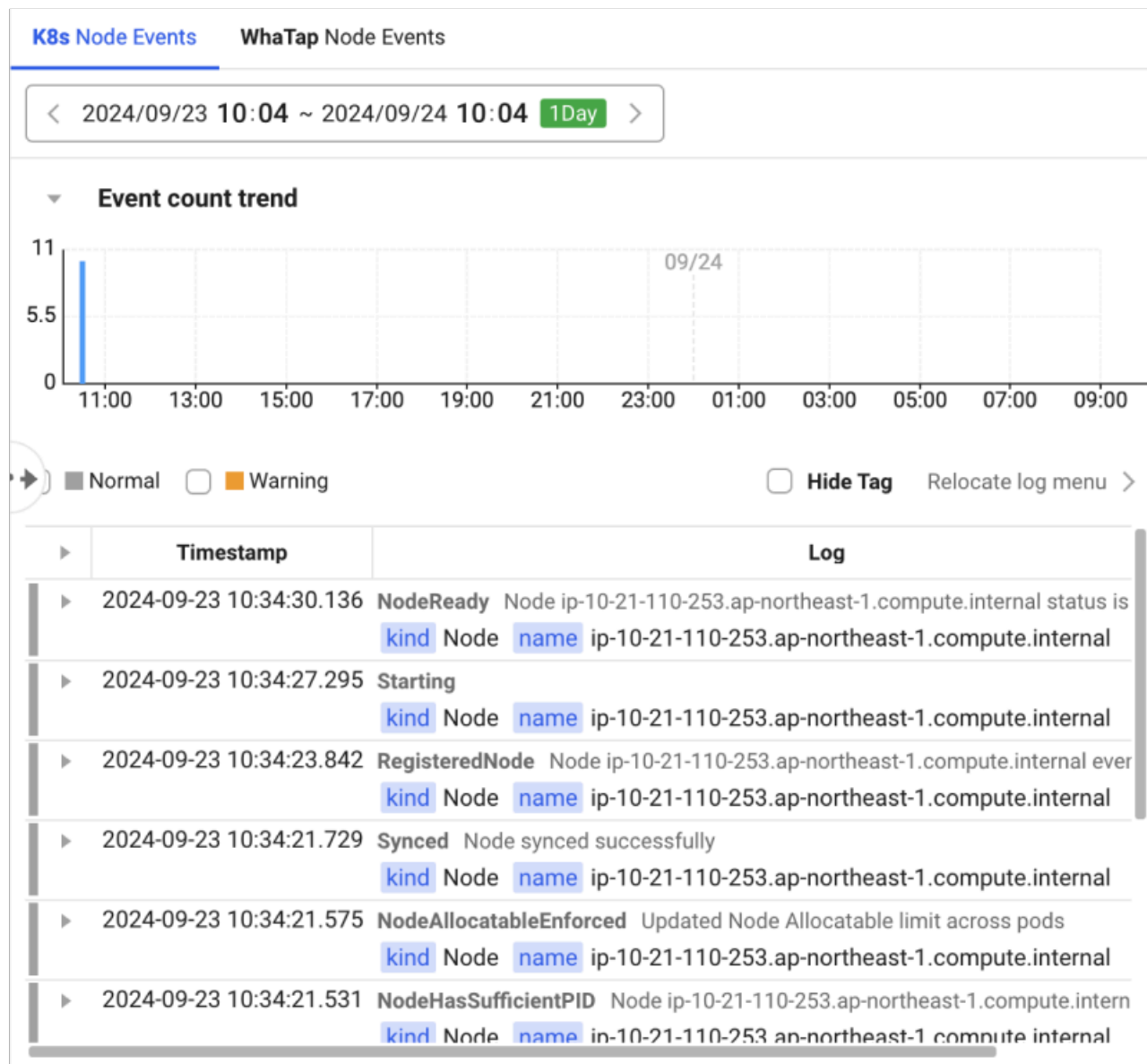
period.



- **AVG:** Average usage for the query period.
- **MAX:** Maximum usage for the query period.
- **Total:** Total traffic usage across multiple network interfaces.
- **In/Out:** Inbound (receiving) and outbound (sending) traffic usage for a specific network interface.

Event

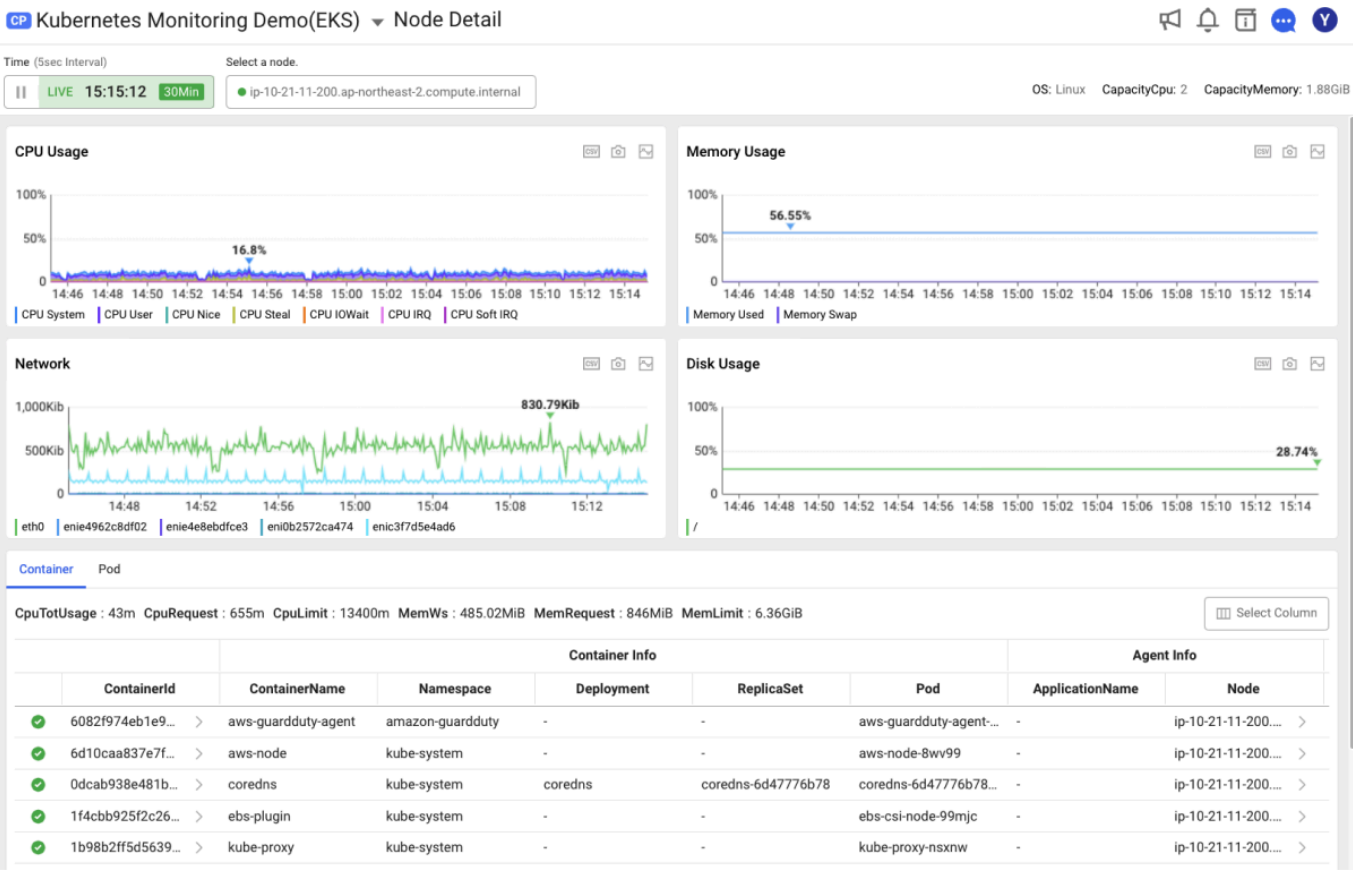
You can see [K8s Node events](#) and [WhaTap Node events](#) that occurred for the time of the timeline in the **Event** tab. [K8s node events](#) include important activity logs within the cluster such as changes in node status, while [WhaTap node events](#) provide information about the events triggered based on the user-defined alert criteria.



Node Details

Home > Project Select > **Node** > **Node details**

The **Node details** screen consists of four charts at the top and **container** and **Pod** details in the node. The node selector allows you to select a node to search.



- [CPU Usage](#): You can see the CPU usage of the node.
- [Memory Usage](#): You can see the memory usage of the node.
- [Network](#): You can see the input/output traffic and error packets of the node's network interface.
- [Disk Usage](#): You can see the disk usage of the node.

Container and Pod details

Container

When you select the [Container](#) tab, at the top of the container details within the node, you can see the sum of the following metrics: `cpu_quota` , `cpu_request` , `cpu_total_milli` , `mem_limit` , `mem_request` , and `mem_working_set` .

- If you select a desired column, you can search the container list sorted by that column.
- When the **Overview** column is selected, you can see **Container summary analysis**.
- If you select a container, the [Container list](#) menu appears. For more information, see [the following](#).

Pod

When you select the [Pod](#) tab, the detailed Pod information is viewed. At the top of the container details, you can see the sum of the following metrics: `cpu_quota` , `cpu_request` , `cpu_total_milli` , `mem_limit` , `mem_request` , and `mem_working_set` .

- If you select a desired column, you can search the Pod list sorted by that column.
- When the **Overview** column is selected, you can see **Pod summary analysis**.
- If you select a Pod, the [Pod list](#) menu appears. For more information, see [the following](#).

Node Disk List

Home > Project Select > **Node** > **Node Disk List**

⚠ Requirements

- The WhaTap Kubernetes agent 1.7.8 or later is required.
- If it is a Kubernetes cluster project (**CP**), you can view the corresponding menu.

It views **Node Disk List**. You can easily check the status of disks by monitoring the current usage, free space, and performance metrics of individual disks mounted on each node in real time in **Node Disk List** for WhaTap Kubernetes Monitoring. This allows you to efficiently manage disk resources in the cluster and proactively find and respond to potential problems.

Basic screen guide

You can check the list of nodes existing in the cluster and the list of disks mounted on the node as follows. The list is displayed based on the disk information collected over the last minute based on the query time.

CP Kubernetes Monitoring Demo(EKS) ▾ Node Disk List

⏸ LIVE 16:27:36

🔍 + There is no filter applied.

Select Column

📄 CSV

ⓘ If the mount point does not exist or disks that are added only after starting the WhaTap Kubernetes agent are not displayed in the list.

| Comparison | OnodeName | DeviceId | FileSystem | BlkSize | TotalSpace | UsedSpace | UsedPercent | FreeSpace |
|--------------------------|-----------------------|----------------|------------|----------|------------|-----------|---------------------|-----------|
| ☐ | ip-10-21-11-144... > | /dev/nvme0n1p1 | xfs | 4.00 kiB | 19.99 GiB | 9.30 GiB | 46.52 % <div></div> | 10.69 GiB |
| ☐ | ip-10-21-11-188... > | /dev/nvme0n1p1 | xfs | 4.00 kiB | 19.99 GiB | 5.20 GiB | 26.03 % <div></div> | 14.79 GiB |
| ☐ | ip-10-21-11-98.a... > | /dev/nvme0n1p1 | xfs | 4.00 kiB | 19.99 GiB | 4.75 GiB | 23.76 % <div></div> | 15.24 GiB |
| ☐ | ip-10-21-13-239... > | /dev/nvme0n1p1 | xfs | 4.00 kiB | 19.99 GiB | 4.64 GiB | 23.21 % <div></div> | 15.35 GiB |
| ☐ | ip-10-21-13-36.a... > | /dev/nvme0n1p1 | xfs | 4.00 kiB | 19.99 GiB | 8.26 GiB | 41.32 % <div></div> | 11.73 GiB |

Hidden node disk

When the WhaTap Kubernetes agent runs, the lists of disks mounted on each node appear. The disk does appear in the list in the following cases:

- The disk must have been mounted to a specific path to display in the list. **The disks that do not have any**

mount point do not appear in the list.

- The disks added after the WhaTap Kubernetes agent runs do not appear in the list.

Filtering

By using the [Filter](#) option, you can select a desired for view. Through the [and](#) condition, multiple filters can be applied.

1. If you select + [Add icon](#) in the [Filter](#) field, the [Add filter](#) window appears as follows:

- Select [Filter Key](#) and [Condition](#).
 - Select a value that meets the conditions.
2. Configure a filter and then select [Apply](#) to apply the filter.
 3. To add multiple filters, select + [Add icon](#) in the [Filter](#) field or [Add filter](#) window.

Select Column

When using the [Select column](#) option, you can select the desired data to view the list.

Select all Deselect all(8) Initialize(default)

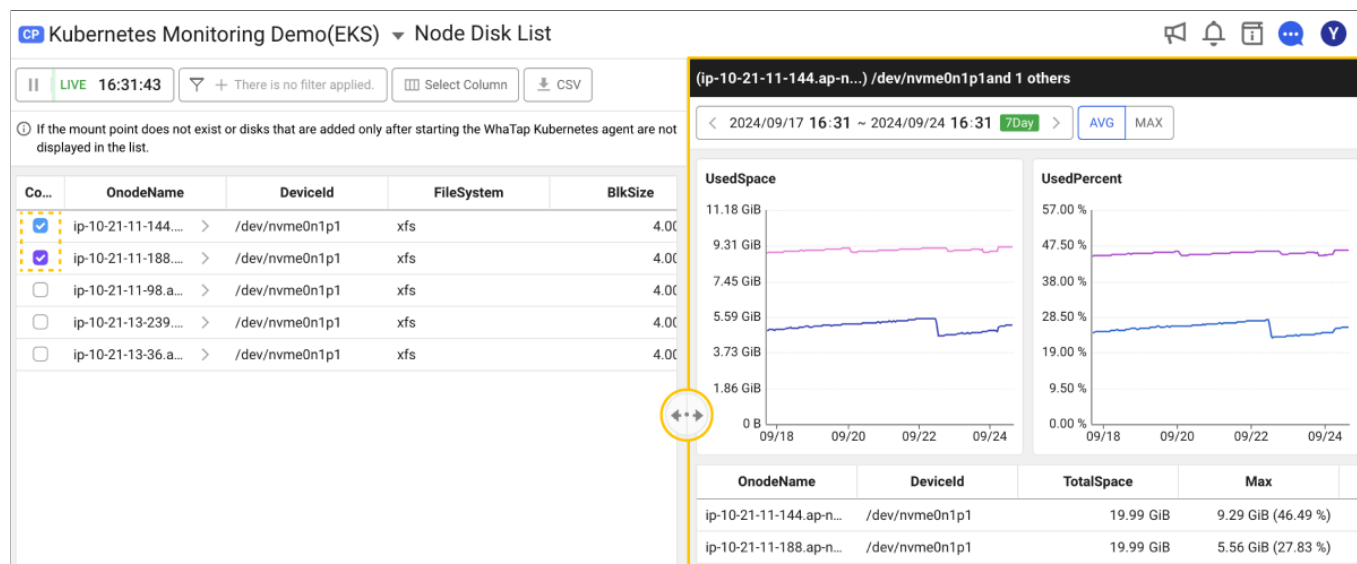
- ☒ OnodeName
- ☒ DeviceId
- ☒ BlkSize
- ☒ TotalSpace
- ☒ UsedSpace
- ☒ UsedPercent
- ☒ FreeSpace
- ☒ FreePercent
- ☐ TotalNode
- ☐ UsedNode
- ☐ UsedNodePercent
- ☐ IoPercent

Apply

- The selected column list is kept even if you re-enter the menu.
- If you select [Reset](#), the predefined default key metric columns are selected. In this case, the column selection history is deleted.
- After selecting a column, you can click [Apply](#) at the lower right to apply the selected values.

Comparing

Through the **Comparison** option, you can compare the disk usage. If you check in the **Comparison** column of the node disk to view in the list, a window comparing the details of the disks appears on the right.



For example, you can see the hourly usage and utilization for each disk through the [UsedSpace](#) and [UsedPercent](#) graphs. Using this, you can intuitively see the changes in disk usage over a specific time zone. You can also check detailed data such as maximum and minimum usage and increase/decrease rate for each disk through the detailed data list at the bottom. The columns are as follows:

- [OnodeName](#): It displays the name of the node to which the disk is connected.
- [DeviceId](#): It indicates the device ID of the disk, which can be used to distinguish a specific device on the disk.
- [Max](#): It displays the maximum usage and peak utilization of the disk recorded during the query period.
- [Min](#): It displays the minimum disk usage and minimum utilization recorded during the query period.
- [First / Last](#): You can see the change in usage through the first and last data collected during the query period.
- [Increase](#): It displays the increase/decrease rate of disk usage during the selected period.
- [Increase\(Per Day\)](#): This value is calculated for increase/decrease rate of the average daily usage during the query period, allowing you to check the disk usage change trend. This value is not displayed if the query period is less than one day.
- [PredictedSpace\(N days later\)](#): This value predicts the expected usage after N days based on the average daily increase/decrease rate. This value is not displayed if the query period is less than one day.

For example, if the query period is 7 days, it displays the expected usage after 7 days.

- ! • By clicking [AVG](#) and [MAX](#), you can see the average usage and maximum usage for the query period.



- By dragging  to the left or right, you can adjust the screen size.

WhaTap GPU Monitoring

WhaTap GPU Monitoring provides integrated analysis of GPU resource status and utilization across server and Kubernetes environments. While GPU is one of many resources in a server, it is a critical resource that directly impacts the performance of AI/ML, LLM, and HPC workloads — and comes with significantly higher costs compared to CPU or memory.

Beyond simply checking **whether a GPU is installed and running**, you need visibility into how much it is being utilized, which workloads are occupying it, whether any anomalies exist, and whether resource allocation is appropriate.

WhaTap GPU Monitoring Scope

! WhaTap GPU Monitoring supported environments

- GPU monitoring in Server environments
- GPU monitoring in Kubernetes environments

Server Environment

- GPU server monitoring periodically collects and stores key metrics such as GPU utilization, memory usage, temperature, power, clock speed, errors, and PCIe/NVLink communication status, enabling comprehensive assessment of GPU device health from a server infrastructure perspective.
- Process-level GPU occupancy data is also provided, so you can see which workloads are using which GPUs and how much. This goes beyond simple device health checks to help you analyze how GPU resources are actually being used within the server.
- Monitoring enables early detection of signs such as overheating, throttling, power limits, abnormal errors, and PCIe/NVLink communication issues — helping you prevent failures and respond quickly. This directly contributes to service stability and improved operational efficiency.

Kubernetes Environment

- In Kubernetes environments, GPUs are primarily allocated to AI training/inference workloads, batch jobs, and high-performance computing applications. Viewing only node-level GPU status is often insufficient for understanding real-world operational conditions.
- WhaTap Kubernetes GPU monitoring provides visibility into GPU resource usage at the cluster, node, and pod/container levels. This helps you understand which workloads are occupying GPUs, whether usage is concentrated on specific nodes, and how efficiently GPUs are being used relative to requests and allocations.
- GPU monitoring from a Kubernetes perspective is also useful for analyzing scheduling efficiency, resource imbalance, and over/under-allocation. Ultimately, it improves the stability of GPU-based workload operations and enhances overall cluster resource utilization.

Enterprise Environment

- Enterprise environments typically run many GPU servers and Kubernetes clusters together. Monitoring individual servers or Kubernetes environments in isolation makes it difficult to get a comprehensive view of overall resource status, so an integrated management approach is required.
- Organizations operating hundreds to thousands of GPUs want continuous visibility into how effectively GPU resources allocated by team or workload are being used. This data is essential for making informed decisions about idle resource reallocation, capacity expansion, and procurement planning.
- By unifying multiple Kubernetes-based GPU environments and bare-metal GPU server environments under a single monitoring framework, you can manage distributed GPU resources from a single perspective. This delivers enterprise-grade GPU monitoring that provides a consolidated view of GPU resources across the organization and connects to Capacity Planning.
- To group GPU usage by team or project by linking Kubernetes pod labels to GPU metrics, see [GPU Service Grouping](#).

Value of WhaTap GPU Monitoring

GPU Resource Visibility

Gain a multi-dimensional view of GPU usage down to the server, cluster, node, pod, and process level.

Early Anomaly Detection and Rapid Response

Quickly identify signs such as temperature spikes, power limits, clock degradation, errors, and communication issues before they escalate into failures.

GPU Utilization Optimization

Identify idle GPUs, uneven usage, and underutilized resources to enable reallocation and operational optimization.

Improved Operational Efficiency

Manage server and Kubernetes environments in an integrated manner — rather than separately — to reduce operational complexity.

Informed Expansion and Investment Decisions

Use real usage data to evaluate the need for GPU expansion, and leverage it as supporting evidence for budget and procurement planning.

Capacity Planning Support

Connect short-term incident response to medium- and long-term resource demand forecasting and operational strategy planning.

GPU Device Anomaly Detection

Methods for detecting anomalies in GPU devices and workloads.

1. Xid-Based Anomaly Detection

NVIDIA GPUs record Xid events in logs when a failure or abnormal condition occurs. Detecting the presence of Xid keywords and codes in system logs or driver logs enables relatively fast identification of GPU device anomalies.

For example, specific Xid events can indicate GPU computation errors, memory access issues, driver/hardware faults, or reset events — making Xid events a valuable failure signal in production environments.

2. Status Metric-Based Anomaly Detection

GPU anomalies are not always reflected in error logs alone. In production environments, monitoring the change patterns of key status metrics — such as GPU utilization, memory usage, temperature, power, clock speed, and PCIe/NVLink communication status — enables earlier anomaly detection.

However, it is important to **interpret these metrics not solely by absolute values, but in conjunction with workload characteristics, task type, time-of-day patterns, and typical baselines**. For example, training, inference, batch processing, and data preprocessing workloads each have different normal GPU usage patterns, so the same value can mean different things depending on the operational context.

From this perspective, the following pattern changes are key areas to monitor:

- GPU utilization is excessively high compared to typical workload patterns, or allocated resources remain persistently underutilized
- Memory occupancy remains abnormally high after a task completes, or memory usage spikes sharply from a certain point
- Temperature continuously rises beyond the normal operating range, or a sustained high-temperature condition persists for an extended period
- Power consumption hovers near the limit for a prolonged period, suggesting performance throttling due to power capping
- Clock speed remains lower than expected relative to the load level, suggesting throttling or abnormal control behavior
- PCIe/NVLink communication volume shows abnormal increases or decreases relative to workload characteristics, or error indicators are observed on InfiniBand

Interpreting status metrics as change patterns within operational context — rather than as individual values — allows you to more accurately identify issues such as performance degradation, overheating, throttling, power limits, communication bottlenecks, and abnormal behavior.

3. Process/Workload-Based Anomaly Detection

Viewing workload information such as processes, pods, and containers occupying the GPU alongside device

metrics helps clarify anomalies that would be difficult to identify from device metrics alone. However, this also requires considering workload type, execution stage, repetition cycle, and typical occupancy patterns — not just whether something is occupying the GPU.

Since training, inference, batch processing, and data preprocessing workloads use GPUs differently, the criteria for normal versus abnormal can vary depending on the type of task — even when the occupancy state appears the same.

From this perspective, the following patterns are key areas to monitor:

- A specific process or workload occupies GPU memory in excess of what its task characteristics would suggest, without releasing it within the expected timeframe
- Memory occupancy is sustained for a long time while GPU utilization is low, suggesting task stagnation, abnormal waiting, or potential memory leaks
- A specific task repeatedly restarts or fails while inefficiently occupying GPU resources
- Workloads are persistently concentrated on a subset of GPUs, causing usage imbalance or localized overheating
- Compared to similar workloads, a specific process shows excessively long execution time, high memory occupancy, or abnormal occupancy patterns
- After a pod or container is rescheduled, GPU occupancy patterns change drastically, suggesting scheduling inefficiency or resource imbalance

These patterns can help identify not only device-level issues but also application or workload operational problems.

GPU Dashboard

The **GPU Dashboard** provides a Kubernetes perspective to track the flow from GPU resource allocation to actual utilization and the per-device status on a map.

- Provides a funnel of stage-by-stage resource flow from Total to Effective so you can see actual utilization against allocation at a glance.
- Displays device-level resource distribution and status by cluster, node, and namespace on the GPU Map.
- Provides a toggle that filters waste resources only, so you can quickly identify GPUs that are allocated but not utilized.

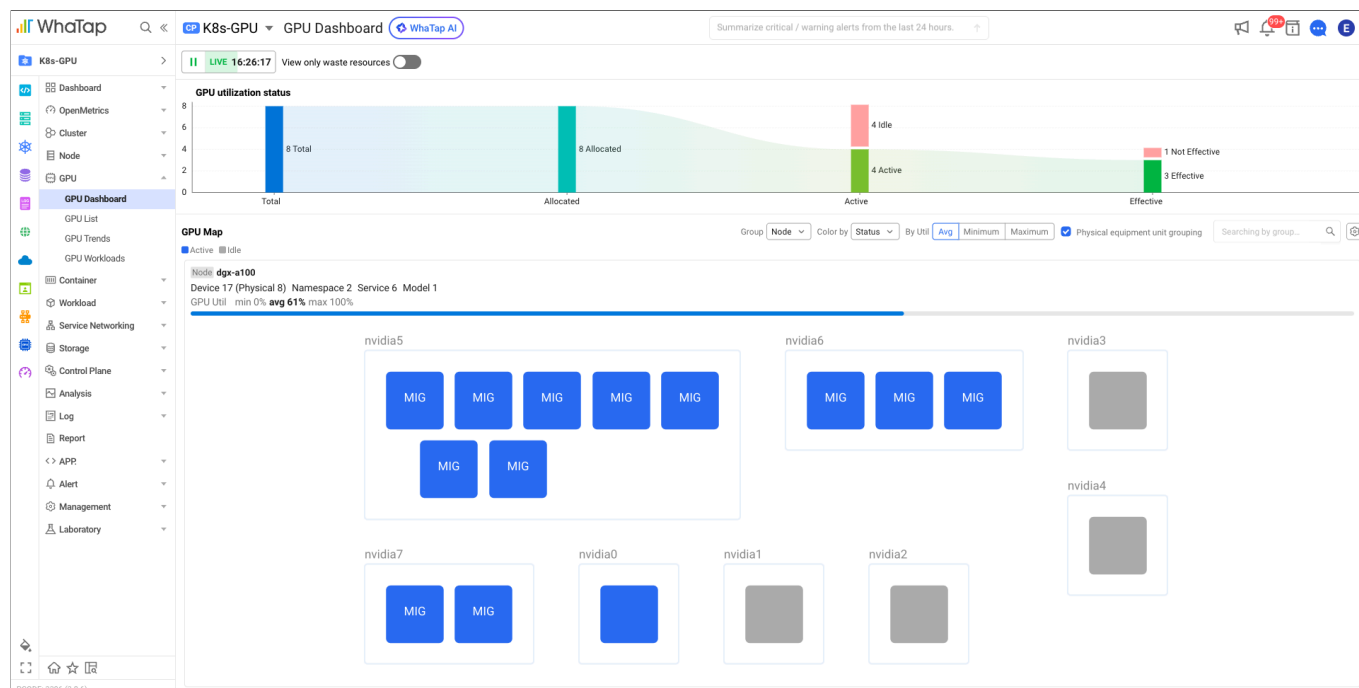
! The GPU menu consists of the **GPU Dashboard**, **GPU List**, **GPU Trend**, and **GPU Workload** pages. This document covers the **GPU Dashboard** page, which provides resource utilization status and the device map.

Permissions & Requirements

- **Supported environment:** Kubernetes cluster project
- **Agent version:** Kubernetes Agent v1.8.7 or higher
- Requires Open Agent installation

Main Screen

A visualization dashboard that lets you identify GPU resource utilization status and per-device status within the cluster at a glance.



At the top of the screen are the LIVE time and the **Show waste resources only** toggle. Turning the toggle on displays only the devices classified as waste resources on the GPU Map.

- The **Show waste resources only** toggle filters the display targets of the GPU Map and does not affect the aggregated values in the **GPU Utilization Status** funnel.
- Selecting the pause button in the time selector at the top of the screen lets you query data from a past time.

GPU Utilization Status

Provides a funnel chart of the flow from GPU resource allocation to actual utilization. The stages narrow in the order of **Total**, **Allocated**, **Active**, and **Effective**, and resources not utilized at each stage are separated and shown as **Idle** and **Not Effective**.

| Stage | Description |
|---------------|---|
| Total | All physical devices |
| Allocated | Devices with a process or Pod allocated |
| Active | Devices with a 5-minute average GPU Utilization of 1% or higher |
| Idle | Devices with a 5-minute average GPU utilization below 1% |
| Effective | Devices with GPU utilization above 50% |
| Not Effective | Devices with GPU utilization of 50% or lower |

The **Allocated** determination criteria changed from whether a process is running to whether a Pod label exists in the GPU metrics. Even when a Pod with a GPU allocated fails to start a process due to a CrashLoop or similar, it is accurately counted as an allocated device. Accordingly, the Allocated description in the table above can be understood as "devices where a Pod with a GPU allocated exists".

In the example above, 8 of 8 Total are Allocated, split into 4 Active and 4 Idle, and the Active are classified into 2 Effective and 2 Not Effective.

Idle and Not Effective are devices that are allocated but not sufficiently utilized, serving as the basis for checking resource waste.

GPU Map

Displays the GPU devices collected at the query time on a map. Each device is placed as a box inside a node card, and the top of the card shows the device, namespace, service, and model counts for that node along with a GPU utilization summary.

Controls

| Control | Option | Description |
|--------------------------|--|--|
| Group | Cluster / Node / Namespace / Service / Model | Criterion for grouping devices |
| Color by | Status / Utilization / VRAM(%) / Temp | Metric for the device box color |
| Util basis | Average / Minimum / Maximum | Statistic used for the utilization display |
| Group by physical device | On / Off | Group and display devices by physical device |
| Group name search | Text input | Filter display targets by group name |

Node Cards and Device Boxes

Each node card displays the devices installed on that node as boxes. The summary at the top of the card shows the following information.

- **Device n (Physical n):** Device count and physical device count
- **Namespace n / Service n / Model n:** Namespace, service, and model counts associated with the node
- **GPU Util (min / avg / max):** Minimum, average, and maximum of the node GPU utilization

Device boxes are displayed differently by status.

| Display | Meaning |
|-----------------|---|
| MIG-labeled box | Device split into MIG instances |
| Colored box | Active device, color determined by the Color by criterion |
| Gray box | Idle device (5-minute average GPU utilization below 1%) |

The legend distinguishes **Active** and **Idle** by color.

Hovering over a device box shows the device type (Physical/MIG), GPU Utilization, Status, Device, Pod, Node, Namespace, and Service information.

- Changing the **Color by** option makes the color of the same device differ depending on the criterion metric (status, utilization, VRAM, temperature).
- A device split by MIG shows one physical device as multiple instances, so the device count and physical device count can differ.

GPU Trends

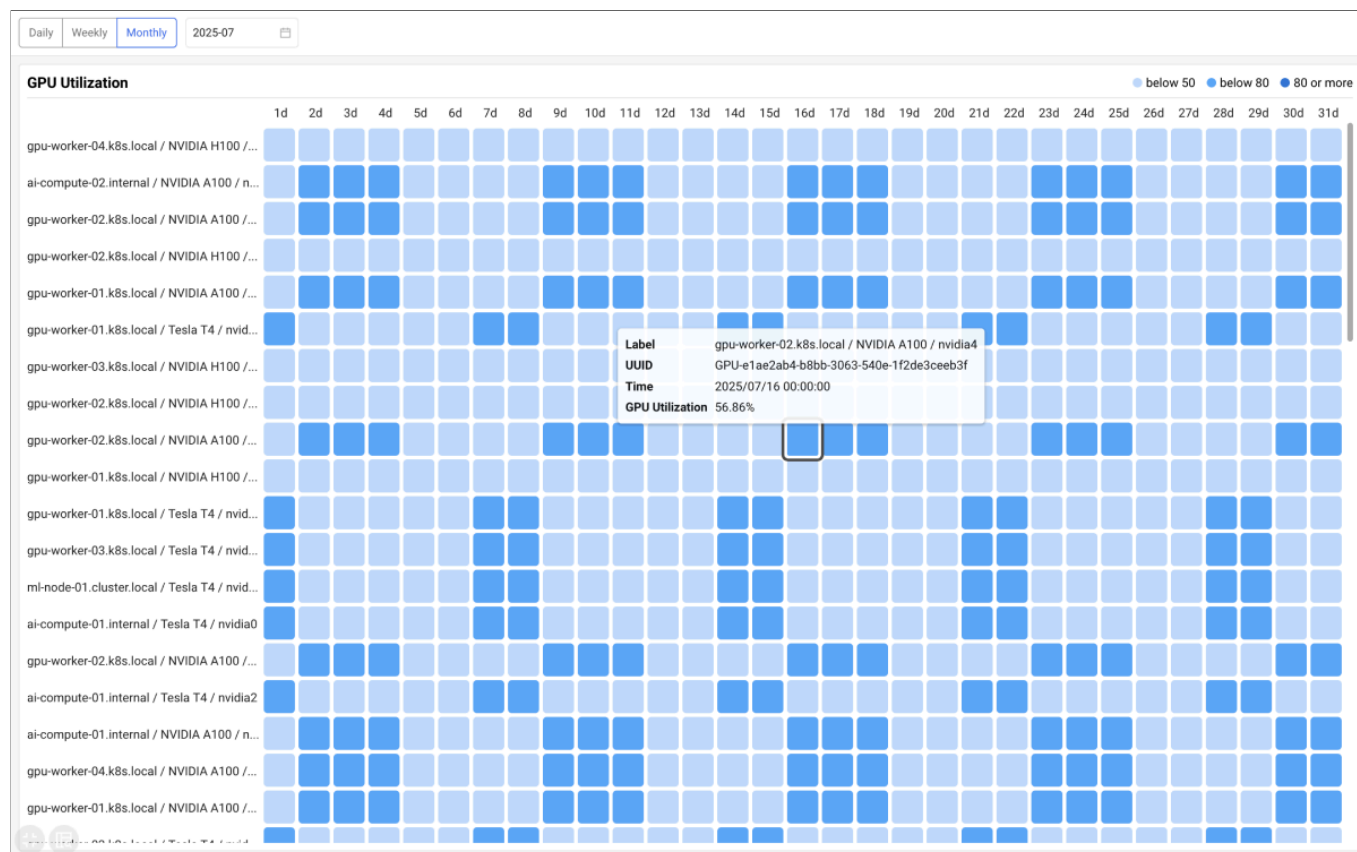
GPU Trends visualizes long-term usage statistics per GPU device, enabling quick identification of devices that were inefficiently used at specific times.

Main Screen

You can check long-term GPU usage statistics and patterns. The query unit supports **daily, weekly, and monthly** views.

Each device's GPU utilization is displayed as blocks, with color intensity representing the level of usage.

- **Less than 50%:** Light Blue
- **Less than 80%:** Blue
- **80% or higher:** Dark Blue



View Detailed Information

Hover over each block to see detailed information for the corresponding device and time in a tooltip.

- **Label:** Device name and GPU model
- **UUID:** Unique GPU identifier
- **Time:** Reference time
- **GPU Utilization:** GPU utilization (%) at that time

GPU Service Grouping

! For an overview of GPU monitoring, see the [WhaTap GPU Monitoring](#) document.

WhaTap Operator's GPU monitoring feature **attaches Kubernetes labels from application (GPU-using) pods to GPU metrics**, enabling grouping and querying of GPU usage by service or project in WhaTap.

- Target: `WhatapAgent` CR (`monitoring.whatap.com/v2alpha1`)
- Related fields: `spec.features.k8sAgent.gpuMonitoring.groupLabel` , `groupLabelAllowlistRegex`
- Key result label: `whatap_kube_label_gpu_group`

✓ What problem does this solve?

Default DCGM metrics (e.g., `DCGM_FI_DEV_GPU_UTIL`) include GPU/node/pod identifiers, but do not contain **"service"-level information** that users operate (e.g., project ID, team, workload type).

As a result:

- Queries like "total GPU utilization used by Project A" are difficult.
- Cost and usage accounting in multi-tenant environments (multiple teams sharing a cluster) is difficult.

With this feature, labels attached to pods (e.g., `prjId=projectA`) are **automatically propagated to DCGM metric labels** and normalized to WhaTap's common label `whatap_kube_label_gpu_group` , making them immediately available as grouping keys in dashboards and searches.

Quick Start

Enable GPU Monitoring + Group Label

```
apiVersion: monitoring.whatap.com/v2alpha1
kind: WhatapAgent
```

```

metadata:
  name: whatap
  namespace: whatap-monitoring
spec:
  features:
    k8sAgent:
      gpuMonitoring:
        enabled: true
        # Label key to use as grouping key
        groupLabel: "prjId"
        # (Optional) Regex for exposed label keys. If not set, ^(prjId)$ is applied automatically
        # groupLabelAllowlistRegex: "^(prjId|team)$"
        # (Optional) Attach cluster label to all GPU metrics
        clusterName: "prod-gpu-cluster"

```

Add Labels to GPU-Using Pods

```

apiVersion: apps/v1
kind: Deployment
metadata:
  name: ml-train
  namespace: ml
spec:
  replicas: 2
  selector:
    matchLabels:
      app: ml-train
  template:
    metadata:
      labels:
        app: ml-train
        prjId: "projectA" # ◀ Grouping key (same as groupLabel)
    spec:
      containers:
        - name: trainer
          image: my-registry/trainer:latest
          resources:
            limits:
              nvidia.com/gpu: 1

```

❗ Labels must be applied at the **Pod level** (`spec.template.metadata.labels`), not at the Deployment level.

Use Cases

Usage by Project (prjId)

- `groupLabel: prjId`
- Pod labels: `prjId: projectA` , `prjId: projectB`
- Group by `whatap_kube_label_gpu_group` in WhaTap dashboard → compare GPU utilization and memory usage by project

Billing by Team

```
gpuMonitoring:
  enabled: true
  groupLabel: "team"
  groupLabelAllowlistRegex: "^(team)$"
```

- Pod labels: `team: search` , `team: recsys` , ...

Exposing Multiple Labels Together

While only one label is normalized via `groupLabel` (`whatap_kube_label_gpu_group`), you can use `groupLabelAllowlistRegex` to also expose additional pod labels as metric labels.

```
gpuMonitoring:
  enabled: true
  groupLabel: "prjId"
  groupLabelAllowlistRegex: "^(prjId|team|env)$"
```

- `whatap_kube_label_gpu_group` ← value of `prjId`
- `team` , `env` labels are also exposed as-is in DCGM metrics (not normalized)

Verification

Check dcgm-exporter Environment Variables

```
kubectl -n whatap-monitoring get ds whatap-node-agent \
-o jsonpath='{.spec.template.spec.containers[?(@.name=="dcgm-exporter")].env}' | jq
```

The output must contain the following two environment variables for pod label collection to be active.

- `DCGM_EXPORTER_KUBERNETES_ENABLE_POD_LABELS=true`
- `DCGM_EXPORTER_KUBERNETES_POD_LABEL_ALLOWLIST_REGEX=^(prjId)$`

Query Metrics Directly (Pod must have labels)

```
# Connect to one dcgm-exporter Pod
POD=$(kubectl -n whatap-monitoring get pod -l name=whatap-node-agent -o name | head -1)
kubectl -n whatap-monitoring exec "$POD" -c dcgm-exporter -- \
  wget -qO- localhost:9400/metrics | grep DCGM_FI_DEV_GPU_UTIL | head
```

```
DCGM_FI_DEV_GPU_UTIL{gpu="0", ..., pod="ml-train-xxxx", namespace="ml", prjId="projectA"} 87
```

Check OpenAgent Scrape Configuration

```
kubectl -n whatap-monitoring get configmap whatap-open-agent-config -o yaml | grep -A2 whatap_kube_label_gpu_group
```

The output must contain the following relabel block. The value of `source_labels` must match the configured `groupLabel` key.

```
- action: replace
  source_labels: [prjId]
  target_label: whatap_kube_label_gpu_group
```

Troubleshooting

| Symptom | Cause / Checklist |
|---|---|
| <code>prjId</code> label not visible in metrics | (1) Check if <code>prjId</code> is set in pod labels (<code>spec.template.metadata.labels</code>), (2) Check if <code>dcgm-exporter</code> has been restarted to apply new env vars |
| <code>prjId</code> is visible but <code>whatap_kube_label_gpu_group</code> is not | OpenAgent pod must be restarted after ConfigMap update. <code>kubectl rollout restart deploy/whatap-open-agent -n whatap-monitoring</code> |
| <code>groupLabel</code> value contains <code>-</code> , <code>.</code> , <code>/</code> | Per Prometheus label key rules, <code>source_labels</code> is composed with <code>_</code> -replaced keys. The allowlist regex uses the original key, so set it as-is |
| Want to expose multiple labels | Specify <code>groupLabelAllowlistRegex</code> as <code>^(prjId team env)\$</code> |
| Need cluster identification on all GPU metrics | Set the <code>clusterName</code> field (attaches <code>cluster=<value></code> to all metrics) |

Field Reference

| Field | Description | Default |
|---|---|--------------------------------------|
| <code>gpuMonitoring.enabled</code> | Enable GPU monitoring (dcgm-exporter) | <code>false</code> |
| <code>gpuMonitoring.groupLabel</code> | Pod label key used for grouping. When set, normalized to <code>whatap_kube_label_gpu_group</code> | - |
| <code>gpuMonitoring.groupLabelAllowlistRegex</code> | Regex for pod label keys that dcgm-exporter exposes | <code>^(<groupLabel>)\$</code> |
| <code>gpuMonitoring.clusterName</code> | Attach <code>cluster</code> label to all GPU metrics | - |

Reference

- git repo: <https://github.com/whatap/whatap-operator.git>
- Implementation: `internal/controller/install_agents.go` (`addDcgmExporterToNodeAgent` , GPU scrape config generation)
- CRD definition: `api/v2alpha1/whatapagent_types.go` (`GpuMonitoringSpec`)
- Examples: `examples/whatap-agent-all-options.yaml` , `examples/whatap-agent-gpu-custom-env.yaml`

Container List

Home > Select Project > [Container](#) > [Container List](#)

ⓘ If the data for specific columns does not appear, your Kubernetes node agent may be outdated. In this case, update the Kubernetes agent so that enough data can be collected.

Through the [Container List](#) menu, you can see the container details of the current project in the list format.

- Real-time data is updated every 10 seconds.
- It provides the functions for container status summary, comparisons, filters, and [CSV](#) file download.

CSV file name, [kubernetes_contianer_list_](#) {pcode} {timestamp} .csv_

Basic screen guide

1 EKS-DEMO ▾ Container List

🔊 🔔 📅 ⋮ 👤

1 **LIVE** 12:35:21 All 82 **RUNNING** 81 **WAITING** 1

🔍 + There is no filter applied. Select Column CSV

2

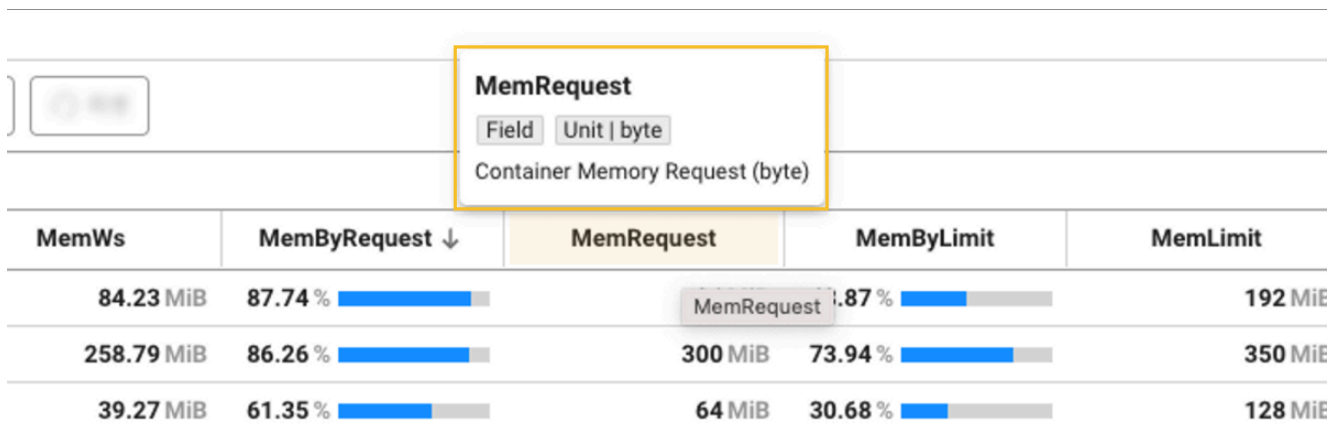
| Co... | | Container Info | | | | | | | CpuTotUsage | CpuRequest |
|--------------------------|-----------------------|-------------------------|----------------------|-------------------|--------------------|----------------|--|---------|-------------|------------|
| Co... | ContainerId | ContainerName | Image | ApplicationName | Node | Created | | | | |
| ☐ | 45f4ebc713c801143... | argama | 224983703333... | argama-110-215 | ip-10-21-110-99... | 10/29 18:36:13 | | 0.98 m | 1 | |
| ☐ | 845840d3f26a0eae1... | argama | 224983703333... | argama-140-225 | ip-10-21-140-92... | 10/17 18:23:50 | | 3 m | 1 | |
| ☐ | 5f236aef9674e5b309... | aws-load-balancer-co... | public.ecr.aws/e... | - | ip-10-21-110-99... | 10/29 13:50:51 | | 3.04 m | | |
| ☐ | 806f4d9ce7de24f88d... | aws-load-balancer-co... | public.ecr.aws/e... | - | ip-10-21-140-92... | 10/29 14:53:01 | | 1.03 m | | |
| ☐ | 34ef8df12976bfb59d... | aws-node | 602401143452... | - | ip-10-21-140-92... | 10/17 09:31:57 | | 2.09 m | | |
| ☐ | 97dfdc7a1f6cfc55... | aws-node | 602401143452... | - | ip-10-21-140-61... | 10/30 10:52:32 | | 1.95 m | | |
| ☐ | f75209e2574de268e... | aws-node | 602401143452... | - | ip-10-21-110-99... | 10/29 09:20:13 | | 4.01 m | | |
| ☐ | b6ff8a71ca3cd389f7... | cache-server | docker.io/library... | - | ip-10-21-110-99... | 10/30 10:48:24 | | 2.87 m | 2 | |
| ☐ | 71ef245f37b730453... | catserver | 224983703333... | catserver-140-224 | ip-10-21-140-61... | 10/30 10:48:25 | | 80.92 m | 2 | |
| ☐ | 5c29fe1428d86ac5fe... | coredns | 602401143452... | - | ip-10-21-110-99... | 10/29 13:51:02 | | 6.01 m | 1 | |
| ☐ | 62fa8dc7a8e6056fba... | coredns | 602401143452... | - | ip-10-21-110-99... | 10/29 13:50:51 | | 0.97 m | 1 | |
| ☐ | 15976914d7af878d6... | csi-attacher | 602401143452... | - | ip-10-21-110-99... | 10/29 13:50:51 | | 0 m | | |
| ☐ | 37540c253d5db2e37... | csi-attacher | 602401143452... | - | ip-10-21-110-99... | 10/29 13:51:12 | | 0 m | | |
| ☐ | 1aafb9bf56fbc930e... | csi-provisioner | 602401143452... | - | ip-10-21-110-99... | 10/29 13:51:12 | | 0 m | | |
| ☐ | ad32aedd84fbc686... | csi-provisioner | 602401143452... | - | ip-10-21-110-99... | 10/29 13:50:51 | | 1 m | | |
| ☐ | 3bf840d390132d450... | csi-resizer | 602401143452... | - | ip-10-21-110-99... | 10/29 13:50:51 | | 0 m | | |

By selecting desired options in the **1** area, you can view the data in **2**.

- On the upper left of the **1** area, you can see the total container count and the container counts for each status.
- When selecting a desired column in the **2** area, you can sort the containers based on the corresponding column.
- When selecting the  **Node Info** icon under the **Node** column in the **2** area, the **Node list** menu appears.

Metric information

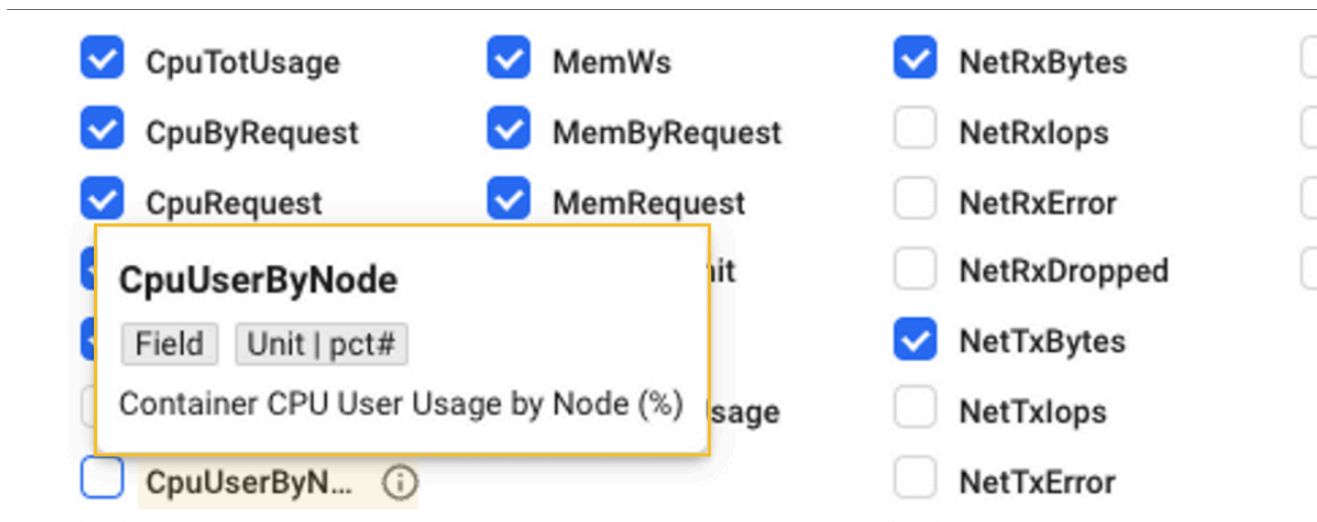
- If you place your mouse cursor over the desired column, the metrics provided by the column are shown as follows:



The screenshot shows a table with columns: MemWs, MemByRequest ↓, MemRequest, MemByLimit, and MemLimit. A tooltip for the MemRequest column is displayed, showing the field name, unit (byte), and a description: 'Container Memory Request (byte)'. The table data is as follows:

| 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 |

- In the column list of the [Select Column](#) option, you can see the metrics for each column in the column list of the option screen. When you place the mouse cursor over a desired column, the [INFO icon](#) appears to the right of the column name. If the mouse cursor is placed over the [INFO icon](#), you can see the metric as follows:



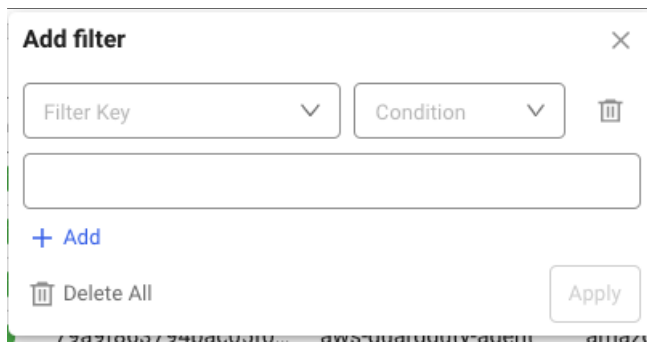
The screenshot shows the 'Select Column' dialog with a list of metrics. A tooltip for 'CpuUserByNode' is displayed, showing the field name, unit (pct#), and a description: 'Container CPU User Usage by Node (%)'. The metrics listed are:

| Column | Selected |
|---------------|-------------------------------------|
| CpuTotUsage | ☒ |
| CpuByRequest | ☒ |
| CpuRequest | ☒ |
| CpuUserByNode | ☐ |
| MemWs | ☒ |
| MemByRequest | ☒ |
| MemRequest | ☒ |
| NetRxBytes | ☒ |
| NetRxlops | ☐ |
| NetRxError | ☐ |
| NetRxDropped | ☐ |
| NetTxBytes | ☒ |
| NetTxlops | ☐ |
| NetTxError | ☐ |

Filtering

By using the [Filter](#) option, you can select a desired Container list for view. Through the [and](#) condition, multiple filters can be applied.

- If you select + [Add icon](#) in the [Filter](#) field, the [Add filter](#) window appears as follows:

A dialog box titled "Add filter" with a close button (X) in the top right corner. It contains two dropdown menus labeled "Filter Key" and "Condition", each with a downward arrow. To the right of the "Condition" dropdown is a trash icon. Below these is a large empty text input field. At the bottom left is a blue "+ Add" button. At the bottom right is a "Delete All" button with a trash icon and an "Apply" button.

- Select [Filter Key](#) and [Condition](#).
 - Select a value that meets the conditions.
2. Configure a filter and then select [Apply](#) to apply the filter.
 3. To add multiple filters, select + [Add icon](#) in the [Filter](#) field or [Add filter](#) window.

Comparing

The [Compare](#) option allows you to see a comparison chart of key metrics for multiple containers. When you select the [Compare](#) check boxes of the containers to view in the list, a window comparing details for the containers appears on the right as shown below.



When using the [Select column](#) option, you can select the desired Container data to view the list.

Select Column
Download

Deselect all(19)
Initialize(default)
×

| | | | | | |
|---|--|--|--|--|---------------------------------------|
| ☐ Container Info | ☐ Agent Info | ☐ CPU | ☐ Memory | ☐ Network | ☐ Block IO |
| ☐ ContainerKey | ☐ Command | ☒ CpuTotUsage | ☒ MemWs | ☒ NetRxBytes | ☐ IoReadBytes |
| ☒ ContainerName | ☐ Application | ☒ CpuByRequest | ☒ MemByRequest | ☐ NetRxlops | ☐ IoReadlops |
| ☒ Namespace | ☒ MicroOname | ☒ CpuRequest | ☒ MemRequest | ☐ NetRxError | ☐ IoWriteBytes |
| ☒ Deployment | ☐ Okind | ☒ CpuByLimit | ☒ MemByLimit | ☐ NetRxDropped | ☐ IoWritelops |
| ☒ ReplicaSet | ☐ OkindName | ☒ CpuLimit | ☒ MemLimit | ☒ NetTxBytes | |
| ☒ Pod | ☐ NodeID | ☐ CpuByNode | ☐ MemMaxUsage | ☐ NetTxlops | |
| ☐ Image | ☒ Node | ☐ CpuUserByNode | | ☐ NetTxError | |
| ☐ ImageId | ☐ Created | ☐ CpuSysByNode | | ☐ NetTxDropped | |
| ☐ ConStatus | | ☐ CpuThrottledCnt | | | |
| | | ☐ CpuThrottledTime | | | |

Apply

- The selected column list is kept even if you re-enter the menu.
- If you select [Reset](#), the predefined default key metric columns are selected. In this case, the column selection history is deleted.
- After selecting a column, you can click [Apply](#) at the lower right to apply the selected values.

Container Timeline

Home > Select Project > **Container** > **Container Timeline**

ⓘ Requirements

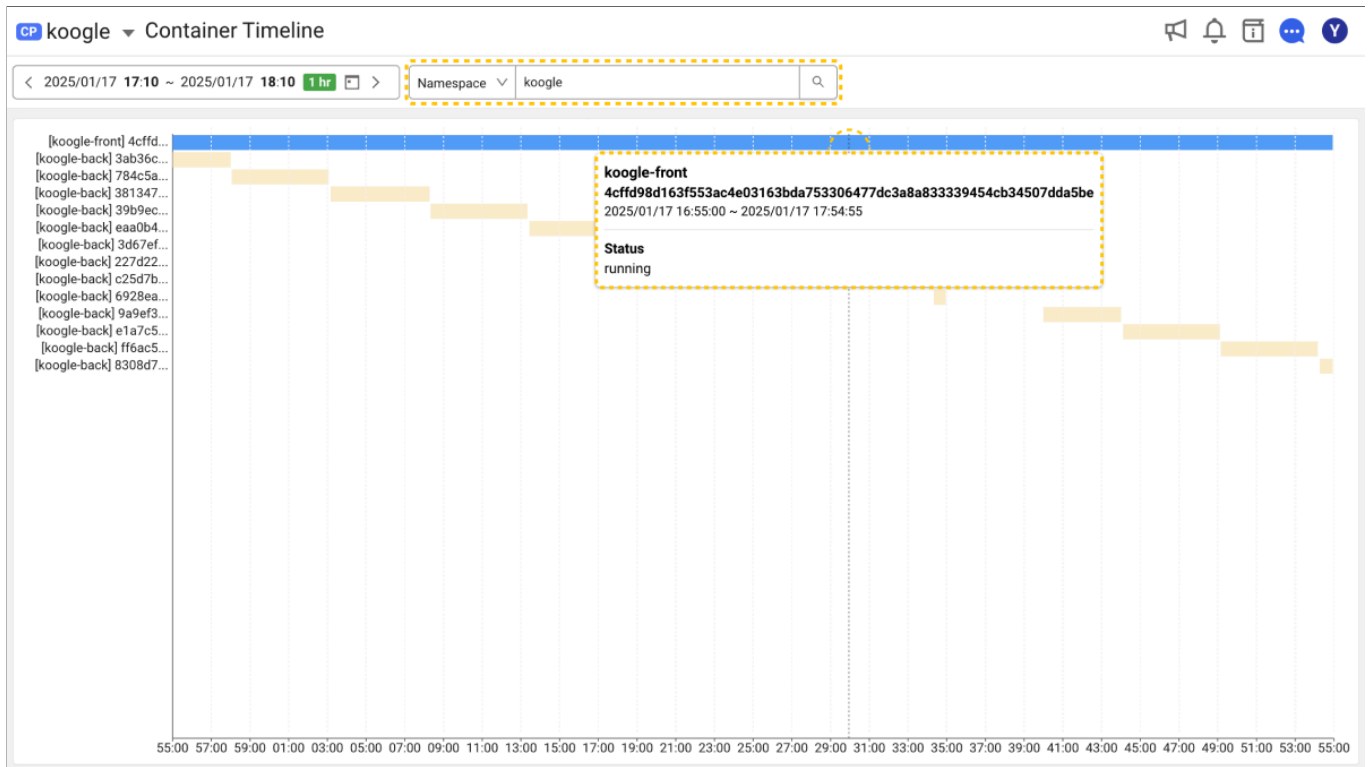
The WhaTap Kubernetes agent 1.7.6 or later is required. Timeline data is provided from the time point collected in version 1.7.6 or later.

The **Container Timeline** menu provides the feature to visually trace state changes of Container over time. You can visually see when the terminated Container was terminated, and which state Container was in at a specific time point. This allows you to analyze the past status, identify the causes of issues in the cluster, and monitor the operational stability.

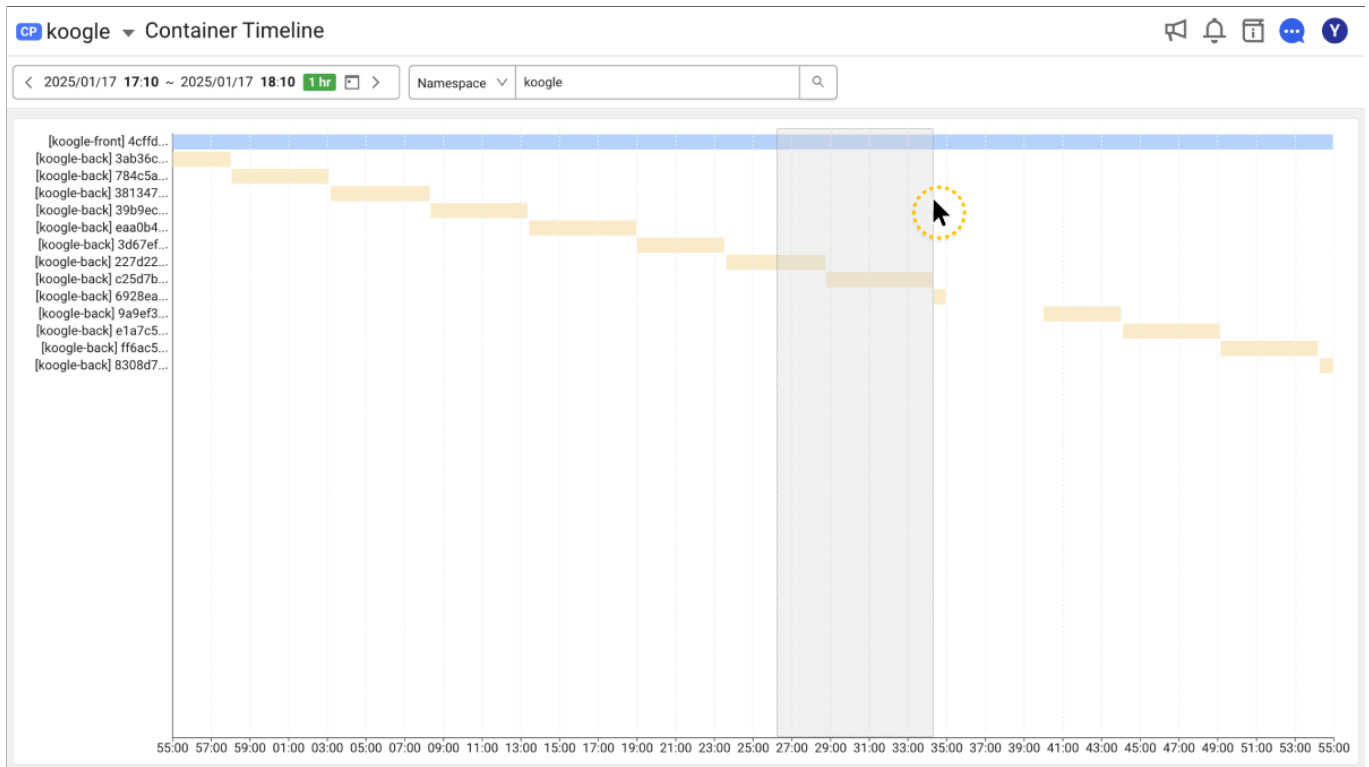
Basic screen guide

After entering the **Container Timeline** menu, select **Filter key** and **Filter value** to view the data. You can see the Container list collected in the last 1 hour. You can visually see the state changes of Container along the time axis.

If you hover your mouse over each time zone, the start and end times of it and container status ([running](#), [waiting](#), [terminated](#)) appear.



You can adjust the time range by dragging the zone on the timeline. Set a desired period to check the past data.

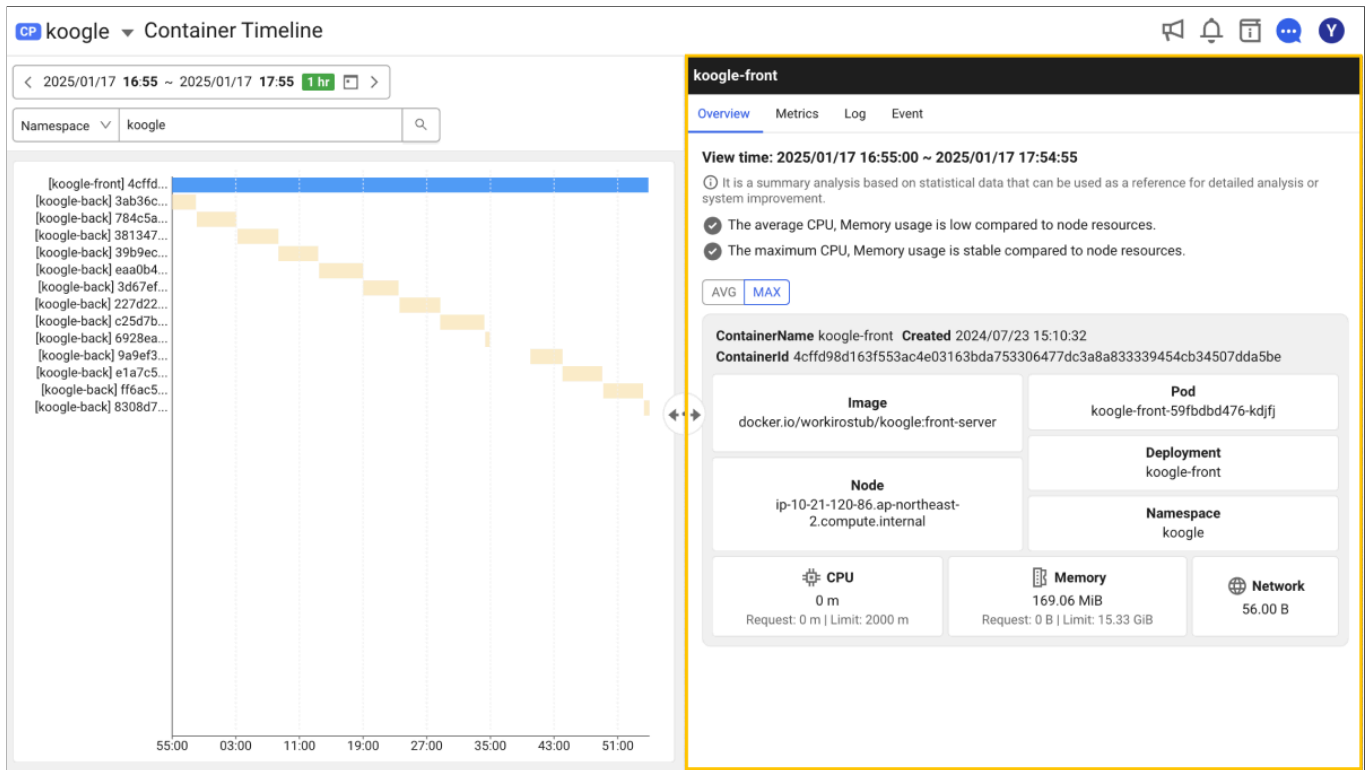


Additionally, when you select a specific period, you can check the Container status during the period in detail through **Display Detail**. For more information, see the [Container timeline detail](#) guidance.

! For more information on how to use the time selector.

Container timeline details

You can check the Container timeline details by selecting a specific segment or Container.



- You can view detailed information of a Container at the time point through the **Display Detail** window on the right by selecting a zone to view.
- You can select a Container name to view from the list on the left of the chart and view the entire status of the corresponding Container through the **Display Detail** window on the right.

! The **Metrics**, **Event**, **Log** tabs of the detailed view window are the same as in Container map's Display Detail.

Overview

The **Overview** tab provides a summary of the selected container and overall resource usage. You can comprehensively see the container information including the container image, the node on which the container is running, and the `ownerReference` of the Pod on which the container is running. You can see the resource usage and status of the container and monitor the performance and operational stability. You can also view the average usage and maximum usage for the selected period by clicking **AVG** and **MAX**.

☒ ip-10-21-11-187.ap-northeast-2.compute.internal
☐ ip-10-21-11-190.ap-northeast-2.compute.internal
☐ ip-10-21-13-161.ap-northeast-2.compute.internal
☐ ip-10-21-13-168.ap-northeast-2.compute.internal
☐ ip-10-21-13-234.ap-northeast-2.compute.internal
☐ ip-10-21-13-55.ap-northeast-2.compute.internal

Container volume information

The items provided in [Container Volume](#) are as follows:

| Item | Description |
|-----------------------------|---|
| ContainerId | Container ID, a unique identifier for the container. This ID is generated by the container runtime. It is required to identify and manage containers within a cluster. |
| Pod Name | Name of the Pod to which the container belongs. In a Kubernetes cluster, multiple containers can run within a Pod. Using the name, it is possible to identify which Pod the container belongs to. |
| MountType | <p>Volume-mounted type. However, because <code>tmpfs</code> is a temporary file system that uses memory, it is not displayed in the corresponding Container Volume list.</p> <ul style="list-style-type: none"> - <code>bind</code> : It means that a specific directory is mounted on the host. - <code>volume</code> : It means that a specific file system (e.g. <code>/var/lib/docker/volumes/</code>) managed by Docker is mounted on the host. |
| Source | Path to the volume on the host. This path indicates where the volume is physically stored. |
| Driver | This volume driver is used when creating the volume. It determines how to store the data and |

| Item | Description |
|-------------|---|
| | manage the volume. Currently, this information is primarily provided by the Docker in runtime. Lower-level container runtimes such as containerd and CRI-O do not directly provide these high-level information details. The monitoring dashboard provides a limited amount of volume data through the runtime API. |
| Mount | The path where the volume has been mounted within the container. This path allows the applications within the container to access the data stored on the volume. |
| Total(Byte) | The total capacity of the volume in bytes. This indicates the maximum amount of data that can be stored on the volume. |
| Used(Byte) | The amount of capacity in bytes in use on the volume. This indicates the amount of data stored on the volume. |
| UsedPercent | The utilization of volume capacity in percentage. This ratio shows the percentage of capacity in use compared to the volume's total capacity. Higher utilization means the volume is running out of space. This can be an important metric for poor performance or risk of data loss. |

Container Images

Home > Select Project > **Container** > **Container Images**

The **Container Images** menu provides available images in the containers, number of containers running with those images, and the total usage amounts for CPU and memory in the list format.

Kubernetes Monitoring Demo(EKS) 컨테이너 이미지

LIVE 11:04:55 전체 21 + 적용된 필터가 없습니다.

다운로드

a 필터 추가하기

필터 키

Registry

Name

Tag

적용

| Registry | Name | Tag | Repo Digest | Containers | Total Container CPU | Total Container Memory |
|--------------------------------|------------------------|------------------------|------------------------------|------------|---------------------|------------------------|
| 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 |
| docker.io | whatap/virtual | latest | sha256:16ebda509ca6f641... | 6 | 8 m | 58.10 MIB |
| 592247757306.dkr.ecr.ap-nor... | apm-php-demo-mariadb | main | sha256:c2ccb4b5e1f29bd91... | 6 | 0 m | 103.71 MIB |
| 592247757306.dkr.ecr.ap-nor... | apm-php-demo-wordpr... | main | sha256:9d7ac873c0a88dc77... | 14 | 324 m | 1.96 GiB |
| 602401143452.dkr.ecr.ap-nor... | eks/coredns | v1.10.1-eksbuild.1 | sha256:3dba81d0c3c8ce51a... | 8 | 0 m | 100.10 MIB |
| 592247757306.dkr.ecr.ap-nor... | apm-php-demo-postgres | main | sha256:ad924171bfad55464... | 2 | 0 m | 19.23 MIB |
| 592247757306.dkr.ecr.ap-nor... | apm-java-demo | main | sha256:867f945ad962ef458f... | 2 | 2 m | 25.41 MIB |
| 592247757306.dkr.ecr.ap-nor... | nodejs_sample_v2 | main | sha256:3207206f677adff845... | 2 | 2 m | 19.77 MIB |
| 592247757306.dkr.ecr.ap-nor... | apm-php-demo-apache | main | sha256:de9c28e2d8eb849b1... | 2 | 0 m | 19.29 MIB |
| 592247757306.dkr.ecr.ap-nor... | php-demo-runtime | php7.3-apache-wp-5.8.3 | sha256:54cd72a83ee3dba65... | 10 | 681 m | 1.75 GiB |
| 592247757306.dkr.ecr.ap-nor... | apm-php-demo-mariadb | main | sha256:712ebae8166a09d4b... | 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 | sha256:361ffffcb30f7db0dd... | 1 | 0 m | 68.79 MIB |

c

a Filter

If **Filter key** is selected in the Filter window, the filtering for **Registry**, **Name**, **Tag**, and **Repo Digest** is provided.

b Download CSV

By selecting **Download CSV** in the upper right corner, you can download the original data as a file in **CSV** format. The file name is generated by adding the project code and timestamp at the time of data inquiry after

```
kubernetes_container_images_.
```

e.g. `kubernetes_container_images_{pcode}_{timestamp}.csv`

Container details view

If you select the  **View containers in detail** icon on the right of the [Name](#) column, the **Container list** menu appears where the image has been searched.

Out Of Memory Killed Container

Home > Project Select > Container > Out Of Memory Killed Container

Through the **Out Of Memory Killed Container** menu, you can quickly search for containers where **Out Of Memory (OOM)** occurred and check the change trend in memory usage. This change trend may be sudden or gradual. By checking this, you can objectively determine the appropriate number of PODs and the memory threshold.

! Example of OOM occurrence

Out Of Memory occurs when the container's memory usage exceeds its limit, or the node on which the container is running encounters an out-of-memory condition. If the container exits with the code 137 , it indicates that SIGKILL has been received by Out Of Memory.

Basic screen guide

CP preview-qa-cluster ▾ Out Of Memory Killed Container

< 2025/03/25 04:54 ~ 2025/03/25 07:54 3 hr >

Container Name

Q

| OomKilledAt | ContainerNa... | ContainerId | Pod | Application... | Created | Command | Image | Namespace | Node | MemLimit | MemWs |
|----------------|----------------|-------------|--------------|----------------|------------------|------------------|-----------------|---------------|-------------|--------------|--|
| 2025/03/25 ... | whatap-mas... | 7dcdf7... | > whatap-... | > - | 2025/02/26 ... | /bin/entryp... | docker.io/wh... | whatap-mon... | ip-10-21... | > 350.00 MiB | <div>286 MiB</div> <div>0 B</div> <div>05:00 07:00</div> |
| 2025/03/25 ... | oom | 46c4ef... | > oom-na... | > oom-na... | > 2025/02/24 ... | java -javaage... | docker.io/wo... | default | ip-10-21... | > 450.00 MiB | <div>525 MiB</div> <div>0 B</div> <div>05:00 07:00</div> |
| 2025/03/25 ... | oom | 12b6ef... | > oom-na... | > oom-na... | > 2025/02/24 ... | java -javaage... | docker.io/wo... | default | ip-10-21... | > 450.00 MiB | <div>525 MiB</div> <div>0 B</div> <div>05:00 07:00</div> |
| 2025/03/25 ... | oom | d8cae8... | > oom-na... | > oom-na... | > 2025/02/24 ... | java -javaage... | docker.io/wo... | default | ip-10-21... | > 450.00 MiB | <div>525 MiB</div> <div>0 B</div> <div>05:00 07:00</div> |
| 2025/03/25 ... | whatap-mas... | e181c7... | > whatap-... | > - | 2025/02/26 ... | /bin/entryp... | docker.io/wh... | whatap-mon... | ip-10-21... | > 350.00 MiB | <div>286 MiB</div> <div>0 B</div> <div>05:00 07:00</div> |
| 2025/03/25 ... | oom | cdc9b7... | > oom-na... | > oom-na... | > 2025/02/24 ... | java -javaage... | docker.io/wo... | default | ip-10-21... | > 450.00 MiB | <div>525 MiB</div> <div>0 B</div> <div>05:00 07:00</div> |
| 2025/03/25 ... | oom | 3d69d9... | > oom-na... | > oom-na... | > 2025/02/24 ... | java -javaage... | docker.io/wo... | default | ip-10-21... | > 450.00 MiB | <div>525 MiB</div> <div>0 B</div> <div>05:00 07:00</div> |
| 2025/03/25 ... | oom | 56176d... | > oom-na... | > oom-na... | > 2025/02/24 ... | java -javaage... | docker.io/wo... | default | ip-10-21... | > 450.00 MiB | <div>525 MiB</div> <div>0 B</div> <div>05:00 07:00</div> |

- You can select a time zone to view by clicking the time selector at the upper left. You can search for up to 2 days.
- You can quickly find information about a specific container to search by entering its name in the input field at the upper right.
- You can check the memory usage trend of the selected container by clicking the [Details icon](#) on the right of the list.
- It provides the OOM containers in the list format. For more information, see the following guide.

OOM killed container information

The columns provided in the **Out Of Memory Killed Container** menu are as follows:

| Columns | Meaning |
|---------------------------------|--|
| OomKilledAt | Time stamp where Out Of Memory Killed occurred |
| ContainerName | Container name |
| ContainerId | Container ID |
| Pod | Name of the Pod to which the container belongs |
| ApplicationName | Name of the application running in the container. |
| Created | Time stamp generated by the container |
| Command | Container's execution command |
| Image | Container's image name |
| Namespace | Name of the namespace to which the container's Pod belongs |
| Node | Node name on which the container is running |
| MemLimit | Container's memory limit size |

| Columns | Meaning |
|---------|--|
| MemWs | Actual memory in use, excluding the memory disabled in the container |

- If you click [containerId](#), the [Container list](#) menu appears.
- If you click [Pod](#), the [Pod list](#) menu appears.
- If you click [_ApplicationName_](#), the [Container application list](#) menu appears where the application is found.
- If you click [Node](#), he [Node list](#) menu appears.

Pod List

Home > Select Project > **Workload** > **Pod List**

It displays the Pod list. If you select a desired column, you can search the Pod list sorted by that column.

CP

koogle-dev-project ▾

Pod List

||

LIVE

15:23:42

All

30

Running 30

▽

+ There is no filter applied.

Select Column

Download

| PodP... | Pod | Pod info | | | | | | | |
|---------|--------------------------|---------------|-------------|-------------------------|---------------|-----------------|------------|----------------|------|
| | | Podip | Namespace | Node | ConRestartCnt | Age | CpuByLimit | CpuLimitByNode | |
| ✓ | amex-6ccf9d89f5-gb... | 10.21.120.39 | titans | ip-10-21-120-186.ap-... | 0 | 13d 00h 29m 04s | 2.16 % | 4.50 % | 0.10 |
| ✓ | api-call-669cbcf45f-k... | 10.21.120.19 | sample | ip-10-21-120-186.ap-... | 0 | 1d 14h 58m 58s | 0.10 % | 100 % | 0.10 |
| ✓ | aws-node-qhv52 | 10.21.120.55 | kube-system | ip-10-21-120-55.ap-n... | 0 | 6h 13m 38s | 0.05 % | 100 % | 0.05 |
| ✓ | aws-node-xljlk | 10.21.120.186 | kube-system | ip-10-21-120-186.ap-... | 0 | 80d 03h 51m 26s | 0 % | 100 % | 0 % |
| ✓ | coredns-76b4dcc5cc-... | 10.21.120.223 | kube-system | ip-10-21-120-186.ap-... | 0 | 80d 03h 58m 04s | 0.10 % | 100 % | 0.10 |
| ✓ | coredns-76b4dcc5cc-... | 10.21.120.203 | kube-system | ip-10-21-120-186.ap-... | 0 | 80d 03h 58m 04s | 0 % | 100 % | 0 % |
| ✓ | customer-7bbd6f949... | 10.21.120.241 | titans | ip-10-21-120-55.ap-n... | 64 | 6h 12m 47s | 0 % | 0 % | 0 % |
| ✓ | datadog-koogle-8827k | 10.21.120.21 | default | ip-10-21-120-186.ap-... | 0 | 20d 22h 54m 24s | 1.00 % | 400 % | 3.99 |
| ✓ | datadog-koogle-clust... | 10.21.120.231 | default | ip-10-21-120-186.ap-... | 0 | 20d 22h 54m 09s | 0.10 % | 100 % | 0.10 |
| ✓ | datadog-koogle-mgm... | 10.21.120.209 | default | ip-10-21-120-55.ap-n... | 0 | 6h 13m 14s | 0.61 % | 400 % | 2.46 |
| ✓ | db-transaction-6f7d9... | 10.21.120.72 | sample | ip-10-21-120-186.ap-... | 0 | 80d 01h 25m 39s | 0 % | 100 % | 0 % |
| ✓ | demo-7c97d7b78c-7... | 10.21.120.132 | default | ip-10-21-120-186.ap-... | 0 | 78d 04h 58m 03s | 0.40 % | 100 % | 0.40 |
| ✓ | dmz-router-extend-13... | 10.21.120.158 | titans | ip-10-21-120-55.ap-n... | 0 | 6h 12m 47s | 1.43 % | 3.50 % | 0.05 |

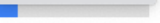
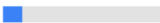
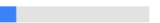
- Real-time data is updated every 10 seconds.
- You can check the total number of Pods and the Pod counts by status on the upper left of the screen.
- It provides the services for searching, column sorting, and detailed view, and [CSV](#) file downloading.

CSV file name, `kubernetes_pod_list_{pcode}_{timestamp}.csv`

❗ In the list-type menus such as [Node list](#), [Container list](#), [Pod list](#), and [Container application list](#), the guides to **Metric information**, **Filter**, and **Column selection** are the same.

Metric information

- If you place your mouse cursor over the desired column, the metrics provided by the column are shown as follows:

| | 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/ |

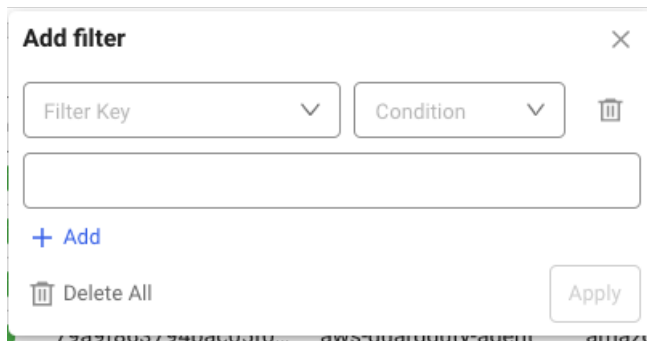
- In the column list of the [Select Column](#) option, you can see the metrics for each column in the column list of the option screen. When you place the mouse cursor over a desired column, the  **INFO icon** appears to the right of the column name. If the mouse cursor is placed over the  **INFO icon**, you can see the metric as follows:

| | | | |
|--|--|---|--|
| ☐ OS | ☐ CpuLoad15 | ☐ MemorySwapTot... | ☐ UsedInodePercent |
| ☒ Image | ☐ CpuNice | ☐ MemorySwapUs... | ☐ IoPercent |
| ☐ Kernel | ☐ CpuSoftIrq | ☐ MemoryTotal | ☒ IoPercentMax |
| ☐ KubeProxy | ☐ CpuSteal | ☒ MemoryUsedPct | ☐ QueueLength |
| ☒ Kubelet | ☒ CpuUsr | | ☐ ReadBps |
| ☒ ContainerRuntime | ☒ Field Unit pct | | ☐ ReadIops |
| | ☒ CpuUsr  | | ☐ TotalSpace |

Filtering

By using the [Filter](#) option, you can select a desired Pod list for view. Through the **and** condition, multiple filters can be applied.

1. If you select + [Add icon](#) in the [Filter](#) field, the [Add filter](#) window appears as follows:



- Select [Filter Key](#) and [Condition](#).
 - Select a value that meets the conditions.
2. Configure a filter and then select [Apply](#) to apply the filter.
 3. To add multiple filters, select + [Add icon](#) in the [Filter](#) field or [Add filter](#) window.

Select Column

When using the [Select column](#) option, you can select the desired Pod data to view the list.

Deselect all(10) Initialize(default) ×

| | | | | |
|--|--|---|--|--|
| ☐ Node Info | ☐ CPU | ☐ Memory | ☐ Disk | ☐ Network |
| ☐ CapacityCpu | ☐ CpuIdle | ☐ MemoryBuffers | ☐ BlkSize | ☐ DroppedIn |
| ☒ Uptime | ☐ CpuIOWait | ☐ MemoryCached | ☐ FreePercent | ☐ DroppedOut |
| ☒ Age | ☐ CpuIrq | ☐ MemoryFree | ☐ FreeSpace | ☐ ErrorIn |
| ☐ Architecture | ☐ CpuLoad1 | ☐ MemoryPageFau... | ☐ TotalNode | ☐ ErrorOut |
| ☐ CapacityMemory | ☐ CpuLoad5 | ☐ MemoryShared | ☐ UsedNode | ☐ HwAddr |
| ☐ OS | ☐ CpuLoad15 | ☐ MemorySwapTot... | ☐ UsedNodePercent | ☐ PacketIn |
| ☒ Image | ☐ CpuNice | ☐ MemorySwapUs... | ☐ IoPercent | ☐ PacketOut |
| ☐ Kernel | ☐ CpuSoftIrq | ☐ MemoryTotal | ☒ IoPercentMax | ☐ TrafficUsedSize |
| ☐ KubeProxy | ☐ CpuSteal | ☒ MemoryUsedPct | ☐ QueueLength | ☒ TrafficUsedSize... |
| ☒ Kubelet | ☐ CpuSys | | ☐ ReadBps | ☐ TrafficIn |
| ☒ ContainerRuntime | ☒ Cpu... | | ☐ ReadIops | ☐ TrafficOut |

Apply

- The selected column list is kept even if you re-enter the menu.
- If you select [Reset](#), the predefined default key metric columns are selected. In this case, the column selection history is deleted.
- After selecting a column, you can click [Apply](#) at the lower right to apply the selected values.

Display Detail

If the [Display Detail](#) icon is selected in Pod, you can see the information details on the Pod.

amex-6ccf9d89f5-gbc24

Resource

Metrics

Log

Event

< 2024/01/11 15:25 >

Container

→ Go to Container list

| ConS... | ContainerId | ContainerName | Image | ConStatus | Pod | CpuByLimit | MemByLimit |
|---------|-------------------|---------------|---------------------------|---------------|-----------------------|------------|------------|
| ✓ | 7cd773f1bcf948... | amex | docker.io/library/tomc... | Up 17 minutes | amex-6ccf9d89f5-gbc24 | 0 % | 71.94 % |

Node

→ Go to Node list

| Status | OnodeName | Age | Image | CapacityCpu | CapacityMemory | Cpu | MemoryUsedPct |
|--------|---------------------|-----------------|----------------|-------------|----------------|--------|---------------|
| ✓ | ip-10-21-120-186... | 80d 03h 52m 16s | Amazon Linux 2 | 2 | 15.33 GiB | 8.75 % | 30.18 % |

Resource relations

Namespace
titans

→

Deployment
amex

→

Pod
amex-6ccf9d89f5-gbc24

→

Container
(amex) 7cd773f1bcf94865f...

+

−

⊕

ⓘ The [Trace](#), [Metrics](#), [Log](#), [Event](#), and [Pod summary analysis](#) tabs of the detailed view window are the same as in Container map's Display Detail.

Resources

After selecting a target, you can see the resource list and relationship diagram for the selected resource in the **Resource** tab in the detailed view.

Summary

It provides a summary of the list of resources for the selected resource. If you select the detailed view icon next to the resource name, the corresponding resource list screen appears.

- **Container:** Goes to **Container list** with only the container filtered.
- **Pod:** Goes to **Pod List** with only the Pod filtered.
- **Node:** Goes to **Node details** with the node name selected.

Relational diagram of resources

You can visually see the relationships and status between resources. As in the list, you can check the detailed metric information by hovering your mouse over a Pod, container, or application. If you select a resource, the resource list page appears.

Pod summary analysis

- Basic Info: You can check the Pod's cluster name, namespace, startup time, current state (Phase), and such.
- Kubernetes Events: You can check if any problem happened by looking at event logs related to the recent Pods.
- Executor: It identifies the controller (DaemonSet, Deployment, etc.) that created the Pod.
- Node Info: It provides detailed specifications of the node on which the Pod is running, including the operating system, kernel version, CPU, and memory.
- Resource Usage: It displays the CPU and memory usage for all containers in the Pod, and the utilization compared to the set resource limits.
- Service networking info: You can see the connected service and Ingress information.

ⓘ Currently, the summary analysis is available only in Korean.

Pod Timeline

Home > Select Project > **Workload** > [Pod Timeline](#)

ⓘ Requirements

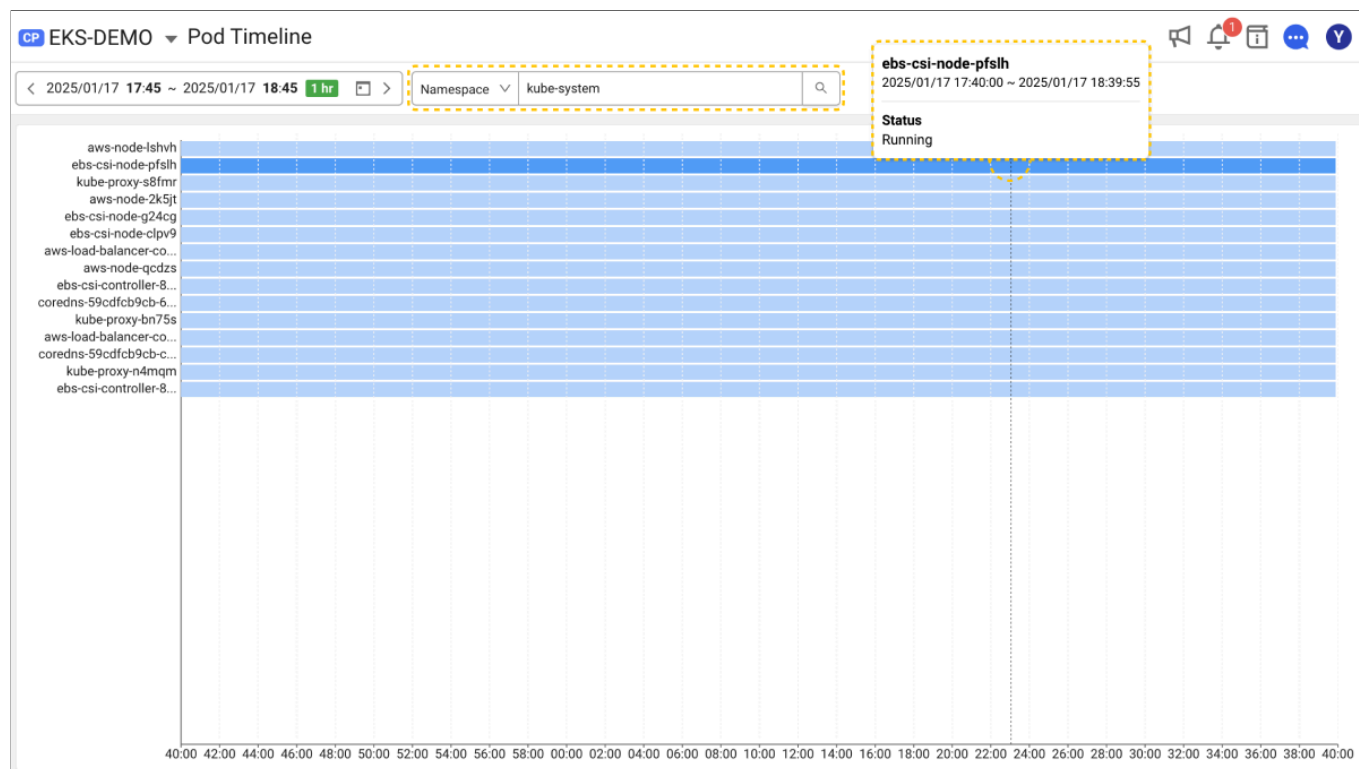
The WhaTap Kubernetes agent 1.7.6 or later is required. Timeline data is provided from the time point collected in version 1.7.6 or later.

The **Pod Timeline** menu provides the feature to visually trace state changes of Pod over time. You can visually see when the terminated Pod was terminated, and which state Pod was in at a specific time point. This allows you to analyze the past status, identify the causes of issues in the cluster, and monitor the operational stability.

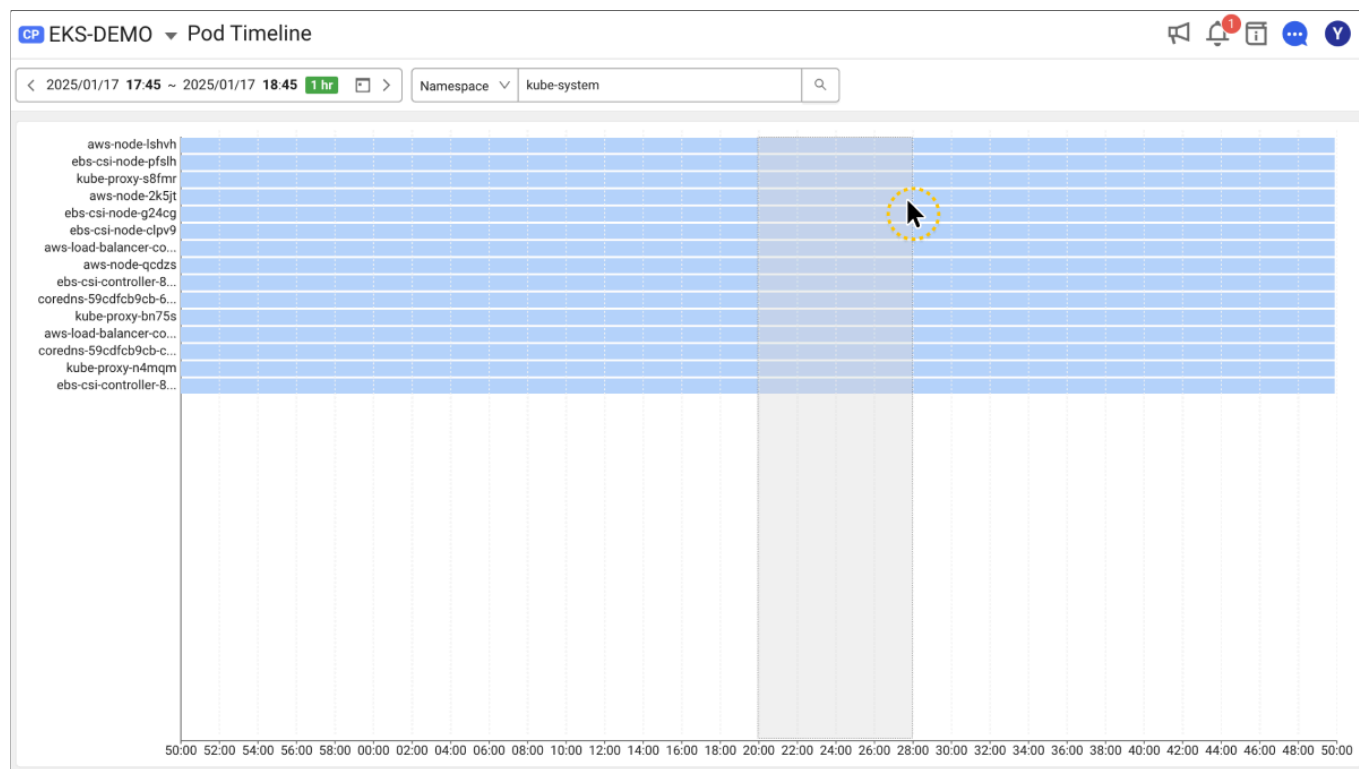
Basic screen guide

After entering the **Pod Timeline** menu, select **Filter key** and **Filter value** to view the data. You can see the Pod list collected in the last 1 hour. You can visually see the state changes of Pod along the time axis.

If you hover your mouse over each time zone, the start and end times of it and Pod status ([Running](#), [Pending](#), [Succeeded](#), [Failed](#)) appear.



You can adjust the time range by dragging the zone on the timeline. Set a desired period to check the past data.

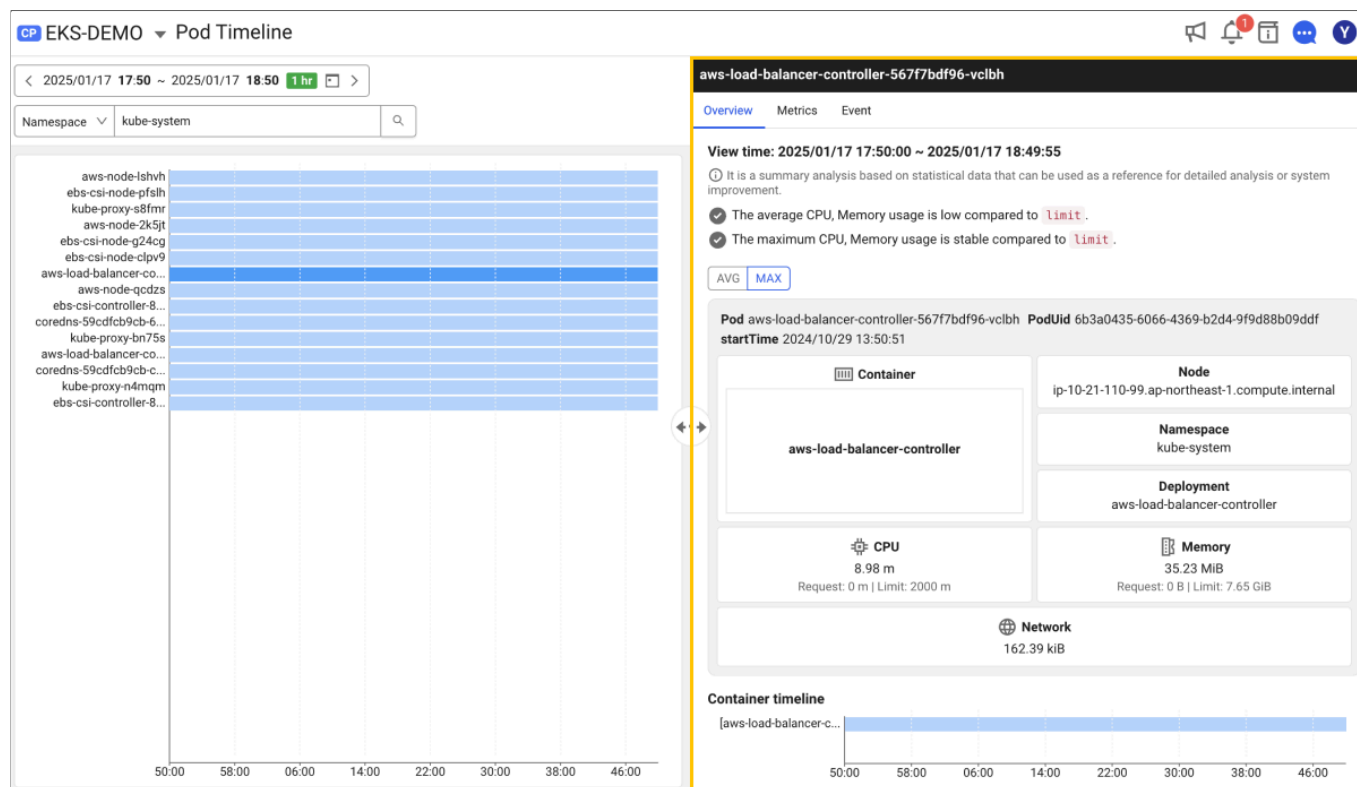


Additionally, when you select a specific period, you can check the Pod status during the period in detail through **Display Detail**. For more information, see the [Pod timeline detail](#) guidance.

! For more information on how to use the time selector.

Pod timeline details

You can check the Pod timeline details by selecting a specific segment or Pod.



- You can view detailed information of a Pod at the time point through the **Display Detail** window on the right by selecting a zone to view.
- You can select a Pod name to view from the list on the left of the chart and view the entire status of the corresponding Pod through the **Display Detail** window on the right.

ⓘ The **Metrics**, **Event** tabs of the detailed view window are the same as in Container map's Display Detail.

Overview

In the **Overview** tab, a summary information of the selected Pod and overall resource usage appears. You can comprehensively see the Pod information such as containers running in the Pod, the node on which the Pod is running, and `ownerReference`. You can see the resource usage and status of the Pod and monitor its performance and operational stability.

- By clicking **AVG** and **MAX**, you can see the average usage and maximum usage for the query period.

- It provides the timeline of the container run in the Pod. If you select a container timeline, the **Container timeline** menu for the selected timeline appears in a new tab.

Pending Pod Status

Home > Select Project > [Workload](#) > [Pending Pod Status](#)

⚠ Requirements

The WhaTap Kubernetes agent 1.7.0 or later is required.

You can comprehensively analyze the cause of delays for each Pod by checking the status of Pods in pending status. The more Pending Pods there are within the cluster, the less efficient and stable the system can be. Maintain the stability of the cluster by monitoring Pods in Pending status to detect and resolve issues early as possible.

Pending Pod list

You can view the list of Pods in Pending status as follows. For each pending Pod, you can briefly check the QoS class, Pod name, timestamp, event message, namespace, and deployment information of the Pod. When you select a Pod, you can view the pending Pod details. For more information, see [the following](#).

CP

Kubernetes Monitoring Demo(EKS)

Pending Pod Status

||

LIVE

16:42:45

No group

All 5

Burstable 2

BestEffort 3

+ There is no filter applied.

Burstable

whatap-virtual-deployment-ff4797d7d-7xdqg

2023-11-20 15:58:58

PodScheduled Unschedulable: 0/5 nodes are available: persistentvolumeclaim "premdata-pvc" not found. preemption: 0/5 nodes are available: 5 Preemption is not helpful for scheduling.

Namespace: default Deployment: whatap-virtual-deployment ReplicasetName: whatap-virtual-deployment-ff4797d7d

Burstable

whatap-virtual-deployment-ff4797d7d-9bjnf

2023-11-20 15:58:58

PodScheduled Unschedulable: 0/5 nodes are available: persistentvolumeclaim "premdata-pvc" not found. preemption: 0/5 nodes are available: 5 Preemption is not helpful for scheduling.

Namespace: default Deployment: whatap-virtual-deployment ReplicasetName: whatap-virtual-deployment-ff4797d7d

BestEffort

virtual-pending-be-7c8df45454-zffmd

2023-11-20 16:07:58

PodScheduled Unschedulable: 0/5 nodes are available: persistentvolumeclaim "premdata-pvc" not found. preemption: 0/5 nodes are available: 5 Preemption is not helpful for scheduling.

Namespace: default Deployment: virtual-pending-be ReplicasetName: virtual-pending-be-7c8df45454

BestEffort

virtual-pending-be-7c8df45454-zrnjk

2023-11-20 16:07:58

PodScheduled Unschedulable: 0/5 nodes are available: persistentvolumeclaim "premdata-pvc" not found. preemption: 0/5 nodes are available: 5 Preemption is not helpful for scheduling.

Namespace: default Deployment: virtual-pending-be ReplicasetName: virtual-pending-be-7c8df45454

BestEffort

qwer-1234

2024-06-12 11:36:13

ContainersReady ContainersNotReady: containers with unready status: [qwer-1234]

Namespace: default

Grouping and QoS Class inquiry

It provides **grouping** for each object. You can easily check the status of all pending Pods related to a specific namespace, node, or replica set, making it easy to identify and analyze problems. You can also check the **number of Pods in Pending status by QoS Class**. When you select a QoS class, you can search for Pods belonging to the QoS class in the list.

||

LIVE

17:37:15

Namespace

All 5

Burstable 2

BestEffort 3

+ There is no filter applied.

▼ Collapse All

▼ default All 2

Burstable 2

Burstable

whatap-virtual-deployment-ff4797d7d-7xdqg

2023-11-20 15:58:58

PodScheduled Unschedulable: 0/5 nodes are available: persistentvolumeclaim "premdata-pvc" not found. preemption: 0/5 nodes are available: 5 Preemption is not helpful for scheduling.

Namespace: default Deployment: whatap-virtual-deployment ReplicasetName: whatap-virtual-deployment-ff4797d7d

Burstable

whatap-virtual-deployment-ff4797d7d-9bjnf

2023-11-20 15:58:58

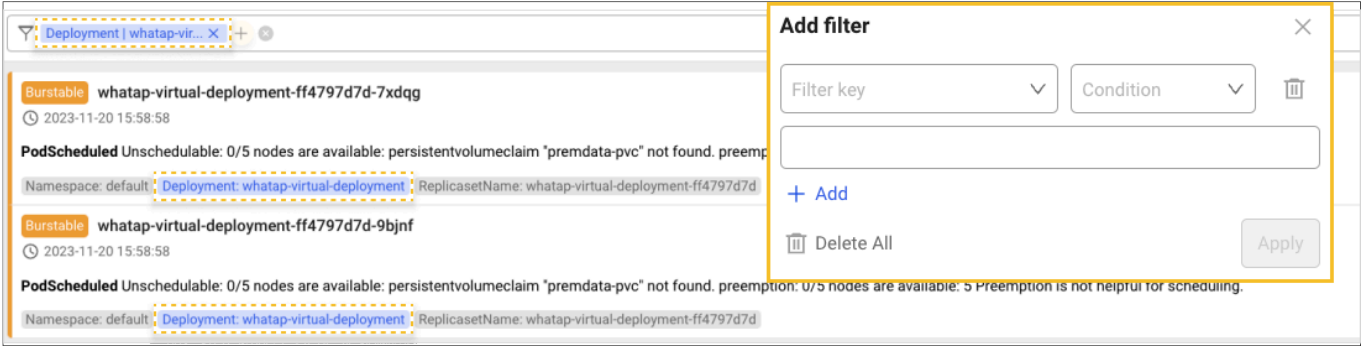
PodScheduled Unschedulable: 0/5 nodes are available: persistentvolumeclaim "premdata-pvc" not found. preemption: 0/5 nodes are available: 5 Preemption is not helpful for scheduling.

Namespace: default Deployment: whatap-virtual-deployment ReplicasetName: whatap-virtual-deployment-ff4797d7d

! For more information about **QoS Class** (Quality of Service Class), see [the following](#) .

Filtering

You can search by additional conditions through filtering. In addition to the filter selection window at the top, you can use the filtering through the tags at the bottom of each pending Pod in the list. When you select a tag, matching conditions for the object are added, and the tags that meet the conditions are displayed in blue.



Pending Pod detail

When you select a Pod from the pending Pod list, you can check detailed information about the Pod through the [Suggestions](#), [Event](#), and [Log](#) tabs as follows:

Kubernetes Monitoring Demo(EKS) Pending Pod Status

CP LIVE 16:43:15 No group All 5 Burstable 2 BestEffort 3

There is no filter applied.

Burstable whatap-virtual-deployment-ff4797d7d-7xdqg
 2023-11-20 15:58:58
PodScheduled Unscheduleable: 0/5 nodes are available: persistentvolumeclaim "premdata-pvc" not found. preemption: 0/5 nodes are available: 5 Preemption is not helpful for scheduling.
 Namespace: default Deployment: whatap-virtual-deployment
 ReplicasetName: whatap-virtual-deployment-ff4797d7d

Burstable whatap-virtual-deployment-ff4797d7d-9bjnf
 2023-11-20 15:58:58
PodScheduled Unscheduleable: 0/5 nodes are available: persistentvolumeclaim "premdata-pvc" not found. preemption: 0/5 nodes are available: 5 Preemption is not helpful for scheduling.
 Namespace: default Deployment: whatap-virtual-deployment
 ReplicasetName: whatap-virtual-deployment-ff4797d7d

BestEffort virtual-pending-be-7c8df45454-zffmd
 2023-11-20 16:07:58
PodScheduled Unscheduleable: 0/5 nodes are available: persistentvolumeclaim "premdata-pvc" not found. preemption: 0/5 nodes are available: 5 Preemption is not helpful for scheduling.
 Namespace: default Deployment: virtual-pending-be ReplicasetName: virtual-pending-be-7c8df45454

BestEffort virtual-pending-be-7c8df45454-zmjk
 2023-11-20 16:07:58
PodScheduled Unscheduleable: 0/5 nodes are available: persistentvolumeclaim "premdata-pvc" not found. preemption: 0/5 nodes are available: 5 Preemption is not helpful for scheduling.
 Namespace: default Deployment: virtual-pending-be ReplicasetName: virtual-pending-be-7c8df45454

Suggestions Event Log

- The Pod's QoS Class is Burstable. If the Pod's schedule needs to be guaranteed, set the container resource request and limit to the same.
- You can check details through the Pod's related events and logs.

Event > Log > Lookup pod manifest [CR](#)

Data(detailed) collection time : 2024-06-12 09:00:36

PodScheduled

Unscheduleable: 0/5 nodes are available: persistentvolumeclaim "premdata-pvc" not found. preemption: 0/5 nodes are available: 5 Preemption is not helpful for scheduling.

There seems to be a problem with the volume mounting process. Check volumeMounts and volume settings.

virtual

```
1 volumeMounts:
2   - mountPath: "/var/run/secrets/kubernetes.io/serviceaccount"
3     name: "kube-api-access-q272d"
4     readOnly: "true"
5
```

volumes

```
1
2   - configMap:
3     defaultMode: "420"
4     name: "keeper.conf"
```

Suggestions

The following provides various information on why the Pod is in Pending status and for troubleshooting it.

- QoS Class analysis**

You can check scheduling recommendations by analyzing the QoS class.

- Detailed data collection time**

You can see when those details were collected.

- Pending Pod event messages and troubleshooting suggestions**

It checks event messages, identifies problems, and suggests solutions based on them.

- Information about issues**

It provides information about the problems occurred. For example, if there is a problem with the volume mounting process, it guides you to the volume settings. If there is a problem with node resources, it guides you to information to check the resource.

Event

It provides information about events from the container running in the selected Pod. It is the same as the [Event](#) tab in [Container Map](#). For more information, see [the following](#).

Log

It provides information about the logs of the container running in the selected Pod. It is the same as the [Log](#) tab in [Container Map](#). For more information, see [the following](#).

ⓘ Container logs can be viewed only after running the container. If a Pod remains in pending status before running the container, it is impossible to check the container log for the Pod.

Container Application List

Home > Select Project > **Workload** > **Container Application List**

It displays the list of containerized applications. If you select a desired column, you can search the application list sorted by that column.

Kubernetes Monitoring Demo(EKS)

Container Application List

||

LIVE 17:24:55

⌵

+

There is no filter applied.

Select Column

⬇

CSV

| Application information | | | User | Active transaction | | | |
|-------------------------|-----------|------|----------------|--------------------|----------|--------|--------|
| ApplicationName | OkindName | Type | ConcurrentUser | DbConnection | HttpCall | Method | Socket |
| virtual1 | virtual | java | 994 | 2 | 21 | 18 | 0 |
| virtual10 | virtual | java | 988 | 3 | 17 | 15 | 0 |
| virtual11 | virtual | java | 981 | 4 | 21 | 17 | 0 |
| virtual12 | virtual | java | 978 | 3 | 23 | 21 | 0 |
| virtual2 | virtual | java | 988 | 0 | 15 | 14 | 0 |
| virtual5 | virtual | java | 978 | 4 | 16 | 15 | 0 |
| virtual6 | virtual | java | 973 | 2 | 19 | 13 | 0 |
| virtual7 | virtual | java | 988 | 1 | 18 | 17 | 0 |
| virtual8 | virtual | java | 981 | 2 | 24 | 20 | 0 |
| virtual9 | virtual | java | 986 | 2 | 21 | 19 | 0 |

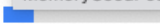
- Real-time data is updated every 10 seconds.
- It provides the services for searching, column sorting, and detailed view, and [CSV](#) file downloading.

CSV file name, `kubernetes_application_list_{pcode}_{timestamp}.csv`

ⓘ In the list-type menus such as [Node list](#), [Container list](#), [Pod list](#), and [Container application list](#), the guides to **Metric information**, **Filter**, and **Column selection** are the same.

Metric information

- If you place your mouse cursor over the desired column, the metrics provided by the column are shown as follows:

| | | 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/ |

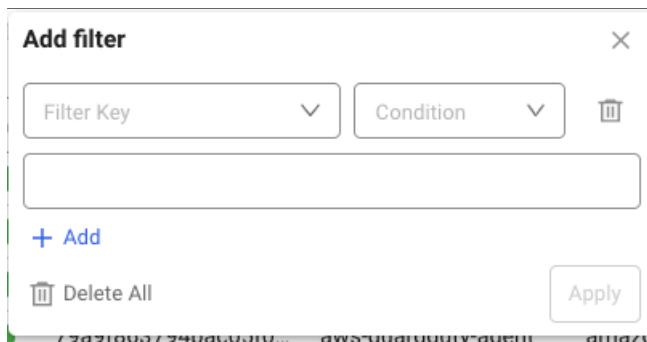
- In the column list of the [Select Column](#) option, you can see the metrics for each column in the column list of the option screen. When you place the mouse cursor over a desired column, the ⓘ [INFO icon](#) appears to the right of the column name. If the mouse cursor is placed over the ⓘ [INFO icon](#), you can see the metric as follows:

| | | | |
|--|--|---|--|
| ☐ OS | ☐ CpuLoad15 | ☐ MemorySwapTot... | ☐ UsedInodePercent |
| ☒ Image | ☐ CpuNice | ☐ MemorySwapUs... | ☐ IoPercent |
| ☐ Kernel | ☐ CpuSoftIrq | ☐ MemoryTotal | ☒ IoPercentMax |
| ☐ KubeProxy | ☐ CpuSteal | ☒ MemoryUsedPct | ☐ QueueLength |
| ☒ Kubelet | ☒ CpuUsr ⓘ | | ☐ ReadBps |
| ☒ ContainerRuntime | | | ☐ ReadIops |
| | | | ☐ TotalSpace |

Filtering

By using the [Filter](#) option, you can select a desired Application list for view. Through the [and](#) condition, multiple filters can be applied.

- If you select + [Add icon](#) in the [Filter](#) field, the [Add filter](#) window appears as follows:



- Select [Filter Key](#) and [Condition](#).
 - Select a value that meets the conditions.
2. Configure a filter and then select [Apply](#) to apply the filter.
 3. To add multiple filters, select + [Add icon](#) in the [Filter](#) field or [Add filter](#) window.

Select Column

When using the [Select column](#) option, you can select the desired Application list data to view the list.

Deselect all(10) Initialize(default) ×

| | | | | |
|--|--|---|--|--|
| ☐ Node Info | ☐ CPU | ☐ Memory | ☐ Disk | ☐ Network |
| ☐ CapacityCpu | ☐ CpuIdle | ☐ MemoryBuffers | ☐ BlkSize | ☐ DroppedIn |
| ☒ Uptime | ☐ CpuIOWait | ☐ MemoryCached | ☐ FreePercent | ☐ DroppedOut |
| ☒ Age | ☐ CpuIrq | ☐ MemoryFree | ☐ FreeSpace | ☐ ErrorIn |
| ☐ Architecture | ☐ CpuLoad1 | ☐ MemoryPageFau... | ☐ TotalNode | ☐ ErrorOut |
| ☐ CapacityMemory | ☐ CpuLoad5 | ☐ MemoryShared | ☐ UsedNode | ☐ HwAddr |
| ☐ OS | ☐ CpuLoad15 | ☐ MemorySwapTot... | ☐ UsedNodePercent | ☐ PacketIn |
| ☒ Image | ☐ CpuNice | ☐ MemorySwapUs... | ☐ IoPercent | ☐ PacketOut |
| ☐ Kernel | ☐ CpuSoftIrq | ☐ MemoryTotal | ☒ IoPercentMax | ☐ TrafficUsedSize |
| ☐ KubeProxy | ☐ CpuSteal | ☒ MemoryUsedPct | ☐ QueueLength | ☒ TrafficUsedSize... |
| ☒ Kubelet | ☐ CpuSys | | ☐ ReadBps | ☐ TrafficIn |
| ☒ ContainerRuntime | ☒ Cpu... | | ☐ ReadIops | ☐ TrafficOut |

Apply

- The selected column list is kept even if you re-enter the menu.
- If you select [Reset](#), the predefined default key metric columns are selected. In this case, the column selection history is deleted.
- After selecting a column, you can click [Apply](#) at the lower right to apply the selected values.

Display Detail

If the [Display Detail](#) icon is selected in Application, you can see the information details on the Application.

virtual1

Resource

Trace

Metrics

Log

Event

Call information

2024/03/15 09:33

Container

→ Go to Container list

| | ContainerId | ContainerName | Image | ConStatus | Pod | CpuByLimit | MemByLimit |
|---|------------------|---------------|---------------------------|-----------------------|--------------------------|------------|------------|
| ✓ | 853d2618dd518... | virtual | docker.io/whatap/virtu... | Up 9 hours 19 minutes | whatap-virtual-deploy... | 28.11% | 67.48% |

Pod

→ Go to Pod list

| | Pod | containerIdsCount | ReplicaSet | Deployment | Namespace | CpuByLimit | MemByLimit |
|---|---------------------|-------------------|--------------------------|--------------------------|--------------|------------|------------|
| ✓ | whatap-virtual-d... | 1 | whatap-virtual-deploy... | whatap-virtual-deploy... | virtual-main | 28.11% | 67.48% |

Node

→ Go to Node list

| | OnodeName | Age | Image | CapacityCpu | CapacityMemory | Cpu | MemoryUsedPct |
|---|--------------------|----------------|----------------|-------------|----------------|--------|---------------|
| ✓ | ip-10-21-13-157... | 1d 09h 22m 10s | Amazon Linux 2 | 2 | 1.88GiB | 10.90% | 65.91% |

Resource relations

ip-10-21-13-157.ap-northeast-2.compute.internal

Namespace virtual-main

Deployment whatap-virtu al-deployment

ReplicaSet whatap-virtu al-deploymen...

Pod whatap-virtu al-deploymen...

Container (virtual) 85 3d2618dd518a...

ⓘ The [Container summary analysis](#), [Trace](#), [Metrics](#), [Log](#), [Event](#), and [Call info](#) tabs of the Display Detail window are the same as those of the Container map's Display Detail.

Resources

After selecting a target, you can see the resource list and relationship diagram for the selected resource in the **Resource** tab in the detailed view.

Summary

It provides a summary of the list of resources for the selected resource. If you select the detailed view icon next to the resource name, the corresponding resource list screen appears.

- **Container:** Goes to **Container list** with only the container filtered.
- **Pod:** Goes to **Pod List** with only the Pod filtered.
- **Node:** Goes to **Node details** with the node name selected.

382

Relational diagram of resources

You can visually see the relationships and status between resources. As in the list, you can check the detailed metric information by hovering your mouse over a Pod, container, or application. If you select a resource, the resource list page appears.

Deployment List

Home > Select Project > [Workload](#) > [Deployment List](#)

The deployment list appears. It helps maintain the system stability and performance by monitoring the overall operation status of applications that consist of multiple Pods. When you select a desired column in the [Deployment list](#), you can sort and view the list based on the column.

| CP Kubernetes Monitoring Demo(EKS) Deployment List | | | | | | | |
|---|------------|-----------|-------------|-----------------|-------------------|---------------------|---------|
| <div> <div> <div> </div> <div>LIVE 14:21:05</div> </div> <div> <input type="text" value="Deployment Name"/> <div>Q</div> </div> <div> <div>⌵</div> <div>Select Column</div> </div> <div> <div>⬇</div> <div>CSV</div> </div> </div> | | | | | | | |
| Deployment | Resource | | | | | | |
| | ReplicaSet | Total Pod | Running Pod | Total Container | Running Container | Total Container CPU | Total C |
| apm-java-demo | 1 | 1 | 1 | 1 | 1 | 0m | |
| coredns | 1 | 2 | 2 | 2 | 2 | 2m | |
| ebs-csi-controll... | 1 | 2 | 2 | 12 | 12 | 4m | |
| nodejs-sample-... | 1 | 1 | 1 | 1 | 1 | 28m | |
| php-demo-apa... | 1 | 1 | 1 | 1 | 1 | 25m | |
| 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 | 93m | |
| whatap-virtual-... | 1 | 10 | 10 | 10 | 10 | 688m | |

- Real-time data is updated every 10 seconds.
- It provides the services for searching, column sorting, and detailed view, and [CSV](#) file downloading.

CSV file name, `kubernetes_deployment_list_{pcode}_{timestamp}.csv`

Select Column

When using the [Select column](#) option, you can select the desired Deployment data to view the list.

| Select all | Deselect all(17) | Initialize(default) | × |
|---|--|---------------------|---|
| ☒ Resource | ☒ Application | | |
| ☒ ReplicaSet | ☒ Application | | |
| ☒ Total Pod | ☒ Normal Active Tr... | | |
| ☒ Running Pod | ☒ Slow Active Tran... | | |
| ☒ Total Container | ☒ Very Slow Active... | | |
| ☒ Running Container | ☒ Total Active Tran... | | |
| ☒ Total Container ... | ☒ Total Transaction | | |
| ☒ Total Container ... | ☒ HTTP Connection | | |
| ☒ Total Network R... | ☒ SQL Connection | | |
| ☒ Total Network Tr... | | | |

Apply

- The selected column list is kept even if you re-enter the menu.
- If you select [Reset](#), the predefined default key metric columns are selected. In this case, the column selection history is deleted.
- After selecting a column, you can click [Apply](#) at the lower right to apply the selected values.

Display Detail

If the [Display Detail](#) icon is selected in Deployment, you can see the information details on the Deployment.

- **Pod:** Goes to **Pod List** with only the Pod filtered.
- **Node:** Goes to **Node details** with the node name selected.
- [ReplicaSet](#): It provides a summary for active Pods and containers based on [ReplicaSet](#).

Relational diagram of resources

You can visually see the relationships and status between resources. As in the list, you can check the detailed metric information by hovering your mouse over a Pod, container, or application. If you select a resource, the resource list page appears.

Service List

Home > Project Select > **Service Networking** > **Service List**

ⓘ The WhaTap Kubernetes agent 1.7.8 or later is required.

It views **Service List**. Through **Service List** of WhaTap Kubernetes Monitoring, you can easily check the network configuration within the cluster and efficiently monitor the configuration and status of each service.

In Kubernetes, a **service** is an object for accessing Pods inside and outside the cluster. In **Service List**, you can check the detailed configuration of each service at a glance, and monitor the status of connected Pods in real time. This allows you to quickly identify and resolve abnormal connectivity issues between services and Pods.

Basic screen guide

You can check the **Service List** information existing in the cluster as follows. The list is displayed based on the service information collected over the last minute based on the query time. You can identify and manage resources by various criteria such as namespace and label selector, and clearly check various properties of the service.

CP

EKS-DEMO

▼ Service List

🔊

🔔

📅

⋮

H

|| LIVE 14:33:42

Pod Phase

Running

Pending

Succeeded

Failed

Unknown

🔍 + There is no filter applied.

Select Column

📄 CSV

| | serviceName | podPhase | namespace | creationTimeMillis | serviceType | clusterIP | externalIP | selector | |
|---|-----------------------|----------|-------------|---------------------|--------------|----------------|------------|---------------------------|-------|
| 🔍 | aws-load-balance... | | kube-system | 2024/06/11 15:38:19 | ClusterIP | 172.20.107.7 | - | app.kubernetes.io/in... | - |
| 🔍 | callee-svc | | java | 2024/06/05 16:05:30 | NodePort | 172.20.1.224 | - | app=callee-java | 32162 |
| 🔍 | callee-with-db-svc | | java | 2024/06/10 16:16:13 | NodePort | 172.20.252.159 | - | app=callee-with-db | 32540 |
| 🔍 | caller-svc | 1 | java | 2024/06/10 18:43:43 | NodePort | 172.20.175.108 | - | app=caller-java | 31322 |
| 🔍 | ext-svc | | batch | 2024/07/15 19:16:58 | ExternalName | | - | - | - |
| 🔍 | kube-dns | | kube-system | 2024/04/08 14:55:39 | ClusterIP | 172.20.0.10 | - | k8s-app=kube-dns | - |
| 🔍 | kubernetes | | default | 2024/04/08 14:53:47 | ClusterIP | 172.20.0.1 | - | - | - |
| 🔍 | metrics-server | | kube-system | 2024/07/19 16:31:55 | ClusterIP | 172.20.116.44 | - | k8s-app=metrics-serv... | - |
| 🔍 | postgres | | java | 2024/06/10 18:23:14 | ClusterIP | 172.20.72.145 | - | app=postgres | - |
| 🔍 | postgres-agent | 1 | java | 2024/06/14 17:47:34 | ClusterIP | 172.20.51.56 | - | app=postgres-agent | - |
| 🔍 | python-fastapi-ser... | | default | 2024/06/05 15:32:21 | NodePort | 172.20.107.189 | - | app=python-fastapi-p... | 30662 |
| 🔍 | svc-cache | | backend | 2024/06/14 16:49:35 | NodePort | 172.20.76.228 | - | app=cache-server, tie... | 30923 |
| 🔍 | svc-catserver | | backend | 2024/06/10 10:35:59 | NodePort | 172.20.158.182 | - | app=catserver, tier=to... | 30530 |
| 🔍 | svc-customer | | backend | 2024/06/07 17:29:09 | NodePort | 172.20.220.5 | - | app=customer, tier=b... | 32154 |
| 🔍 | svc-front-ui | | frontend | 2024/06/10 09:35:28 | NodePort | 172.20.208.84 | - | app=nginx | 32393 |
| 🔍 | svc-help | | backend | 2024/06/07 17:26:14 | NodePort | 172.20.36.242 | - | app=help, tier=python | 30761 |
| 🔍 | svc-himalaya | | backend | 2024/06/07 19:27:31 | NodePort | 172.20.253.1 | - | app=himalaya, tier=b... | 31303 |
| 🔍 | svc-kubecon2023 | 2 | edu | 2024/06/19 14:57:06 | NodePort | 172.20.107.211 | - | app=kubecon2023, ti... | 32204 |
| 🔍 | svc-mail-db | | backend | 2024/06/13 12:48:31 | NodePort | 172.20.123.115 | - | app=mail, tier=postgr... | 31213 |

! Pod Phase

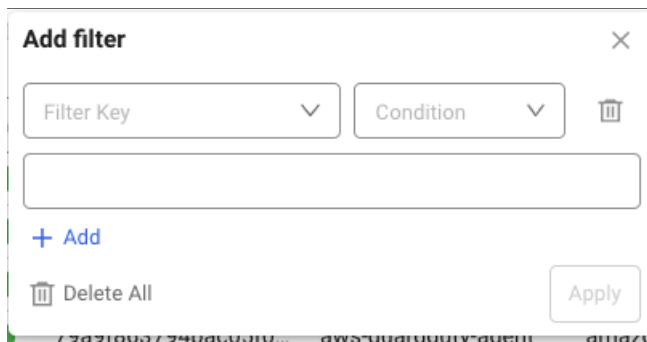
The **podPhase** column displays the numbers of Pods connected to the service for each state. The status of a Pod is distinguished in colors. See the [Pod Phase](#) guidance above regarding the Pod status.

! The Filter and Select Column instructions in the list menu are the same.

Filtering

By using the **Filter** option, you can select a desired for view. Through the **and** condition, multiple filters can be applied.

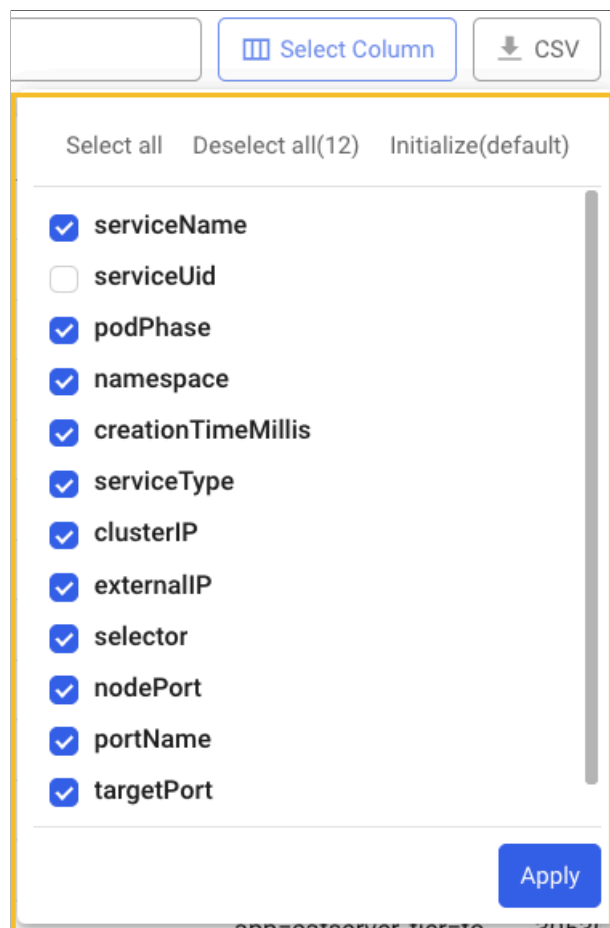
1. If you select **+ Add icon** in the **Filter** field, the **Add filter** window appears as follows:



- Select [Filter Key](#) and [Condition](#).
 - Select a value that meets the conditions.
2. Configure a filter and then select [Apply](#) to apply the filter.
 3. To add multiple filters, select [+ Add icon](#) in the [Filter](#) field or [Add filter](#) window.

Select Column

When using the [Select column](#) option, you can select the desired data to view the list.



- The selected column list is kept even if you re-enter the menu.
- If you select [Reset](#), the predefined default key metric columns are selected. In this case, the column selection history is deleted.
- After selecting a column, you can click [Apply](#) at the lower right to apply the selected values.

Display Detail

When you select the  **Display Detail** icon of the service to search in the list, you can see the list of Pods connected to the service.

✕ kube-dns

< 2024/07/17 14:02 >

Pod >

| | Pod | Namespace | containerIdsCount | CpuByLimit | MemByLimit | ReplicaSet | Deployment | Daer |
|---|------------------------|-------------|-------------------|------------|------------|--------------------|------------|------|
| ✓ | coredns-59cdfcb9cb-... | kube-system | 1 | 0.43 % | 9.46 % | coredns-59cdfcb9cb | coredns | - |
| ✓ | coredns-59cdfcb9cb-... | kube-system | 1 | 0.20 % | 10.25 % | coredns-59cdfcb9cb | coredns | - |

Resource relations

Namespace
kube-system

Service
kube-dns

Pod
coredns-59
cdfcb9cb-m...

Pod
coredns-59
cdfcb9cb-v...

ip-10-21-110-49.ap-northeast-1.compute.internal

ip-10-21-140-228.ap-northeast-1.compute.internal

+

↑

↓

-

⊕

Ingress List

Home > Project Select > **Service Networking** > **Ingress List**

 The WhaTap Kubernetes agent 1.7.13 or later is required.

It views **Ingress List**. Requests from outside Kubernetes are ultimately routed to the applications inside the container through **Ingress** and connected to a specific service or Pod depending on the Ingress configuration. Ingress Monitoring plays an important role in understanding and managing the flow of outbound traffic into the Kubernetes cluster.

WhaTap's **Ingress List** provides a list view that visualizes routing policies for each Ingress. This gives you a quick overview of how each Ingress is distributing traffic.

Basic screen guide

You can view the major Ingress policy-based data and the numbers of Pods and containers linked to each policy by status, based on the Ingress policy as follows. You can comprehensively view the connection status and major object information for each Ingress policy through the list and detailed view. You can also check the icons for each status of Pods and containers at the top of the list.

CP EKS-DEMO ▾ Ingress List



 LIVE 11:39:46

 + There is no filter applied.

 Column Settings

 CSV

| PodPhase  Running  Pending  Succeeded  Failed  Unknown ContainerState  Running  Waiting  Terminated | | | | | | | | |
|---|-------------------|--|----------|---|------------------|------------------|------------------------|---------------------|
| De... | IngressName | Rules | PodPhase | ContainerState | IngressNamespace | IngressClassName | IngressLoadBalancer... | CreationTime |
|  | ingress-catserver | catserver.whatapk8s.net, | | | backend | alb | - | 2024/06/11 16:22:25 |
|  | ingress-front | japan.whatapk8s.net/ ⇄  1 | |  1 | frontend | alb | - | 2024/06/11 15:49:11 |
|  | ingress-help | help.whatapk8s.net/ ⇄ : | | | backend | alb | - | 2024/06/11 18:08:06 |

 The **Filter** and **Select Column** instructions in the list menu are the same.

Filtering

By using the [Filter](#) option, you can select a desired for view. Through the [and](#) condition, multiple filters can be applied.

1. If you select [+ Add icon](#) in the [Filter](#) field, the [Add filter](#) window appears as follows:

- Select [Filter Key](#) and [Condition](#).
 - Select a value that meets the conditions.
2. Configure a filter and then select [Apply](#) to apply the filter.
 3. To add multiple filters, select [+ Add icon](#) in the [Filter](#) field or [Add filter](#) window.

Select Column

When using the [Select column](#) option, you can select the desired data to view the list.

Select column

- The selected column list is kept even if you re-enter the menu.
- If you select [Reset](#), the predefined default key metric columns are selected. In this case, the column selection history is deleted.
- After selecting a column, you can click [Apply](#) at the lower right to apply the selected values.

Display Detail

When you select the  **Display Detail** icon of the Ingress policy to search in the list, you can see the information details on the Ingress policy.

X
catserver.whatapk8s.net/ ⇌ svc-catserver:8000

Overview
Service
Pod
Container

▶ catserver.whatapk8s.net/ ⇌ svc-catserver:8000
Object Manifest

| | | | |
|-------------------------------|--------------------------------------|---------------------------|-------------------------|
| IngressName | ingress-catserver | Host | catserver.whatapk8s.net |
| IngressUid | 0032d8ed-a29d-4bf4-9bc3-b525c6cef187 | Path | / |
| IngressNamespace | backend | BackendServiceName | svc-catserver |
| IngressLoadBalancerips | - | BackendServicePort | 8000 |
| PathType | Prefix | | |

Traffic coming to `catserver.whatapk8s.net/` is forwarded to the service `svc-catserver:8000` via the `ingress-catserver` Ingress policy in the `backend` namespace. `ingress-catserver` Ingress was created at 2024/06/11 16:22:25.

- ✔ No container was found to connect to the `svc-catserver:8000` service. Check whether a Pod matching the Service selector exists.
- ✔ LoadBalancer information not found.

Namespace
backend

→

Ingress
ingress-ca
tserver(ca...

→

Service
svc-catser
ver(port:8...

Summary

It provides **Overview** analytics for Ingress basics and the selected Ingress policy. You can see the Ingress namespace, associated hosts and routes, and the backend service data with which traffic is transmitted. This allows us to look into the flow of outbound traffic and the structure of Ingress policies. Additionally, you can visualize the relationship diagram of major objects connected through the policy, allowing you to see the resource relationships within the policy at a glance.

 You can see the [Object manifest](#) information at the top of the **Overview** tab.

Service

It provides a list of [services](#) linked to the selected Ingress policy. This allows you to see which services Ingress is connected to. You can view the connection status of each service and service details.

Pod

It provides the **Pod** list linked to the selected Ingress policy. You can see the status, basic information, and summary analysis for each connected Pod.

Container

It provides the **Container** list linked to the selected Ingress policy. You can see the status, performance, and summary analysis for containers that pass traffic according to the Ingress policy.

PV/PVC List

ⓘ Requirements

- The WhaTap Kubernetes agent 1.8.1 or later is required.
- If it is a Kubernetes cluster project (CP), you can view the corresponding menu.

Persistent Volume (PV) indicates a physical storage resource available in the Kubernetes cluster. **Persistent Volume Claim** (PVC) is an object that declaratively defines the storage requirements of the Pod. PVC is associated with a PV and provides an access path for the Pod to read and write data.

WhaTap Kubernetes can monitor the storage status of the cluster through the [Persistent Volume List](#) and [Persistent Volume Claim List](#) menus. This menu intuitively provides the usage status and connection relationships of storage resources, and allows you to check information such as storage capacity, access mode, and event status at a glance.

Persistent Volume List

Home > Select Project > **Storage** > [Persistent Volume List](#)

In [Persistent Volume List](#), you can view the PVs that exist in the cluster. The list is displayed based on the PV information collected over the last minute on the query time. You can intuitively check the connection with PVC and the binding of PV. You can also easily check the PVC information details associated with the PV.

CP

EKS-DEMO

▼ Persistent Volume List

II

LIVE

14:13:05

🔍

There is no filter applied.

Select Column

📄 CSV

| PersistentVolume | | | | | | | | |
|------------------------|---------|----------------------|------|-----------------|--------------------|--------------|------------|---|
| PvName | PvPhase | PvType | Path | PvReclaimPolicy | PvStorageClassName | PvVolumeMode | PvCapacity | P |
| pvc-088b92e4-aaa1-4... | Bound | awsElasticBlockStore | - | Delete | mall-db-ebs | Filesystem | 20.00 GiB | |
| pvc-0aac1d7a-08e9-4... | Bound | awsElasticBlockStore | - | Delete | gp2 | Filesystem | 1.00 GiB | |
| pvc-18a17078-2fe2-4... | Bound | awsElasticBlockStore | - | Delete | gp2 | Filesystem | 1.00 GiB | |
| pvc-57585a2c-ce65-4... | Bound | awsElasticBlockStore | - | Delete | gp2 | Filesystem | 1.00 GiB | |
| pvc-9f991935-02db-4... | Bound | awsElasticBlockStore | - | Delete | gp2 | Filesystem | 1.00 GiB | |
| pvc-ab958b08-aa81-4... | Bound | awsElasticBlockStore | - | Delete | gp2 | Filesystem | 1.00 GiB | |
| pvc-b1281c9a-5ec4-4... | Bound | awsElasticBlockStore | - | Delete | gp2 | Filesystem | 1.00 GiB | |
| pvc-b3bc2248-d0ff-4... | Bound | awsElasticBlockStore | - | Delete | gp2 | Filesystem | 1.00 GiB | |
| pvc-cd641924-1a1e-4... | Bound | awsElasticBlockStore | - | Delete | gp2 | Filesystem | 1.00 GiB | |
| pvc-ce50683c-a74b-4... | Bound | awsElasticBlockStore | - | Delete | mall-db-ebs | Filesystem | 20.00 GiB | |
| pvc-d4608536-a190-4... | Bound | awsElasticBlockStore | - | Delete | gp2 | Filesystem | 1.00 GiB | |
| pvc-de33f55c-30e9-4... | Bound | awsElasticBlockStore | - | Delete | gp2 | Filesystem | 1.00 GiB | |
| pvc-e2af8cbe-481a-4... | Bound | awsElasticBlockStore | - | Delete | gp2 | Filesystem | 1.00 GiB | |
| pvc-f9a493f1-7dde-4... | Bound | awsElasticBlockStore | - | Delete | gp2 | Filesystem | 1.00 GiB | |

- If the CSV button is clicked, you can download the CSV file.

Persistent Volume Claim List

Home > Select Project > **Storage** > [Persistent Volume Claim List](#)

In [Persistent Volume Claim List](#), you can view the PVCs that exist in the cluster. The list is displayed based on the PVC information collected over the last minute on the query time. You can intuitively check the connection with PV and the binding of PVC. You can also easily check the PV information details associated with the PVC.

| CP EKS-DEMO ▾ Persistent Volume Claim List | | | | | | 🔊 🔔 18 📄 ⋮ 👤 | | |
|--|----------|--------------|----------------------|----------------|---------------------|--|---------|--------|
| <div> <div>⏸ LIVE 14:17:43</div> <div>🔍 + There is no filter applied.</div> </div> | | | | | | <div> <div>📄 Select Column</div> <div>📄 CSV</div> </div> | | |
| PersistentVolumeClaim | | | | | | | | |
| pvcName | PvcPhase | pvcNamespace | PvcStorageClassNa... | pvcAccessModes | pvcRequestedStorage | PvName | PvPhase | |
| mall-db-ebs-pvc | 🟢 Bound | default | mall-db-ebs | ReadWriteOnce | 20.00 GiB | pvc-088b92e4-a... | 🟢 Bound | awsEla |
| redis-claim-redis-1 | 🟢 Bound | backend | gp2 | ReadWriteOnce | 1.00 GiB | pvc-0aac1d7a-0... | 🟢 Bound | awsEla |
| data-redis-cluster-4 | 🟢 Bound | backend | gp2 | ReadWriteOnce | 1.00 GiB | pvc-18a17078-2f... | 🟢 Bound | awsEla |
| data-redis-cluster-2 | 🟢 Bound | backend | gp2 | ReadWriteOnce | 1.00 GiB | pvc-57585a2c-c... | 🟢 Bound | awsEla |
| data-redis-cluster-1 | 🟢 Bound | backend | gp2 | ReadWriteOnce | 1.00 GiB | pvc-9f991935-02... | 🟢 Bound | awsEla |
| redis-claim-redis-2 | 🟢 Bound | backend | gp2 | ReadWriteOnce | 1.00 GiB | pvc-ab958b08-a... | 🟢 Bound | awsEla |
| data-redis-cluster-5 | 🟢 Bound | backend | gp2 | ReadWriteOnce | 1.00 GiB | pvc-b1281c9a-5... | 🟢 Bound | awsEla |
| data-redis-cluster-3 | 🟢 Bound | backend | gp2 | ReadWriteOnce | 1.00 GiB | pvc-b3bc2248-d... | 🟢 Bound | awsEla |
| data-redis-cluster-0 | 🟢 Bound | backend | gp2 | ReadWriteOnce | 1.00 GiB | pvc-cd641924-1... | 🟢 Bound | awsEla |
| mall-db-ebs-pvc | 🟢 Bound | backend | mall-db-ebs | ReadWriteOnce | 20.00 GiB | pvc-ce50683c-a... | 🟢 Bound | awsEla |
| redis-data-redis-1 | 🟢 Bound | backend | gp2 | ReadWriteOnce | 1.00 GiB | pvc-d4608536-a... | 🟢 Bound | awsEla |
| redis-data-redis-0 | 🟢 Bound | backend | gp2 | ReadWriteOnce | 1.00 GiB | pvc-de33f55c-30... | 🟢 Bound | awsEla |
| redis-claim-redis-0 | 🟢 Bound | backend | gp2 | ReadWriteOnce | 1.00 GiB | pvc-e2af8cbe-48... | 🟢 Bound | awsEla |
| redis-data-redis-2 | 🟢 Bound | backend | gp2 | ReadWriteOnce | 1.00 GiB | pvc-f9a493f1-7d... | 🟢 Bound | awsEla |

- If the [CSV](#) button is clicked, you can download the [CSV](#) file.
- Click the > button on the right of the [PvName](#) column. Then the [Persistent Volume List](#) menu appears.

Filtering

By using the [Filter](#) option, you can select a desired for view. Through the [and](#) condition, multiple filters can be applied.

1. If you select + [Add icon](#) in the [Filter](#) field, the [Add filter](#) window appears as follows:

Add filter

Filter Key

Condition

🗑

+ Add

🗑 Delete All

Apply

- Select [Filter Key](#) and [Condition](#).

- Select a value that meets the conditions.
2. Configure a filter and then select [Apply](#) to apply the filter.
 3. To add multiple filters, select + [Add icon](#) in the [Filter](#) field or [Add filter](#) window.

Columns

| Group | Column name | Description |
|------------------|------------------------------------|--|
| PersistentVolume | PvUid | Unique ID of the persistent volume. |
| | PvName | Name of the persistent volume. |
| | PvPhase | It displays the current status of the persistent volume. |
| | PvType | Storage type of the persistent volume. |
| | Path | It indicates the mounting path of the persistent volume. |
| | PvReclaimPolicy | It indicates the reclamation policy of the persistent volume. |
| | PvStorageClassName | Name of the storage class assigned to the PV. |
| | PvVolumeMode | It indicates the volume mode property of the persistent volume. |
| | PvCapacity | It displays the total storage capacity of the persistent volume. |
| | PvMessage | This message explains why the current persistent volume had a specific phase. |
| | PvReason | This keyword displays the errors related to the persistent volumes, briefly indicating the cause of the problem or the current status. |

| Group | Column name | Description |
|-----------------------|---------------------|--|
| PersistentVolumeClaim | PvcUid | Unique ID of the persistent volume claim. |
| | PvcName (PVC) | Name of the persistent volume claim. |
| | PvcNamespace | It indicates the namespace to which the persistent volume claim belongs. |
| | PvcStorageClassName | Name of the storage class set for the persistent volume claim. |
| | PvcAccessModes | It indicates the access mode of the persistent volume claim. |
| | PvcRequestedStorage | Storage capacity requested by the persistent volume claim. |

- ⓘ • For more information about the reclamation policy of the persistent volume, see [the following](#).
[Retain](#), [Delete](#)
- For more information about the volume mode of the persistent volume, see [the following](#).
[Filesystem](#), [Block](#)
- For more information about the access mode of the persistent volume, see [the following](#).
[ReadWriteOnce](#), [ReadOnlyMany](#), [ReadWriteMany](#)

Phase

PvPhase

[PvPhase](#) displays the status of the persistent volume.

| Phase | Description |
|-----------|-------------------------------------|
| Available | The PV is not yet bound to any PVC. |

| Phase | Description |
|--------------------------|--|
| Bound | The PV is bound to a specific PVC and is in use. |
| Released | The PVC has been deleted, but the PV has not yet been reclaimed. |
| Failed | PV is unusable because it failed to auto-reclaim. |

PvcPhase

[PvcPhase](#) indicates the status of the persistent volume claim.

| Phase | Description |
|-------------------------|---|
| Pending | The PVC is not yet bound to a PV or is waiting for PV allocation. |
| Bound | The PVC is bound to a specific PV and is in use. |
| Lost | The connected PV has been deleted or is no longer valid and cannot be used. |

Select Column

When using the [Select column](#) option, you can select the desired data to view the list.

Select all Deselect all(16) Initialize(default)

| | |
|--|---|
| ☐ PersistentVolume | ☐ PersistentVolumeClaim |
| ☐ PvUid | ☐ PvcUid |
| ☒ PvName | ☒ PvcVolumeName |
| ☒ PvPhase | ☒ PvcPhase |
| ☒ PvType | ☒ pvcNamespace |
| ☒ Path | ☒ PvcStorageClass... |
| ☒ PvReclaimPolicy | ☒ pvcAccessModes |
| ☒ PvStorageClassN... | ☒ PvcRequestedStor... |
| ☒ PvVolumeMode | |
| ☒ PvCapacity | |
| ☒ PvMessage | |
| ☒ PvReason | |

Apply

- The selected column list is kept even if you re-enter the menu.
- If you select [Reset](#), the predefined default key metric columns are selected. In this case, the column selection history is deleted.
- After selecting a column, you can click [Apply](#) at the lower right to apply the selected values.

! You can bulk select or unselect [PersistentVolume](#) or [PersistentVolumeClaim](#) column group.

Kube-apiserver Dashboard

Select Home > cluster project (**CP**) > **Control Plane** > **kube-apiserver Dashboard**

ⓘ Requirements

- The WhaTap Kubernetes agent 1.5.6 or later is required.
- If it is a Kubernetes cluster project (**CP**), you can view the corresponding menu.

The kube-apiserver is a component that handles the requests inside and outside the **Control Plane**. After collecting the status and performance metric of kube-apiserver through the **kube-apiserver Dashboard** menu, you can easily monitor the availability and performance of the cluster by tracing the kube-apiserver's response time, throughput, request failure, and such.

Basic screen guide



The **kube-apiserver Dashboard** menu reflects key metrics of kube-apiserver for the last hour at 1-minute intervals. You can see the following metrics. For more information about metrics, see the [Metric guide](#) below.

- To monitor the cluster's load and usage, the status and trends for request counts by method, resource, and response code.
- The number of API requests being processed is provided to help you understand the cluster load.
- It provides the number of created OS threads and current goroutines.

Enabling the kube-apiserver monitoring

When installing the WhaTap Kubernetes agent, modify the **yaml** file downloaded according to the following instructions to enable the kube-apiserver monitoring.

1. Add the agent container specifications for kube-apiserver monitoring under the spec of the `whatap-master-agent` deployment.

```
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"]
```

See the following example:

```
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. Add the following to the env settings in the `whatap-master-agent` container specifications of the `whatap-master-agent` deployment.

```
env:
  - name: collect_control_plane_monitoring_enabled
    value: "true"
```

See the following example:

```
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. Add the following roles to the `whatap` ClusterRole to collect the control plane monitoring performance metrics.

```
rules:
  - nonResourceURLs: ["/metrics"]
    verbs: ["*"]
```

See the following example:

```
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: ["*"]
---
```

! You can also enable the kube-apiserver monitoring via the `kubectl apply` or `kubectl edit` command.

kube-apiserver metric

| Classification | Metric | Description |
|-------------------------------|--|---|
| API GET request | GET requests per minute | Number of GET requests per minute of kube-apiserver |
| | Trend of GET request counts per second | Trend of GET request counts per second of kube-apiserver |
| | Trend of GET request counts per minute for each resource and subresource | Trend of the GET request counts per minute by resource and subresource incoming to kube-apiserver |
| API POST request | POST requests per minute | Number of POST requests per minute of kube-apiserver |
| | Trend of POST request counts per second | Trend of POST request counts per second of kube-apiserver |
| | Trend of POST request counts per minute for each resource and subresource | Trend of the POST request counts per minute by resource and subresource incoming to kube-apiserver |
| Requests by API response code | [2xx] Request count per minute | The number of requests per minute with the response codes between 200 and 300 among kube-apiserver requests. |
| | [2xx] Trend of request counts per minute for each response code | Response code-based request counts per minute with the response codes between 200 and 300 among kube-apiserver requests. |
| | [2xx] Trend of request counts per minute for each resource and subresource | Trend of request counts per minute grouped by resource and subresource with the response codes between 200 and 300 among kube-apiserver requests. |
| | [3xx] Request count per minute | The number of requests per minute with the response codes between 300 and 400 among kube-apiserver requests. |

| Classification | Metric | Description |
|--------------------------------------|---|---|
| | [3xx] Trend of request counts per minute for each response code | Response code-based request counts per minute with the response codes between 300 and 400 among kube-apiserver requests. |
| | [3xx] Trend of request counts per minute for each resource and subresource | Trend of request counts per minute grouped by resource and subresource with the response codes between 300 and 400 among kube-apiserver requests. |
| | [400~] Request count per minute | The number of requests per minute with the response codes above 400 among kube-apiserver requests. |
| | [400~] Trend of request counts per minute for each response code | Response code-based request counts per minute with the response codes below 400 among kube-apiserver requests. |
| | [400~] Trend of request counts per minute for each resource and subresource | Trend of request counts per minute grouped by resource and subresource with the response codes above 400 among kube-apiserver requests. |
| Go Metrics | Number of goroutines | Number of current goroutines |
| | Number of threads | Current OS thread count |
| Number of active requests (1 second) | Number of requests run in the last 1 second | Number of requests run within 1 second in kube-apiserver during data collection |

Kube-apiserver Metrics Search

Select Home > cluster project (CP) > **Control Plane** > **kube-apiserver Metrics Search**

 Requirements

- The WhaTap Kubernetes agent 1.5.6 or later is required.
- If it is a Kubernetes cluster project (CP), you can view the corresponding menu.

The kube-apiserver is a component that handles the requests inside and outside the **Control Plane**. You can see the original data consistency for the metrics collected from kube-apiserver through the **kube-apiserver Metrics Search** menu.

Basic screen guide

CP koogle-dev-project kube-apiserver Metrics Search

Time

Max count

Category

Filter

|| LIVE 17:54:28 5Min

100

kube_apiserver_request_duration_seconds_sum

+ There is no filter applied.

Download

| time | component | dryRun | group | resource | scope | subresource | verb | vers |
|----------------|-----------|--------|-------|------------|-----------|-------------|-------|------|
| 03/25 17:50:00 | | | | | | /healthz | GET | |
| 03/25 17:50:00 | | | | | | /livez | GET | |
| 03/25 17:50:00 | | | | | | /readyz | GET | |
| 03/25 17:50:00 | | | | | | openapi/v3 | GET | |
| 03/25 17:50:00 | | | | | | openapi/v3/ | GET | |
| 03/25 17:50:00 | apiserver | | | | | api | GET | |
| 03/25 17:50:00 | apiserver | | | | | apis | GET | |
| 03/25 17:50:00 | apiserver | | | configmaps | cluster | | LIST | v1 |
| 03/25 17:50:00 | apiserver | | | configmaps | namespace | | LIST | v1 |
| 03/25 17:50:00 | apiserver | | | configmaps | namespace | | WATCH | v1 |
| 03/25 17:50:00 | apiserver | | | configmaps | resource | | GET | v1 |
| 03/25 17:50:00 | apiserver | | | configmaps | resource | | PUT | v1 |

The **kube-apiserver Metrics Search** menu provides the feature to search for monitoring source data collected for the kube-apiserver by **Category**. Data collected for the last 5 minutes is reflected at an interval of 1 minute.

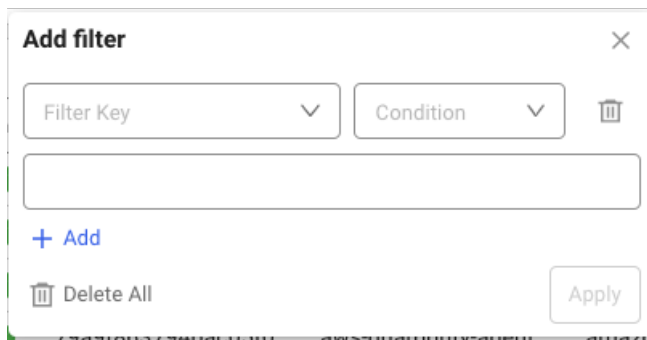
- You can view the data from a specific point in the past by using the [Time selector](#) on the upper left. To view the past data, select desired options and then click the  **Search** icon.

- You can set **Max count** for view.
- You can filter metrics to process the original data as needed.
- Select the **Download** icon at the upper right corner and then download desired data as a **CSV** format file.

Filtering

By using the **Filter** option, you can select a desired for view. Through the **and** condition, multiple filters can be applied.

1. If you select **+ Add icon** in the **Filter** field, the **Add filter** window appears as follows:



- Select **Filter Key** and **Condition**.
 - Select a value that meets the conditions.
2. Configure a filter and then select **Apply** to apply the filter.
 3. To add multiple filters, select **+ Add icon** in the **Filter** field or **Add filter** window.

Enabling the kube-apiserver monitoring

When installing the WhatTap Kubernetes agent, modify the **yaml** file downloaded according to the following instructions to enable the kube-apiserver monitoring.

1. Add the agent container specifications for kube-apiserver monitoring under the spec of the **whatap-master-agent** deployment.

```
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"]
```

See the following example:

```
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. Add the following to the env settings in the `whatap-master-agent` container specifications of the `whatap-master-agent` deployment.

```
env:
  - name: collect_control_plane_monitoring_enabled
    value: "true"
```

See the following example:

```
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. Add the following roles to the `whatap` ClusterRole to collect the control plane monitoring performance metrics.

```

rules:
  - nonResourceURLs: ["/metrics"]

```

```
verbs: ["*"]
```

See the following example:

```
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: ["*"]
---
```

ⓘ You can also enable the kube-apiserver monitoring via the `kubectl apply` or `kubectl edit` command.

kube-apiserver metrics

| Classification | Metric | Description |
|-------------------------------|--|---|
| API GET request | GET requests per minute | Number of GET requests per minute of kube-apiserver |
| | Trend of GET request counts per second | Trend of GET request counts per second of kube-apiserver |
| | Trend of GET request counts per minute for each resource and subresource | Trend of the GET request counts per minute by resource and subresource incoming to kube-apiserver |
| API POST request | POST requests per minute | Number of POST requests per minute of kube-apiserver |
| | Trend of POST request counts per second | Trend of POST request counts per second of kube-apiserver |
| | Trend of POST request counts per minute for each resource and subresource | Trend of the POST request counts per minute by resource and subresource incoming to kube-apiserver |
| Requests by API response code | [2xx] Request count per minute | The number of requests per minute with the response codes between 200 and 300 among kube-apiserver requests. |
| | [2xx] Trend of request counts per minute for each response code | Response code-based request counts per minute with the response codes between 200 and 300 among kube-apiserver requests. |
| | [2xx] Trend of request counts per minute for each resource and subresource | Trend of request counts per minute grouped by resource and subresource with the response codes between 200 and 300 among kube-apiserver requests. |
| | [3xx] Request count per minute | The number of requests per minute with the response codes between 300 and 400 among kube-apiserver requests. |

| Classification | Metric | Description |
|--------------------------------------|---|---|
| | [3xx] Trend of request counts per minute for each response code | Response code-based request counts per minute with the response codes between 300 and 400 among kube-apiserver requests. |
| | [3xx] Trend of request counts per minute for each resource and subresource | Trend of request counts per minute grouped by resource and subresource with the response codes between 300 and 400 among kube-apiserver requests. |
| | [400~] Request count per minute | The number of requests per minute with the response codes above 400 among kube-apiserver requests. |
| | [400~] Trend of request counts per minute for each response code | Response code-based request counts per minute with the response codes below 400 among kube-apiserver requests. |
| | [400~] Trend of request counts per minute for each resource and subresource | Trend of request counts per minute grouped by resource and subresource with the response codes above 400 among kube-apiserver requests. |
| Go Metrics | Number of goroutines | Number of current goroutines |
| | Number of threads | Current OS thread count |
| Number of active requests (1 second) | Number of requests run in the last 1 second | Number of requests run within 1 second in kube-apiserver during data collection |

Etcd Dashboard

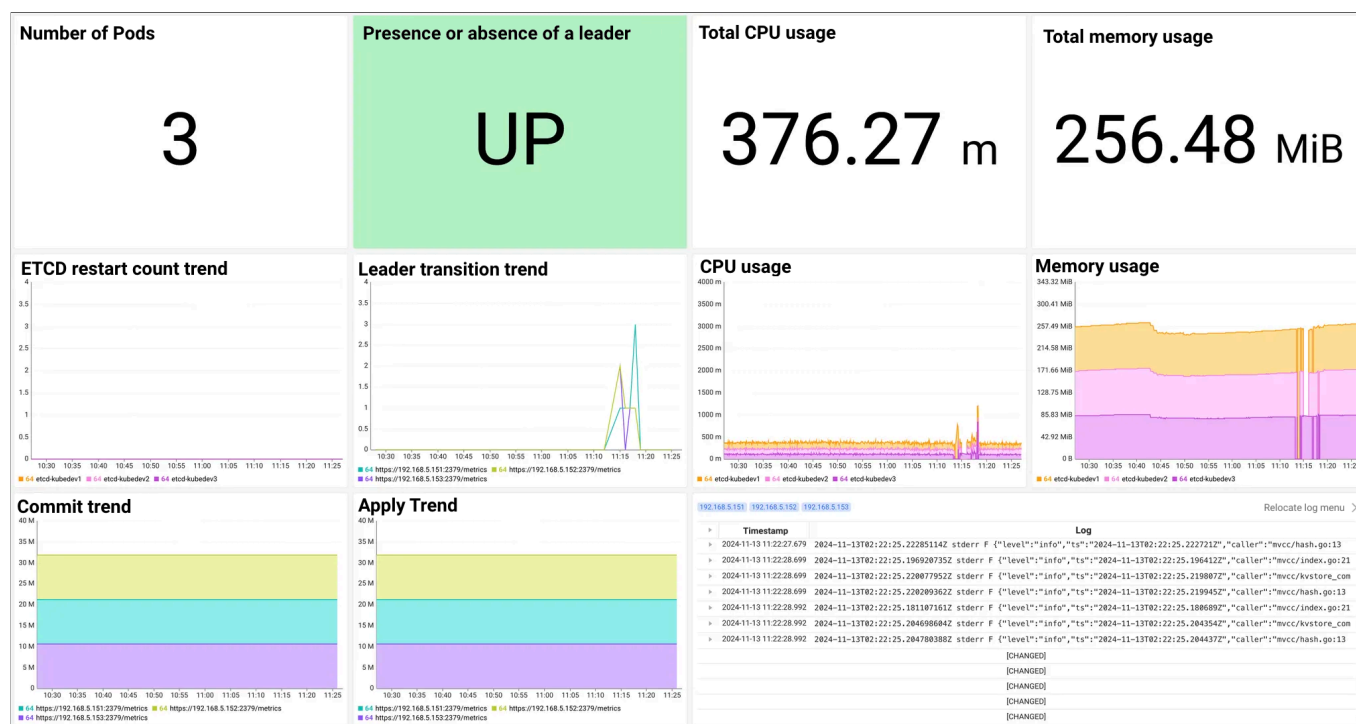
Select Home > cluster project (**CP**) > **Control Plane** > **etcd Dashboard**

Requirements

- The WhaTap Kubernetes agent 1.7.16 or later is required.
- If it is a Kubernetes cluster project (**CP**), you can view the corresponding menu.

etcd is a core component in the Kubernetes environment that stores the data related to the cluster status. WhaTap provides **etcd Dashboard** to monitor the performance and status of etcd. This increases the stability of the cluster and enables a quick response when a problem occurs.

Basic screen guide



etcd Dashboard provides key etcd metrics collected over the last one hour at an interval of 1 minute. The key

metrics you can see on the dashboard are as follows. For the metrics details, see the [metric guide](#) at the bottom.

- The resources used by etcd, number of Pod restarts, and log details are provided.
- The commits and number of applies in etcd are provided.
- It provides the existence status of etcd leaders and the history of leader changes.

Activation of the etcd monitoring

Activate the agent by modifying the downloaded [YAML](#) file and registering the mTLS certificate when installing the WhaTap Kubernetes agent.

Checking the certificate and creating the secret

Checking the etcd certificate

An mTLS certificate is required for etcd. The etcd certificates are typically stored in the [/etc/kubernetes/pki/etcd](#) path on the control plane node. The file owner is the root account and the required certificates are as follows:

- [ca.crt](#)
- [server.crt](#)
- [server.key](#)

On the machine where the `kubectl` command is executed, check if the certificate exists using the following command.

```
sudo ls /etc/kubernetes/pki/etcd
```

Creating the secret

If a certificate exists, create a secret with the following command.

```
sudo kubectl create secret generic whatap-etcd-cert -n whatap-monitoring --from-file=ca.crt=/etc/kubernetes/pki/etcd/ca.crt  
--from-file=server.crt=/etc/kubernetes/pki/etcd/server.crt --from-file=server.key=/etc/kubernetes/pki/etcd/server.key
```

When a secret is created successfully, the following message appears.

```
secret/whatap-etcd-cert created
```

In case the root account is unavailable

If you cannot use the `kubectl` command from the root account, see the following instructions.

1. After copying the certificate, go to the accessible path.
2. Change the owner of the duplicated certificate file to the current owner.

In case there is no namespace

If you create a secret before applying the Kubernetes agent YAML, the secret creation may fail because the `whatap-monitoring` namespace does not exist. After creating the namespace, create a secret again with the following command.

```
kubectl create namespace whatap-monitoring
```

Modifying the WhaTap Kubernetes agent YAML

Modify the WhaTap Kubernetes agent `YAML` file to add container settings and then attach the secret.

Configuring the container

1. Add the etcd-related settings to the args section in the `whatap-control-plane-helper` container of Deployment.

```
-collect_etcd_monitoring_enabled=true
-etcd_hosts=user_etcd_ip, user_etcd_ip
```

See the following example:

Example

```
-etcd_hosts=192.168.1.45,192.168.1.46,192.168.1.47
```

2. Add the following to the environment variables (env) of the `whatap-master-agent` container of Deployment.

```
- name: "collect_etcd_monitoring_enabled"
  value: "true"
```

Connecting the secret

1. Add a secret to the volumes section in `whatap-master-agent` of Deployment.

```
- name: "whatap-etcd-cert"
  secret:
    secretName: "whatap-etcd-cert"
```

2. Mount the secret into the `whatap-control-plane-helper` container.

```
volumeMounts:
- name: "whatap-etcd-cert"
  readOnly: true
  mountPath: "/etc/kubernetes/pki/etcd"
```

Full YAML example

Below is an example of the [YAML](#) file with all settings applied. See the highlighted area.

```
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:
  - command:
    - "/bin/entrypoint.sh"
    env:
    - name: "WHATAP_LICENSE"
      value: "x208gnc8i1kko-z2p19shk7npc4t-z788sgbj3n123a"
    - name: "WHATAP_HOST"
      value: "59.13.101.109"
    - name: "WHATAP_PORT"
      value: "61574"
    - name: "WHATP_MEM_LIMIT"
      valueFrom:
        resourceFieldRef:
          containerName: "whatap-master-agent"
          resource: "limits.memory"
    - name: "collect_control_plane_monitoring_enabled"
      value: "true"
    - name: "collect_etcd_monitoring_enabled"
      value: "true"
    image: "whatap/kube_mon"
    name: "whatap-master-agent"
    ports:
    - containerPort: 6600
    resources:
      limits:
        cpu: "200m"
        memory: "350Mi"
      requests:
        cpu: "100m"
        memory: "300Mi"
    volumeMounts:
    - mountPath: "/bin/entrypoint.sh"
      name: "start-script-volume"
      readOnly: true
      subPath: "entrypoint.sh"
    - mountPath: "/whatap_conf"
      name: "whatap-config-volume"

```

```
- args:
  - "-collect_control_plane_monitoring_enabled=true"
  - "-collect_etcd_monitoring_enabled=true"
  - "-etcd_hosts=192.168.1.45,192.168.1.46,192.168.1.47"
```

command:

```
- "/data/agent/master/whatap_control_plane_helper"
```

image: "whatap/kube_mon"

imagePullPolicy: "Always"

name: "whatap-control-plane-helper"

ports:

```
- containerPort: 9496
```

volumeMounts:

```
- name: "whatap-etcd-cert"
```

readOnly: true

mountPath: "/etc/kubernetes/pki/etcd"

serviceAccount: "whatap"

volumes:

```
- configMap:
```

defaultMode: 448

name: "master-start-script"

```
name: "start-script-volume"
```

```
- emptyDir: {}
```

name: "whatap-config-volume"

```
- name: "whatap-etcd-cert"
```

secret:

secretName: "whatap-etcd-cert"

etcd metrics

| Type | Description |
|--|------------------------------------|
| Number of Pods | Number of the current etcd Pods |
| etcd restart count trend | etcd Pod restart count sum trend |
| Reader change trend | etcd Pod reader change count trend |

| Type | Description |
|--------------------------------------|---|
| Reader exists or not | Whether the etcd Pod as the reader role exists or not |
| CPU usage sum | etcd Pod CPU usage sum |
| CPU usage | etcd Pod CPU usage trend |
| Memory usage sum | etcd Pod memory usage sum |
| Memory usage | etcd Pod memory usage trend |
| Commit trend | Number of etcd change commits |
| Apply trend | Number of applies after a commit |
| etcd log | Logs generated from the etcd Pod |

Kube-scheduler Dashboard

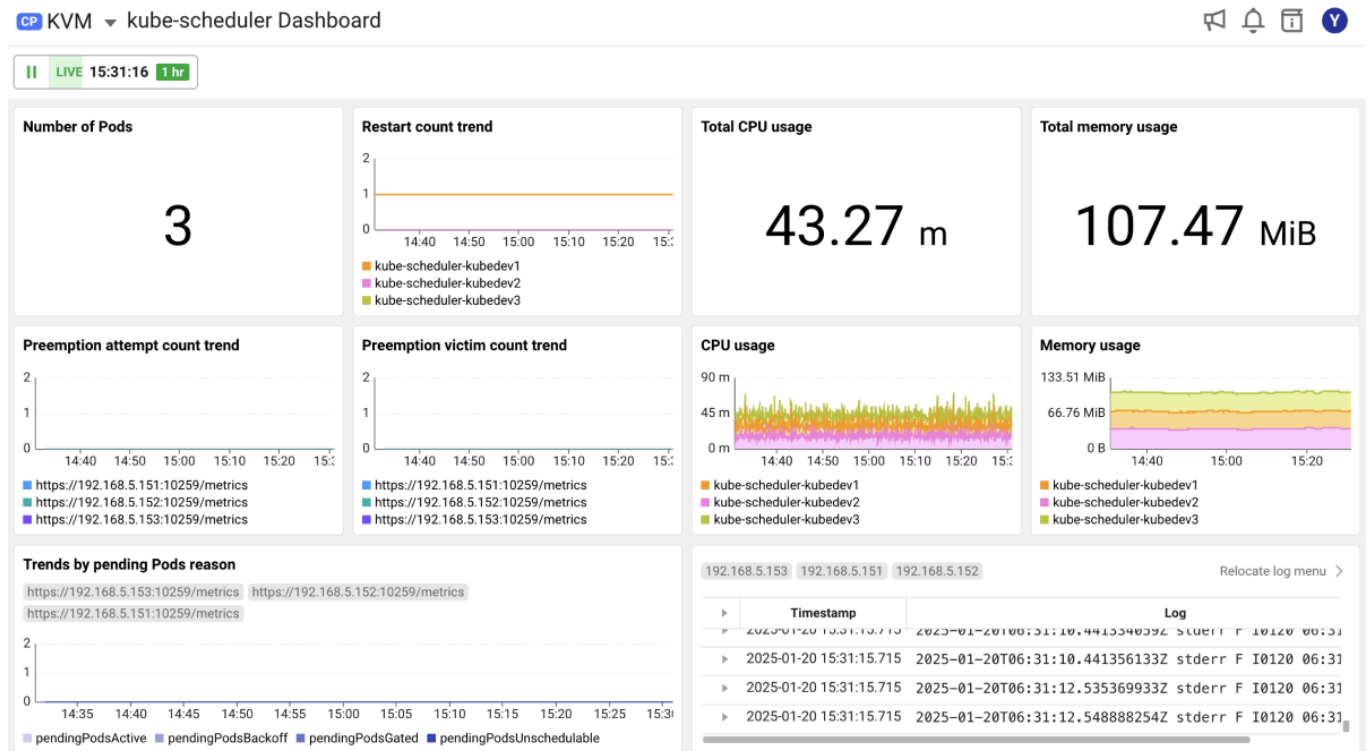
Select Home > cluster project (**CP**) > **Control Plane** > **kube-scheduler Dashboard**

ⓘ Requirements

- The WhaTap Kubernetes agent 1.8.1 or later is required.
- If it is a Kubernetes cluster project (**CP**), you can view the corresponding menu.
- You can use this menu if the cluster can observe the vanilla Kubernetes or kube-scheduler Pods.

kube-scheduler is a major component responsible for scheduling Pods in the Kubernetes environment. WhaTap provides the **kube-scheduler Dashboard** to monitor the performance and scheduling status of kube-scheduler in real time. The **kube-scheduler Dashboard** can help you proactively prevent the cluster from entering an unschedulable state or encountering issues due to scheduler overload in advance.

Basic screen guide



kube-scheduler Dashboard provides visualizations of key metrics collected over the past hour, broken down at 1-minute intervals. The key metrics you can see on the dashboard are as follows. For the metrics details, see the [metric guide](#) at the bottom.

- It provides the resource usage trend, Pod restart count, and detailed logs for `kube-scheduler`.
- It provides the trend for the numbers of preemption attempts by `kube-scheduler` and the number of victims due to preemption.
- It provides the trends for each reason of pending Pods assigned to the pending queue inside the `kube-scheduler` Pod.

Enabling the kube-scheduler monitoring

Enabling requirements

- You must be able to observe the `kube-scheduler` Pod via the command, `kubectl get pods -n kube-system -l component=kube-scheduler`.
- To enable secure communication with `kube-scheduler`, change the `--bind-address` setting to be accessible from outside. Typically `0.0.0.0` is used.

Enabling procedure

! Enabling order

1. Add the `GET` role for `secrets` to RBAC of the whatap service account.
2. Generate `secrets` based on the whatap service account.
3. Apply the `kube-scheduler` monitoring activation option to the `whatap-master-agent` deployment.

1. Role configuration

Add the `secrets` resource to the resources accessible from `rules` of whatap ClusterRole inside the `whatap-monitoring` namespace. After creating the following content as a `yaml` file, execute the `kubectl apply -f <yaml file name>.yaml` command for application.

ClusterRole.yaml

```
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"
  - "secrets"
  - "replicasets"
  - "roles"
  - "rolebindings"
  - "clusterroles"
  - "clusterrolebindings"
  - "jobs"
  - "cronjobs"
  - "statefulsets"
  - "serviceaccounts"
  - "configmaps"
  - "storageclasses"
  - "horizontalpodautoscalers"
  - "replicationcontrollers"
  verbs:
  - "get"
  - "list"
  - "watch"
- apiGroups:
  - ""
  resources:
  - "pods/exec"

```

```

verbs:
- "create"
- apiGroups:
- ""
resources:
- "configmaps"
verbs:
- "*"
- nonResourceURLs:
- "/metrics"
verbs:
- "*"

```

2. Creating the secret

After creating the following content as a **yaml** file, execute the `kubectl apply -f yamlfilename.yaml` command for application.

secrets.yaml

```

apiVersion: v1
kind: Secret
type: kubernetes.io/service-account-token
metadata:
  name: whatap-scheduler-monitoring-token
  namespace: whatap-monitoring
  annotations:
    kubernetes.io/service-account.name: "whatap"

```

3. Deployment modification

After modifying the `whatap-master-agent` deployment content, execute the `kubectl apply -f <yaml file name>.yaml` command for application.

- **Addition of environment variables**

Add the following to the `env` section of the `whatap-master-agent` container under `containers` .

```

env:

```

```
- name: "collect_kube_scheduler_monitoring_enabled"
  value: "true"
```

- **Adding command arguments**

Add the following to the `args` section of the `whatap-control-plane-helper` container under `containers` .

```
- args:
  - "-collect_kube_scheduler_monitoring_enabled=true"
```

See the following example for adding environment variables and command arguments.

```
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:
      priorityClassName: high-priority
      containers:
        - command:
            - "/bin/entrypoint.sh"
          env:
            - name: "WHATAP_LICENSE"
              value: "c1024123i1kko-241245127abdbe-612632344235"
            - name: "WHATAP_HOST"
              value: "192.168.1.0"
```

```

- name: "WHATP_MEM_LIMIT"
  valueFrom:
    resourceFieldRef:
      containerName: "whatap-master-agent"
      resource: "limits.memory"
- name: "collect_kube_scheduler_monitoring_enabled"
  value: "true"
image: "whatap/kube_mon"
name: "whatap-master-agent"
ports:
- containerPort: 6600
resources:
  limits:
    cpu: "200m"
    memory: "350Mi"
  requests:
    cpu: "100m"
    memory: "300Mi"
volumeMounts:
- mountPath: "/bin/entrypoint.sh"
  name: "start-script-volume"
  readOnly: true
  subPath: "entrypoint.sh"
- mountPath: "/whatap_conf"
  name: "whatap-config-volume"
- args:
  - "-collect_kube_scheduler_monitoring_enabled=true"
command:
- "/data/agent/master/whatap_control_plane_helper"
image: "whatap/kube_mon"
imagePullPolicy: "Always"
name: "whatap-control-plane-helper"
ports:
- containerPort: 9496
serviceAccount: "whatap"
volumes:
- configMap:
    defaultMode: 448
    name: "master-start-script"

```



```

name: "start-script-volume"
- emptyDir: {}
name: "whatap-config-volume"

```

kube-scheduler metrics

| Type | Description |
|---|--|
| Number of Pods | Number of kube-scheduler Pods running |
| kube-scheduler restart count trend | Trend of sums of restarts of the kube-scheduler Pod |
| CPU usage sum | Total CPU usage of the kube-scheduler Pod |
| CPU usage | CPU usage trend of the kube-scheduler Pod |
| Memory usage sum | Total memory usage of the kube-scheduler Pod |
| Memory usage | Memory usage trend of the kube-scheduler Pod |
| kube-scheduler log | Log generated from the kube-scheduler Pod |
| Trend of numbers of preemption attempts | Trend of numbers of preemption attempts for Pod scheduling |
| Trend of numbers of preemption victims | Trend of numbers of Pods evicted due to preemption |
| Trend by pending Pod reason | Trend by reason of waiting Pods in pending state without being scheduled |

Object Manifest Information

Home > Project Select > **Analysis** > **Object Manifest**

⚠ Requirements

The WhaTap Kubernetes agent 1.5.5 or later is required. If the version is less than 1.5.5, the [master meta info](#) screen appears.

The Kubernetes Master Agent automatically collects manifests for objects such as Pods, services, and ReplicaSets in clusters once a day. The data is stored by date, allowing users to easily check the cluster status as of a specific date. The collected data can be compared by date, making it easy to identify changes in clusters over time.

Additionally, you can freely set the collection time zone by specifying the container environment variables before deployment of the Kubernetes Master Agent. For more information about the configuration, see the guide to [Container environment variables](#).

Container environment variables

- **store_k8s_object_manifest_enabled**

Default `true`

The object manifest collection feature is enabled. The default value is `true`, the collection feature is enabled without any further settings.

- **store_k8s_object_manifest_timezone**

Default `UTC`

Set the time zone for collection of object manifests. If there is no separate setting, collection starts at midnight based on `UTC` by default.

- Can be set as a region-based time zone or in GMT offset format.
- Supports a precise time setting in minutes.

For example, a non-standard time zone for a specific region can be specified by setting it to

GMT+0810 .

- **store_k8s_object_manifest_max_fetch_limit**

Default 50

Set the number of objects to request to the cluster at a time. If there is no separate setting, 50 items are requested upon request based on the default value of 50 .

For example, when setting the option 30 in a cluster with a total number of Pods of 90, pods are requested 3 times for 30 pods each.

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
      # Added environment variables
```

```

- 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"

```

Basic screen guide

Object list

When you select a desired object from the object type list, you can see the list of objects. It displays the objects such as [Name](#), [Namespace](#), [Uid](#), and [CreatedTime](#). The following screen is an example when selecting [Deployment](#) from the object type list.

Kubernetes Monitoring Demo(EKS) ▾

Object Manifest Information

Now

Date

2024/01/23

Master

Agent

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 2:26:40 PM

Workload

Cronjob

DaemonSet

Deployment

Job

Pod

ReplicaSet

StatefulSet

Network

Ingress

Service

Config/Storage

ConfigMap

PersistentVolume

PersistentVolumeClaim

StorageClass

Security

Role

ServiceAccount

Cluster

ClusterRoleBinding

ClusterRole

Node

| Name | Namespace | Uid | CreatedTime |
|-------------------------------------|-------------------|--------------------------------------|----------------------|
| 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 |
| coredns | kube-system | 79a69bee-1b62-4fa8-85c3-c4036c7f8d43 | 2023-09-13T05:28:09Z |
| ebs-csi-controller | kube-system | 9877183f-a1f9-41b5-abf1-0e8a718ed45d | 2023-10-30T06:09:38Z |
| apm-java-demo | multitx-sample | e86ead35-d92a-4300-9bce-0390b650cdab | 2023-10-27T07:07:33Z |
| nodejs-sample-v2 | multitx-sample | 153484cd-95f5-4904-9912-8100e67976b3 | 2023-11-10T07:18:58Z |
| php-demo-apache-deployment | multitx-sample | b64ffcd-095d-41c5-81b0-80bcd2c95bbe | 2023-11-10T00:50:07Z |
| php-demo-wordpress-apache-deploy... | multitx-sample | a563989e-9af8-4bdb-9308-ad490ec9de21 | 2023-11-01T00:12:56Z |
| testing-http-sender | multitx-sample | 4fa6b5bd-bf26-4ba9-b709-6fa0be44f2a3 | 2023-10-31T08:10:03Z |
| whatap-virtual-deployment | virtual-main | 09b8c928-dca0-43b6-bdb0-07209c5773ee | 2023-09-14T00:21:58Z |
| whatap-virtual-deployment | virtual | b2b003cf-0394-483a-80ed-af03a032a1db | 2023-09-18T08:22:15Z |
| whatap-master-agent | whatap-monitoring | 4e027d67-06f0-4f01-9f91-33d376e3e42d | 2023-09-14T00:20:32Z |

Object Manifest information details

When you select [Display Detail](#) in the object list, you can see the manifest of the object selected in the [Object Manifest Detail](#) window.

Object Manifest Detail

Kind Deployment Name **aws-load-balancer-controller** Namespace kube-system uid c2b7f68c-751e-48c4-90b9-f8e45aac976a CreatedTime 2024-11-25T05:57:37Z

<

2025/03/18 ~ 2025/03/25

7 day

>

Compare

Now

2025/03/25 09:00

2025/03/25 09:00

2025/03/24 09:00

2025/03/24 09:00

2025/03/23 09:00

2025/03/23 09:00

2025/03/22 09:00

2025/03/22 09:00

2025/03/21 09:00

2025/03/21 09:00

2025/03/20 09:00

2025/03/20 09:00

2025/03/19 09:00

2025/03/19 09:00

2025/03/18 09:00

2025/03/18 09:00

2025/03/25 09:00

Please enter your search term

Q

```

1  metadata:
2    generation: "9"
3    uid: "c2b7f68c-751e-48c4-90b9-f8e45aac976a"
4  managedFields:
5    - apiVersion: "apps/v1"
6      manager: "helm"
7      fieldsV1:
8        f:metadata:
9          f:annotations:
10             f:meta.helm.sh/release-namespace: {}
11             f:meta.helm.sh/release-name: {}
12             .: {}
13          f:labels:
14             f:helm.sh/chart: {}
15             f:app.kubernetes.io/instance: {}
16             f:app.kubernetes.io/name: {}
17             f:app.kubernetes.io/version: {}
18             .: {}
19             f:app.kubernetes.io/managed-by: {}
20        f:spec:
21          f:template:
22            f:metadata:
23              f:annotations:
24                .: {}
25                f:prometheus.io/scrape: {}
26                f:prometheus.io/port: {}
27              f:labels:
28                f:app.kubernetes.io/instance: {}
29                f:app.kubernetes.io/name: {}
30                .: {}
31            f:spec:
32              f:securityContext:
33                f:fsGroup: {}
34                .: {}
35              f:restartPolicy: {}
36              f:schedulerName: {}
37              f:containers:
38                k:{"name":"aws-load-balancer-controller"}:
39                  f:livenessProbe:
40                    f:httpGet:
41                      f:port: {}
42                      f:scheme: {}
43                      f:path: {}
44                      .: {}
45                    f:successThreshold: {}
46                    f:initialDelaySeconds: {}
47                    f:failureThreshold: {}

```

metadata

generation

uid

managedFields

resourceVersion

name

namespace

creationTimestamp

annotations

labels

spec

template

replicas

selector

revisionHistoryLimit

strategy

progressDeadlineSeconds

status

replicas

readyReplicas

conditions

updatedReplicas

observedGeneration

availableReplicas

List of Object Manifest saving times

- You can see the list of the times saved for object manifests on the left. Object manifests are automatically saved at a specified time per day.

- The object manifest saving time can be changed by modifying the value of the master agent's container environment variable, `store_k8s_object_manifest_timezone` .
- Object manifests are saved at a scheduled time per day, but are also additionally saved when any warning events occur.

! The warning events in which the object manifests are saved indicate Kubernetes event objects of Warning type.

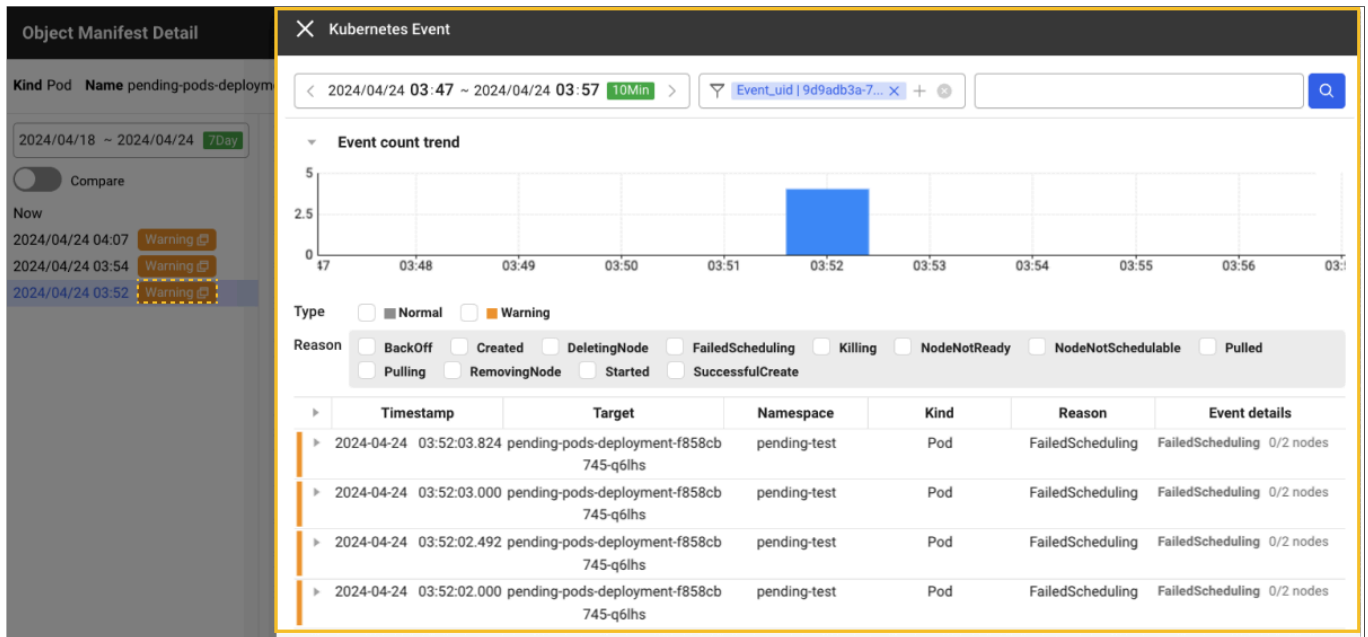
Search and navigate object manifests

The [Object Manifest Detail](#) window provides a search feature to help you find specific keywords in the object manifest. The string entered in the search bar is highlighted in the object manifest. This allows you to quickly find specific settings or fields in the complex manifest structure.

Additionally, the [Object Manifest Detail](#) window provides a navigation pane on the right that provides a summary of the entire structure of the manifests in tree format. This allows you to see key items without having to scroll when working with large manifest documents.

Linking the Kubernetes events

When you click [Warning](#)  in the [Object manifest detail](#) window, Kubernetes events appear as follows:



! The Kubernetes event linking is supported by WhaTap Master Agent 1.6.1 or later.

Comparison of object manifests

When ☒ Compare is enabled, you can compare two saved object manifests by selecting in the saved time list. Any changes are highlighted.

Object Manifest Detail

Kind Pod Name pending-pods-deployment-f858cb745-q6lhs Namespace pending-test uid 2b839d92-8002-4c3c-aafd-021ab5f7b888 CreatedTime 2024-04-23T18:52:02Z

2024/04/18 ~ 2024/04/24 7Day

Compare

Now

2024/04/24 04:07 Warning

2024/04/24 03:54 Warning

2024/04/24 03:52 Warning

2024/04/24 03:52 Warning

```

60   operation: "Update"
61   fieldType: "FieldsV1"
62-  resourceVersion: "46210568"
63   name: "pending-pods-deployment-f858cb745-q6lhs"
64   namespace: "pending-test"
65   creationTimestamp: "2024-04-23T18:52:02Z"
66   generateName: "pending-pods-deployment-f858cb745-"
67   labels:
68     app: "pending-pods"
69     pod-template-hash: "f858cb745"
70   ownerReferences:
71     - uid: "08e4b783-a08b-4a4b-996b-6a419f89c76d"
72       controller: "true"
73       apiVersion: "apps/v1"
74       kind: "ReplicaSet"
75       name: "pending-pods-deployment-f858cb745"
76       blockOwnerDeletion: "true"
77   apiVersion: "v1"
78   kind: "Pod"
79   spec:
80     dnsPolicy: "ClusterFirst"
81     terminationGracePeriodSeconds: "30"
82     enableServiceLinks: "true"
83     serviceAccountName: "default"
84     volumes:
85       - projected:
86         sources:
87           - serviceAccountToken:

```

2024/04/24 04:07 Warning

```

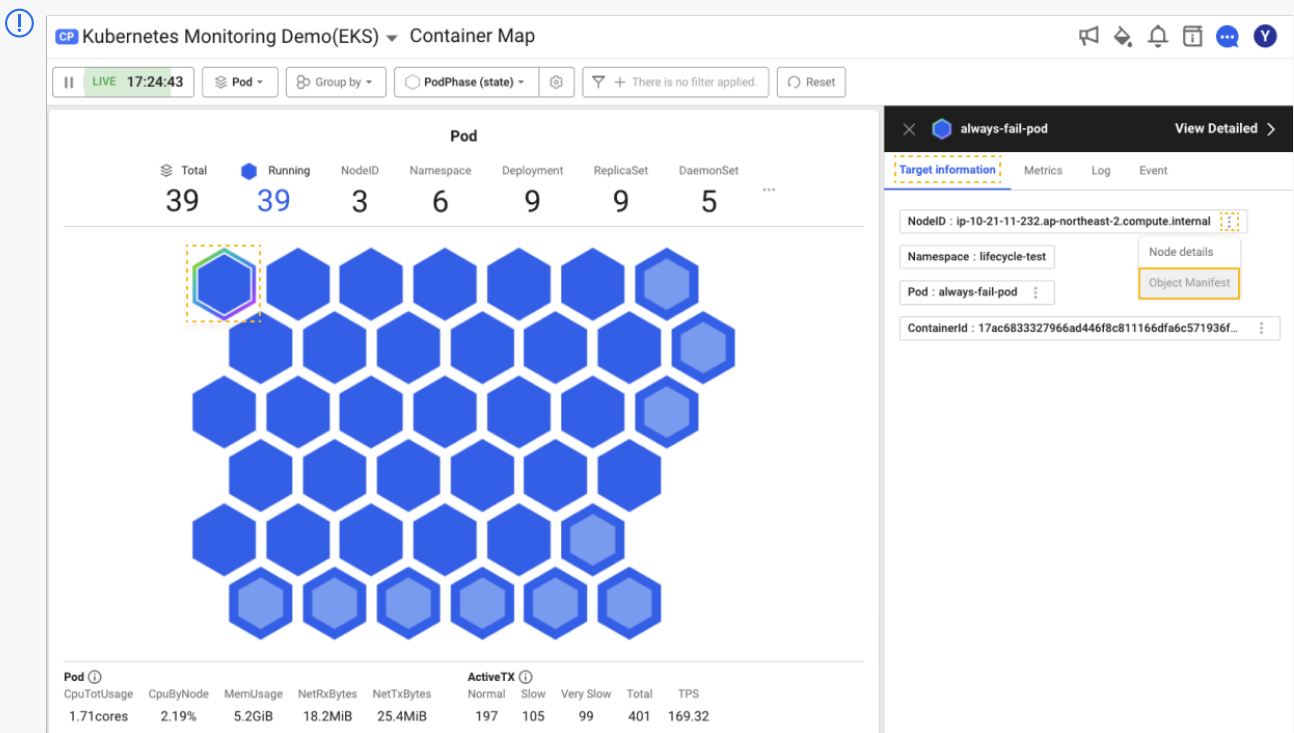
86+  f:status: {}
87+  .. {}
88+  f:reason: {}
89+  f:startTime: {}
90+  f:containerStatuses: {}
91+  f:hostIP: {}
92+  subresource: "status"
93+  time: "2024-04-23T19:07:34Z"
94   operation: "Update"
95   fieldType: "FieldsV1"
96+  resourceVersion: "46213105"
97   name: "pending-pods-deployment-f858cb745-q6lhs"
98   namespace: "pending-test"
99   creationTimestamp: "2024-04-23T18:52:02Z"
100  generateName: "pending-pods-deployment-f858cb745-"
101  labels:
102    app: "pending-pods"
103    pod-template-hash: "f858cb745"
104  ownerReferences:
105    - uid: "08e4b783-a08b-4a4b-996b-6a419f89c76d"
106      controller: "true"
107      apiVersion: "apps/v1"
108      kind: "ReplicaSet"
109      name: "pending-pods-deployment-f858cb745"
110      blockOwnerDeletion: "true"
111  apiVersion: "v1"
112  kind: "Pod"
113  spec:
114    dnsPolicy: "ClusterFirst"
115+  nodeName: "ip-10-21-120-65.ap-northeast-2.compute.internal"
116    terminationGracePeriodSeconds: "30"
117    enableServiceLinks: "true"
118    serviceAccountName: "default"
119    volumes:
120      - projected:
121        sources:
122          - serviceAccountToken:

```

! Viewing object manifests on the container map

You can see the [object manifests](#) in the [container map](#).

444

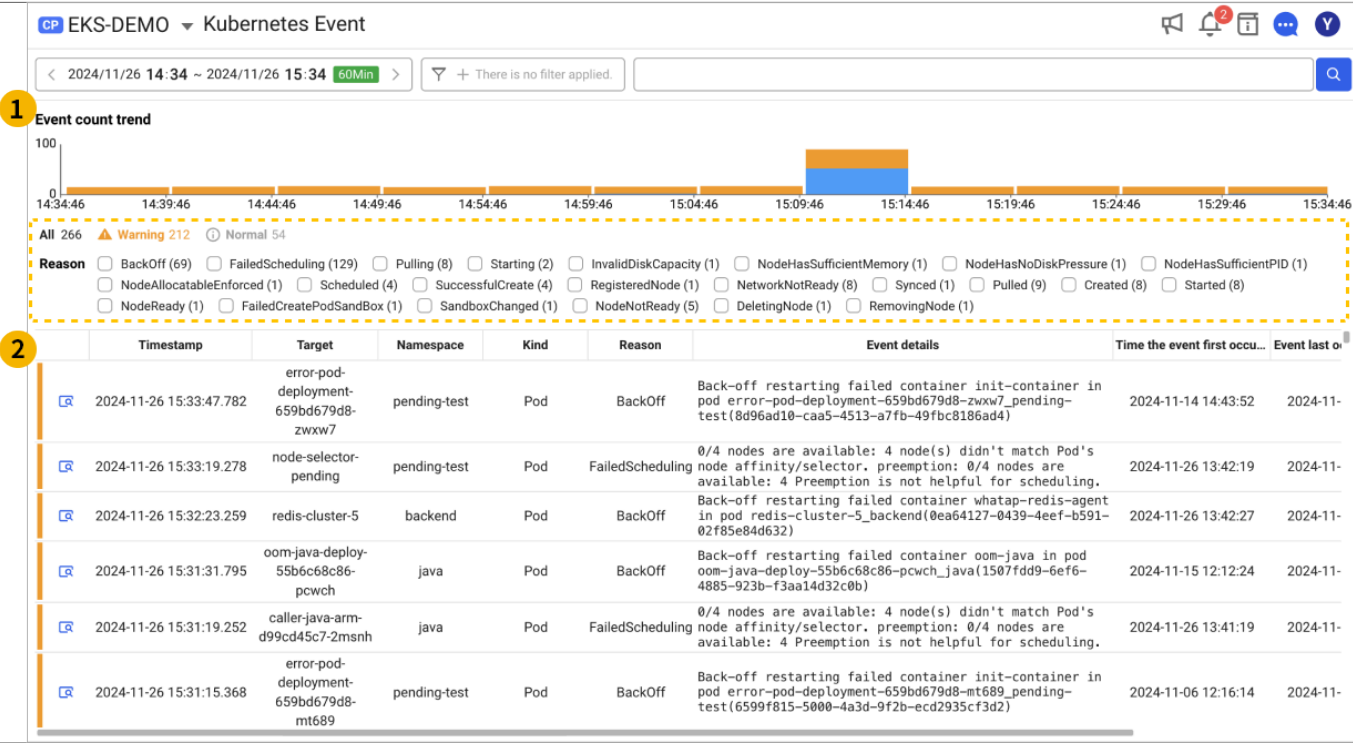


- Select a desired block.
- In the summary window, select the [Target Info](#) tab to view the Pod or container data.
- On the right of the [Pod](#), [Deployment](#), [ReplicaSet](#), and [Node](#) tags, click [More](#) and then select [Object Manifest](#).

Kubernetes Event

Home > Select Project > Analysis > Kubernetes Event

The WhaTap **Kubernetes Event** menu allows you to perform various queries and searches for Kubernetes events without using commands. You can check the cluster events and also view past events through the time selector.



- You can find the desired events through the filtering options and search bar.
- You can quickly filter by event type, such as Normal or Warning, and Reason.

ⓘ

• For more information about Kubernetes events, see [the following](#).

Event count trend

The **1** **Event count trend** presented in the bar chart format allows you to visually see the change in the number of events that occurred within the selected time range.

You can select a desired time zone to view and see the events occurred in the zone. This allows you to quickly identify the time zone when events were concentrated and analyze its details.

Event list

You can see the list of Kubernetes events that occurred during a specific time period in the **2** area. WhaTap stores the event data up to 30 days, allowing you to view and analyze past events.

You can also quickly filter desired events by selecting the event type (**Warning**, **Normal**), or **Reason** at the top of the list.

| Item | Description |
|-----------------------------|---|
| Timestamp | It is the time when the event occurred. |
| Target | It is the target on which the event occurred. If Kind is Pod, it indicates the Pod name. If Kind is Node, it indicates the node name. |
| Namespace | It is the namespace in which the event occurred. |
| Kind | It is the type of object on which the event occurred. |
| Reason | It is the reason why the event occurred. |
| First event occurrence time | It is the time when the event was first observed. |
| Event occurrence count | It is the number of events that occurred continuously over a period of time. |

| Item | Description |
|-------------------------------|---------------------------------|
| Event content | It is a specific event details. |

Event details

You can check the **event details** by selecting the [details](#) icon on the left of the event to view in the event list [🔍](#).

CP EKS-DEMO ▾ Kubernetes Event

< 2024/11/26 14:37 ~ 2024/11/26 15:37 60Min >

Reason | BackOff X +

🔍

Event count trend

All 272 ⚠ Warning 217 🟢 Normal 55

Reason

☐ FailedScheduling (133)
☒ BackOff (70)
☐ Pulling (9)
☐ Starting (2)
☐ InvalidDiskCapacity (1)
☐ NodeHasSufficientMemory (1)
☐ NodeHasNoDiskPressure (1)
☐ NodeHasSufficientPID (1)
☐ NodeAllocatableEnforced (1)
☐ Scheduled (4)
☐ SuccessfulCreate (4)
☐ RegisteredNode (1)
☐ NetworkNotReady (8)
☐ Synced (1)
☐ Pulled (9)
☐ Created (8)
☐ Started (8)
☐ NodeReady (1)
☐ FailedCreatePodSandbox (1)
☐ SandboxChanged (1)
☐ NodeNotReady (5)
☐ DeletingNode (1)
☐ RemovingNode (1)

| | Timestamp | Target | Namespace | Kind | Reason |
|---|-------------------------|---------------------------------------|--------------|------|---------|
| 🔍 | 2024-11-26 15:37:25.782 | redis-55d465bc96-z4jf9 | backend | Pod | BackOff |
| 🔍 | 2024-11-26 15:37:19.262 | redis-cluster-5 | backend | Pod | BackOff |
| 🔍 | 2024-11-26 15:37:12.767 | redis-55d465bc96-z4jf9 | backend | Pod | BackOff |
| 🔍 | 2024-11-26 15:36:30.795 | oom-java-deploy-55b6c68c86-pcwch | java | Pod | BackOff |
| 🔍 | 2024-11-26 15:36:15.367 | error-pod-deployment-659bd679d8-mt689 | pending-test | Pod | BackOff |

BackOff Pod: redis-55d465bc96-z4jf9

Suggestions

Adjacent Log

Event timeline

Pod summary analysis

Object Manifest

Data(detailed) collection time : 2024/11/26 15:37:25

BackOff:Back-off restarting failed container redis in pod redis-55d465bc96-z4jf9_backend(cc03d7a6-e1a1-461b-8bde-e1835fbc353d)

Check the container logs.

Pod: redis-55d465bc96-z4jf9

```

2024-11-26T06:37:11.352143735Z stdout F sleeptime=3
2024-11-26T06:37:11.352140923Z stdout F value728
2024-11-26T06:37:11.352138137Z stdout F sleeptime=4
2024-11-26T06:37:11.352135363Z stdout F delete key392
2024-11-26T06:37:11.352132116Z stdout F sleeptime=3
2024-11-26T06:37:11.352127583Z stdout F value576
2024-11-26T06:37:11.352124435Z stdout F sleeptime=5
2024-11-26T06:37:11.352122031Z stdout F value712
2024-11-26T06:37:11.352119623Z stdout F sleeptime=3
2024-11-26T06:37:11.352117075Z stdout F delete key885

```

Check whether the environment variables set in the container are correct.

Container: redis

```

1
2  - name: "NODE_IP"
3    valueFrom:
4      fieldRef:
5        apiVersion: "v1"
6        fieldPath: "status.hostIP"
7  - name: "NODE_NAME"

```

Suggestions

The [Suggestions](#) tab provides insights into the cause of an event and provides directions for tracing the problem based on the event message that occurred.

ⓘ Currently, only the [BackOff](#) and [NodNotReady](#) events of the [Warning](#) type are provided in this feature.

Adjacent logs

You can check the context before and after an event by viewing adjacent logs based upon occurrence of the selected event. You can set the desired adjacent range by using the time selection button.

ⓘ Log query role

- If you have **Log query role**, you can see all adjacent logs.
- If your have no **log query role**, you can view only the adjacent Kubernetes events.

Event timeline

It provides all events that occurred from the time the target of the selected event ([Kind](#)) was created to the present in a timeline format. This allows you to see the target's event history at a glance. This provides a comprehensive view for the resource lifecycle, including creation, normal operation, alert generation, and recovery.

Summary analysis

It provides a summary at a glance so you can see the performance and resource status for the Pod or node where the event occurred.

Pod Startup Analysis

Home > Select Project > **Analysis** > **Pod Startup Analysis**

ⓘ It is supported by WhaTap Node Agent **1.4.2 or later**. If the version is lower, the [Previous menu](#) (formerly **Pod initialization performance**) screen is provided.

For smooth cluster operation, it is required to analyze the performance of Pods until they operate stably and optimize the size of resources allocated to Pods. The **Pod Startup Analysis** menu traces the change for 5 minutes after the Pod starts. You can optimize resource allocation by analyzing the resource usage in the three-level sections from **Pending** state to **Running** state and **Stable Running** state.

Through this, operators can have benefits such as improved resource efficiency, cost savings, improved scalability, and ease of scaling.

Basic screen guide

CP

EKS-DEMO

▼

Pod Startup Analysis

2025/02/24 11:25

Show Pods Created within 30 Days

☐

Name

1

Manage container resources efficiently by analyzing pod initiation performance.

Set appropriate resource requests and limits with analytics data from pod initiation performance suggested by WhaTap. Analysis of the selected pod will take place for 5 minutes after initiation. Understand resource usage that are divided by **Pending > Running > Stable Running** segments so you can use resources efficiently and keep your cluster running stable.

* Pod initiation performance feature uses data from the last 30 days. Therefore, the initiation performance of pods older than 30 days from creation date cannot be analyzed.

View in WhaTap Docs

Collapse Pod Startup Analysis Guide

Pod creation

About 5 minutes

End of analysis

Pending

Running

Stable Running

Avg CpuByLimit (5min)

CpuByLimit

Point when the CPU usage is lower than the average usage for about 5 minutes compared to the CPU limit after Pod creation

2

| De... | startTime | Pod | qosClass | ConRestartCnt | CpuLimit | MemLimit | CpuRequest | MemRequest |
|--------------------------|---------------------|-------------------------|------------|---------------|----------|------------|------------|------------|
| ☐ | 2024/11/05 15:03:51 | argama-5dbb48cbff-... | Burstable | 0 | 250 m | 384.00 MiB | 150 m | 128.00 MiB |
| ☐ | 2025/02/20 18:00:41 | argama-5dbb48cbff-x... | Burstable | 0 | 250 m | 384.00 MiB | 150 m | 128.00 MiB |
| ☐ | 2025/01/08 22:36:32 | aws-load-balancer-co... | BestEffort | 0 | 2000 m | 7.65 GiB | 0 m | 0 B |
| ☐ | 2024/10/29 13:50:51 | aws-load-balancer-co... | BestEffort | 0 | 2000 m | 7.65 GiB | 0 m | 0 B |
| ☐ | 2024/10/29 09:20:14 | aws-node-terminat... | Burstable | 0 | 2000 m | 7.65 GiB | 25 m | 0 B |

Option area at the top

You can use the following options in the top 1 area.

- You can select the start time via the time selector. When you select the start time for query, you can check the data in the Pod list of the 2 area.
- By enabling the Show only Pods created within 30 days toggle, you can view the Pods within 30 days after creation in the Pod list in the 2 area.
- You can search for the desired Pods by entering characters in the Pod search field.

!

• The Pod Startup Analysis menu provides the time selector feature starting from WhaTap Kubernetes agent 1.7.6.

451

- The **Pod Startup Analysis** menu analyzes the Pod startup data for 5 minutes within the last 30 days starting from Pod creation. This means that the startup performance cannot be analyzed for Pods that are older than 30 days after creation.

Pod list area

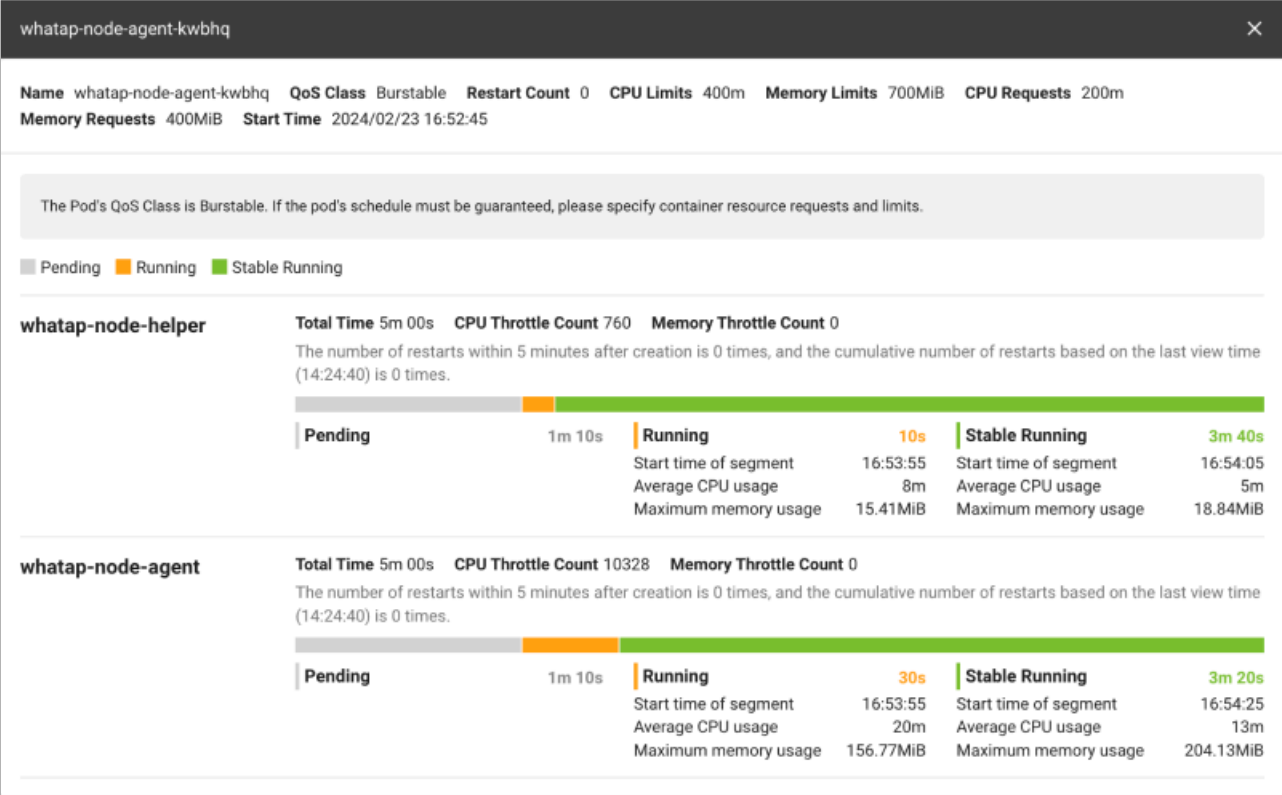
In the **2** area, you can see the Pod list. The header columns of the table where the Pod list is expanded contains the following data.

- **QoS Class**: Quality of Service of the Pod. QoS classes determine the priority of Pods when resources are insufficient. Pods with higher priorities are terminated first.
 - **Guaranteed**: Most available resources in the node are guaranteed and are assigned priority over other Pods. It is terminated last.
 - This is the case when all containers in the Pod have CPU and memory requests.
 - This is the case when all containers in the Pod have CPU and memory limits and their values are the same as the requests.
 - **Burstable**: The Pod allocates some resources when the node runs out of resources.
 - This is the case when the conditions of the Guaranteed class are not met.
 - This is applied when any of the Pod's containers have requests or limits set for CPU and memory.
 - **BestEffort**: It has lower priority than other Pods because there is no resource request or limit. It is terminated first.
 - This is the case when all containers in the Pod have no requests and limits for CPU and memory.
- **Restart Count**: Total number of restarts of containers within the Pod.
- **CPU/Memory Limits**: Sum of CPU/memory limits (`resources.limits`) set to the container within the Pod.
- **CPU/Memory Requests**: Sum of CPU/memory requests (`resources.requests`) set to the container within Pod.
- **Start Time**: Pod generation time (`metadata.startTime`).

- For more information about the QoS Class, see [the following link](#).
- For more information about the Pod and container resource management, see [the following link](#).

Detailed view guide

From the Pod list of the 2 area, select [Display Detail](#) at the utmost left of the item to analyze. You can see the analysis of the startup performance by container in the selected Pod for each zone as follows.



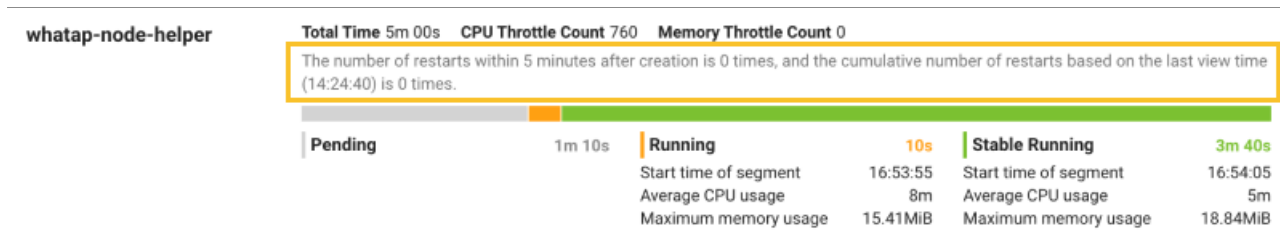
- **Pending**: Section that enters the **Running** state after creation of a Pod.
- **Running**: Section before the Pod passes the **Pending** state and becomes stable.
- **Stable Running**: Section where the Pod has passed the **Running** state and has stabilized.

ⓘ Stable state (Stable Running)

WhaTap calculates the average usage compared to the container's CPU limit for about 5 minutes immediately after Pod creation. The point at which CPU usage becomes less than this average is defined as the stabilization start state.

Analyzing in detail

- If a container is restarted multiple times within 5 minutes, only the information of the container that started first appears. Check the number of restarts for each container. If there are numerous restarts, check whether the container is running normally.



- If the **Pending** section becomes longer, it means that it took a long time for the Pod to enter the **Running** state. Check if there are any factors that may cause the Pod's **Pending** state.

❗ For more information about the Pod's pending state, see [the following link](#).

- If the **Running** section becomes longer, it means that there is a section with high resource usage when the container in the Pod first runs. Check if there is any problem with the container execution process.
- If the overall initial resource usage is high, consider increasing container resource requests and limits.

Pod Startup Analysis

Home > Select Project > [Analysis](#) > [Pod Startup Analysis](#)

ⓘ Below WhaTap node agent 1.4.2

- Instead of the new [Pod initialization analysis](#) menu, the existing menu (**Pod initialization performance**) is provided.
- It can be accessed only in the cluster project (**CP**).

JVM-based applications including the springboot may fail the liveness check at initialization by using a large number of resources. If the liveness check fails over a specific number of times, the Pod is deleted and restart is attempted infinitely, which may cause deployment failure.

If you use the [Pod Startup Analysis](#) function of WhaTap Kubernetes, by analyzing the peak usage and initialization time at Pod startup, you can get help in setting the optimal initial delay (liveness check exemption time).

Basic screen guide

CP

Infrastructure Monitoring (Demo/2023)

Pod Initialization Performance

🔊

🔍

🔔

📅

⋮

Y

1

Running 43

🔄

2

🔍

Pod Name, QoS Class

🔍

| State | Name | QoS class | Start time | Initialization start time | CPU Limit at in... | CPU Request a... | CPU Throttle a... | Memory Limit ... | Memory Requ... | Memory Throt... |
|-------|-------------------------------|------------|----------------------|---------------------------|--------------------|------------------|-------------------|------------------|----------------|-----------------|
| ✓ | whatap-node-agent-ivlhx | Burstable | 2023-11-08T02:28:16Z | 2023-11-08 11:29:30 | 400 | 200 | 31 | 700 M | 400 M | 0 |
| ✓ | whatap-node-agent-hqcg2 | Burstable | 2023-11-02T05:09:16Z | 2023-11-02 14:09:55 | 400 | 200 | 69 | 700 M | 400 M | 0 |
| ✓ | whatap-node-agent-4tbg5 | Burstable | 2023-11-08T02:28:16Z | 2023-11-08 11:29:30 | 400 | 200 | 40 | 700 M | 400 M | 0 |
| ✓ | whatap-master-agent-7c76... | Burstable | 2023-11-08T02:27:39Z | 2023-11-08 11:28:35 | 200 | 100 | 501 | 350 M | 300 M | 0 |
| ✓ | whatap-virtual-deployment-... | Burstable | 2023-11-08T02:26:13Z | | 300 | 50 | 0 | 256 M | 256 M | 0 |
| ✓ | whatap-virtual-deployment-... | Burstable | 2023-11-08T02:26:13Z | | 300 | 50 | 0 | 256 M | 256 M | 0 |
| ✓ | whatap-virtual-deployment-... | Burstable | 2023-11-08T02:26:13Z | | 300 | 50 | 0 | 256 M | 256 M | 0 |
| ✓ | whatap-virtual-deployment-... | Burstable | 2023-11-08T02:26:19Z | | 300 | 50 | 0 | 256 M | 256 M | 0 |
| ✓ | whatap-virtual-deployment-... | Burstable | 2023-11-08T02:26:18Z | | 300 | 50 | 0 | 256 M | 256 M | 0 |
| ✓ | whatap-virtual-deployment-... | Burstable | 2023-11-08T02:26:24Z | 2023-11-08 11:26:45 | 300 | 50 | 23 | 256 M | 256 M | 0 |
| ✓ | whatap-virtual-deployment-... | Burstable | 2023-11-08T02:26:31Z | 2023-11-08 11:26:45 | 300 | 50 | 12 | 256 M | 256 M | 0 |
| ✓ | whatap-virtual-deployment-... | Burstable | 2023-11-08T02:26:24Z | 2023-11-08 11:26:45 | 300 | 50 | 9 | 256 M | 256 M | 0 |
| ✓ | whatap-virtual-deployment-... | Burstable | 2023-11-08T02:26:31Z | 2023-11-08 11:26:45 | 300 | 50 | 31 | 256 M | 256 M | 0 |
| ✓ | whatap-virtual-deployment-... | Burstable | 2023-11-08T02:26:30Z | 2023-11-08 11:26:45 | 300 | 50 | 14 | 256 M | 256 M | 0 |
| ✓ | whatap-virtual-deployment-... | Burstable | 2023-11-08T02:26:31Z | 2023-11-08 11:26:45 | 300 | 50 | 31 | 256 M | 256 M | 0 |
| ✓ | whatap-virtual-deployment-... | Burstable | 2023-11-08T02:26:24Z | 2023-11-08 11:26:45 | 300 | 50 | 12 | 256 M | 256 M | 0 |
| ✓ | whatap-virtual-deployment-... | Burstable | 2023-11-08T02:26:24Z | 2023-11-08 11:26:45 | 300 | 50 | 11 | 256 M | 256 M | 0 |
| ✓ | whatap-virtual-deployment-... | Burstable | 2023-11-08T02:26:31Z | 2023-11-08 11:26:45 | 300 | 50 | 17 | 256 M | 256 M | 0 |
| ✓ | whatap-virtual-deployment-... | Burstable | 2023-11-08T02:26:24Z | 2023-11-08 11:26:45 | 300 | 50 | 20 | 256 M | 256 M | 0 |
| ✓ | testing-http-sender-58d44d... | BestEffort | 2023-11-02T05:21:29Z | 2023-11-02 14:24:40 | 0 | 0 | 0 | 0 | 0 | 0 |

- On the upper left of the **1** area, you can see the Pod counts for each status.
- You can refresh information by selecting the  Refresh icon on the upper right of the **1** area.
- You can select a column by selecting the  Select column icon on the left of the **1** area.
- In the input field on the right of **1** area, you can search for the Pod **Name** or **QoS class**.
- When selecting a desired column in the **2** area, you can sort the list based on the corresponding column.

Column guide

| | | | |
|--|---|--|---|
| Select all Deselect all Initialize(default) × | | | |
| ☒ About pod | ☒ Initialization begins | ☒ Initializing | ☒ Initialization complete |
| ☒ State | ☒ Initialization start time | ☒ Time taken for initialization (seconds) | ☒ Initialization completion time |
| ☒ Name | ☒ CPU Limit at initialization start | ☒ Average CpuByLimit during initialization | ☒ Average CpuByLimit after initialization is complete |
| ☒ QoS class | ☒ CPU Request at initialization start | ☒ Maximum memory during initialization | ☒ Maximum memory after initialization complete |
| ☒ Start time | ☒ CPU Throttle at initialization start | | ☒ After initialization is complete (milliseconds) (up to 5 minutes) |
| | ☒ Memory Limit when initialization starts | | |
| | ☒ Memory Request at initialization start | | |
| | ☒ Memory Throttle at initialization start | | |

The Pod Startup Analysis list can classify columns according to **About pod** and Pod initialization timeline as follows:

About pod

You can see the information for **Name**, **State**, **QoS class**, and **Start time** of the Pod.

- **State:** Indicates the life cycle of the Pod.
 -  Pending: Indicates that a Pod start has been approved inside the Kubernetes cluster, but no container has been created.
 -  Running: Indicates that the Pod is bound to the node, all containers are created, and at least one container is running or starting or restarting.
 -  Succeeded: Indicates that all containers in the Pod have been successfully terminated and have not been restarted.
 -  Failed: Indicates that one or more containers in the Pod failed and were terminated.
 -  Unknown: Indicates that communication with the Pod is not possible for some reasons.
- **QoS class:** The quality of service of the Pod. The QoS class determines the priority of Pods when resources are insufficient. Higher priority Pods are terminated before lower priority Pods.
 - **Guaranteed:** Most available resources in the node are guaranteed and are assigned priority over other

Pods. They are terminated last.

- This is the case when all containers in the Pod have CPU and memory requests.
- This is the case when all containers in the Pod have CPU and memory limits and their values are the same as the requests.
- **Burstable:** The Pod allocates some resources when the node runs out of resources.
 - This is the case when the conditions of the Guaranteed class are not met.
 - This is applied when any of the Pod's containers have requests or limits set for CPU and memory.
- **BestEffort:** It has lower priority than other Pods because there is no resource request or limit. It is terminated first.
 - This is the case when all containers in the Pod have no requests and limits for CPU and memory.
- **Start time:** Indicates the Pod's start time or created time.

Start initialization

Start initialization indicates the time point when the first metric data has been collected.

- **Initialization start:** Indicates the data for the initialization started time. It is the time point of first data collection.
- **CPU limit at start of initialization:** Indicates the CPU limit when initialization has started.
- **CPU request at start of initialization:** Indicates the CPU request when initialization has started.
- **CPU throttle count at start of initialization:** A value that is incremented when the container exceeds the CPU limit at the start of initialization.
- **Memory limit at start of initialization:** Indicates the memory limit set when initialization has started.
- **Memory request at start of initialization:** Indicates the memory request requested when initialization has started.
- **Memory throttle count at start of initialization:** A value that is incremented when the container exceeds the memory limit at the start of initialization.

Initializing

Initializing indicates the previous time point when data with a CpuByLimit value lower than the average value for 5 minutes is generated after initialization.

- **Time spent initializing (in seconds):** Indicates the time taken to initialize the Pod.

Time when the CpuByLimit value lower than the average value has been collected among the data for 5 minutes at the start of initialization - first data collection time

- **Average CpuByLimit during initialization:** Indicates the average value of CpuByLimit during initialization.
- **Maximum memory during initialization:** Indicates the maximum amount of memory used by the container during initialization.

! CpuByLimit (`cpu_per_quota`) indicates the total CPU utilization based on the CPU limit.

Initialization complete

Initialization complete indicates the time point when data with a CpuByLimit value lower than the average value for 5 minutes is generated after initialization.

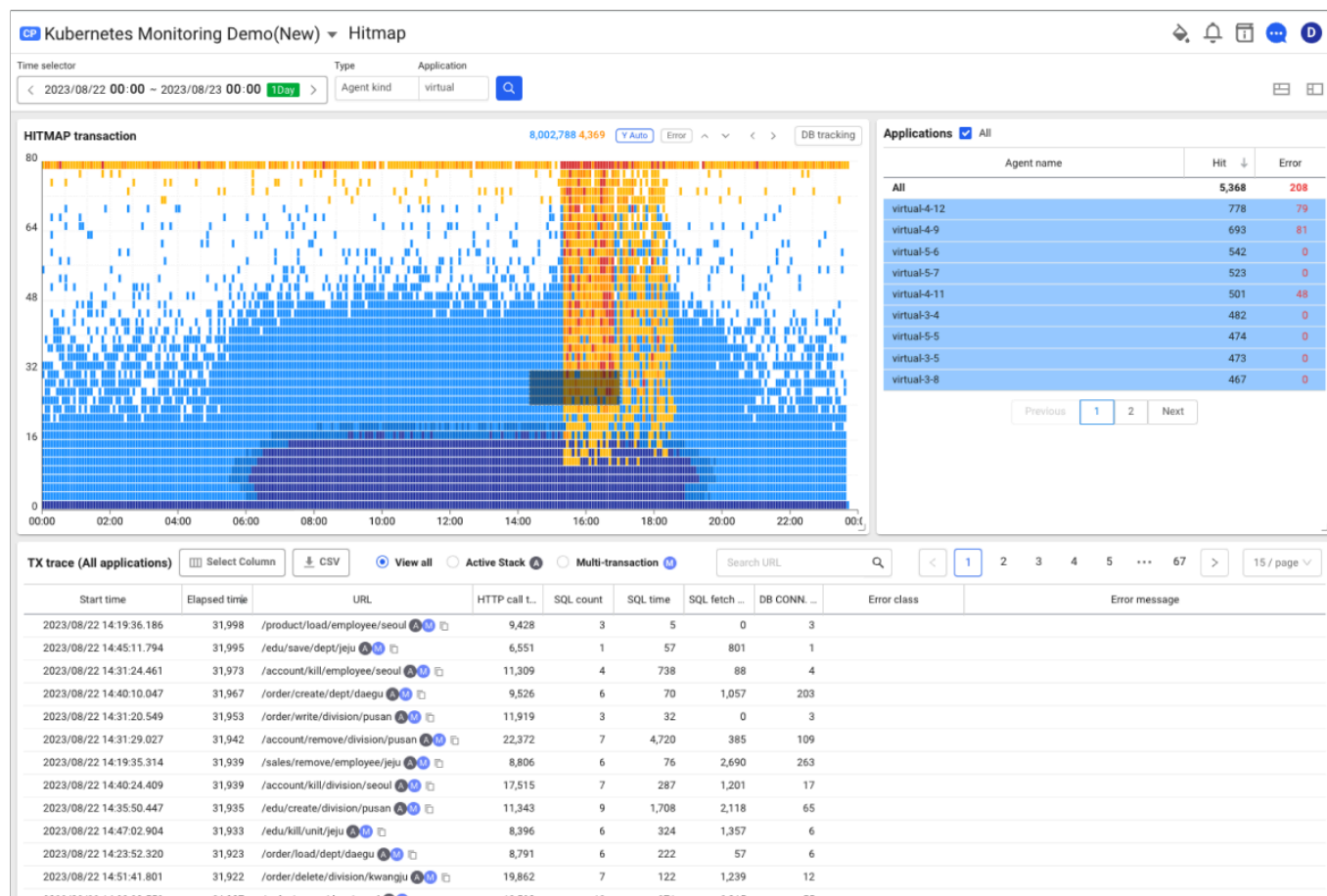
- **Initialization complete:** Indicates the time when initialization is completed.
- **Average CpuByLimit after initialization:** Indicates the average value of CpuByLimit after initialization.
- **Maximum memory after initialization:** Indicates the maximum amount of memory used by the container after initialization.
- **After initialization is complete (seconds) (up to 5 minutes):** Indicates the time in milliseconds for 5 minutes after initialization.

Application Hitmap

Home > Select Project > [Analysis](#) > HITMAP

You can check the hitmap of the applications in the container. The hitmap displays the distribution of response times as a map for the numbers of transactions. The search conditions are divided into time selection, classification, and application. The classification criteria are as follows:

| Classification | Description |
|----------------|---|
| Agent | Name of the application agent installed in the container |
| Agent type | Agent type by which agents are grouped (<code>OKIND</code>) |
| Agent server | Name of the Kubernetes node where the agents are running |



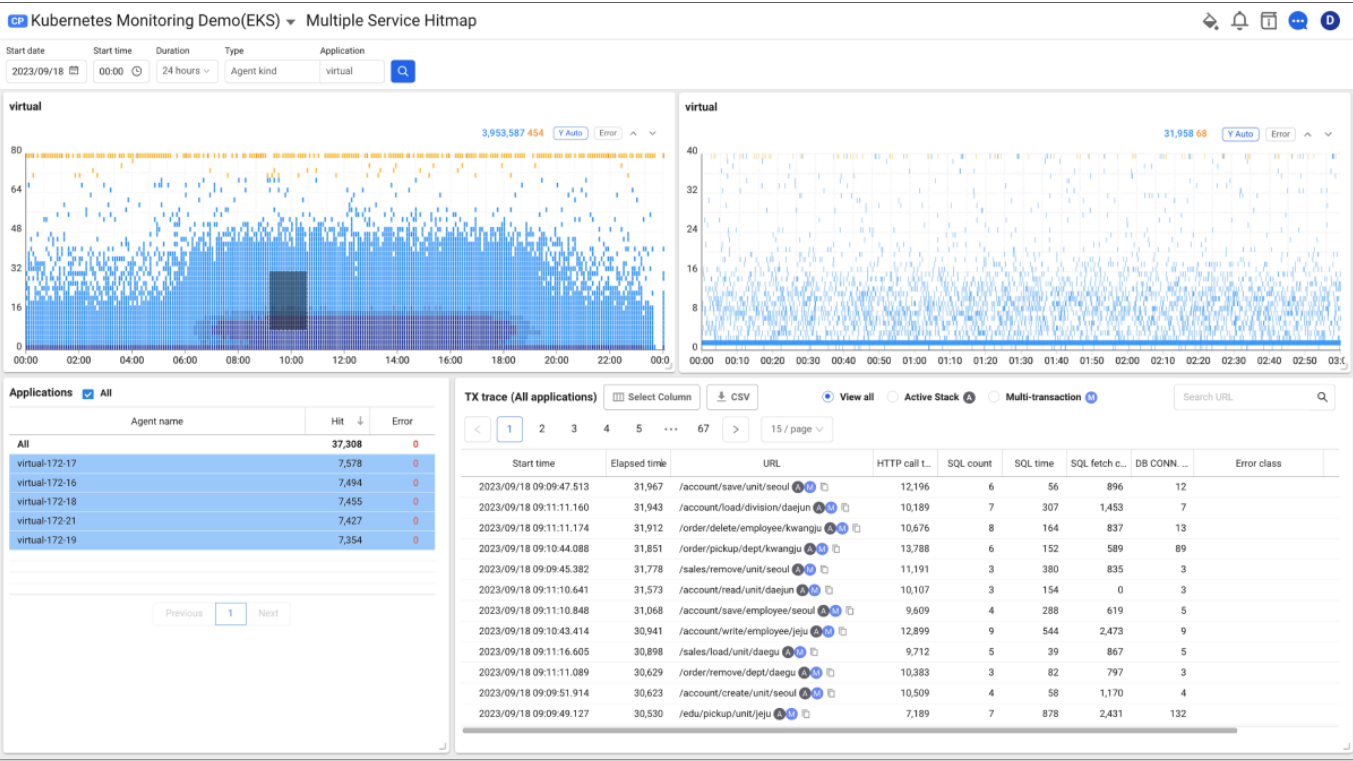
You can check the trace data by dragging a specific area of the hitmap. Trace data can be searched based on the URL, and can be checked by filtering data based on the transaction trace with active stacks and traces performed with multi-transactions.

! For more information about the hitmap, see the following.

- [Java](#)
- [Node.js](#)
- [Python](#)

Multi Service Hitmap

Home > Select Project > Analysis > Multiple Service Hitmap



You can see the hitmap chart for multiple services and compare and analyze the status of each service.

The search conditions are as follows:

| Item | Description |
|------------|---|
| Start Date | Searched date |
| Start Time | Not available for 3-day, 7-day, or 1-month searches |
| Period | <div><div>• 5min/10min/30min</div><div>• 1Hr/3Hr/24Hr</div></div> |

| Item | Description |
|-------------|--|
| | <ul style="list-style-type: none">• 3days/7days/1month |
| Type | Agent type |
| Application | Application |

❗ For more information about the hitmap, see the following.

- [Java](#)
- [Node.js](#)
- [Python](#)

MSA Analysis

What is the MSA analysis?

The MSA analysis function analyzes the proportion of calls between transactions collected through the WhaTap's **Multiple transaction trace** function. Based on this, dependencies between microservices can be checked at the URL level rather than instance-centric basis.

Microservice Architecture Pattern

For example, there are five microservices: A, B, C, D, and E. If a user calls A, it is assumed that the following three call patterns occur according to the logic.

1. A→B, C calls
2. A→B, C calls + then B→D call
3. A→B, C calls + then B→D, E calls

If these three patterns have been performed once, MSA analysis can check the association analysis data depending on the reference URL.

Microservice Architecture Caller & Callee

! For more information about **multiple transaction trace**, see [the following](#).

MSA Transaction Statistics

Home > Select Project > **Analysis** > **MSA Analysis**

The **MSA Analysis** menu is a tool that can effectively analyze the performance and stability for each service in the microservice environment. By analyzing the detailed call relationships for transactions, you can optimize the system performance and proactively identify potential problems based on the analysis.

CP

Kubernetes Monitoring Demo(EKS) ▾

MSA Analysis

Time selector

Filter

Order By

< 2023/10/10 00:00 ~ 2023/10/11 00:00 1Day >

Agent group ... ▾

Count ▾

Q

| | Detail | Project | Project code | Agent group name | Spec | Transaction | Count | Error | Average response |
|----|---|-----------------------|--------------|------------------|------|--------------------------------|-------|-------|------------------|
| 1 |  | netes Monitoring Demc | 33194 | virtual | | /account/save/employee/seoul | 4,978 | 1 | 2,539 |
| 2 |  | netes Monitoring Demc | 33194 | virtual | | /account/save/employee/pusan | 4,459 | 2 | 2,534 |
| 3 |  | netes Monitoring Demc | 33194 | virtual | | /account/save/employee/daegu | 4,147 | 0 | 2,448 |
| 4 |  | netes Monitoring Demc | 33194 | virtual | | /account/save/employee/kwangju | 3,956 | 2 | 2,384 |
| 5 |  | netes Monitoring Demc | 33194 | virtual | | /account/save/employee/daejun | 3,752 | 0 | 2,420 |
| 6 |  | netes Monitoring Demc | 33194 | virtual | | /account/save/employee/jeju | 3,742 | 1 | 2,520 |
| 7 |  | netes Monitoring Demc | 33194 | virtual | | /account/save/dept/seoul | 3,559 | 1 | 2,445 |
| 8 |  | netes Monitoring Demc | 33194 | virtual | | /account/save/dept/pusan | 3,556 | 1 | 2,724 |
| 9 |  | netes Monitoring Demc | 33194 | virtual | | /account/save/dept/daegu | 3,526 | 1 | 2,527 |
| 10 |  | netes Monitoring Demc | 33194 | virtual | | /account/save/dept/kwangju | 3,409 | 0 | 2,479 |
| 11 |  | netes Monitoring Demc | 33194 | virtual | | /account/save/dept/jeju | 3,318 | 1 | 2,690 |
| 12 |  | netes Monitoring Demc | 33194 | virtual | | /account/save/division/seoul | 3,200 | 1 | 2,558 |
| 13 |  | netes Monitoring Demc | 33194 | virtual | | /account/save/division/kwangju | 3,189 | 0 | 2,318 |
| 14 |  | netes Monitoring Demc | 33194 | virtual | | /account/save/division/pusan | 3,186 | 1 | 2,534 |
| 15 |  | netes Monitoring Demc | 33194 | virtual | | /account/save/division/daegu | 3,184 | 2 | 2,671 |
| 16 |  | netes Monitoring Demc | 33194 | virtual | | /account/save/dept/daejun | 3,182 | 0 | 2,339 |
| 17 |  | netes Monitoring Demc | 33194 | virtual | | /account/save/division/jeju | 3,078 | 0 | 2,416 |
| 18 |  | netes Monitoring Demc | 33194 | virtual | | /account/save/unit/seoul | 3,045 | 1 | 3,127 |
| 19 |  | netes Monitoring Demc | 33194 | virtual | | /account/save/division/daejun | 3,044 | 1 | 2,455 |
| 20 | | netes Monitoring Demc | 33194 | virtual | | /account/save/unit/pusan | 3,042 | 0 | 2,356 |

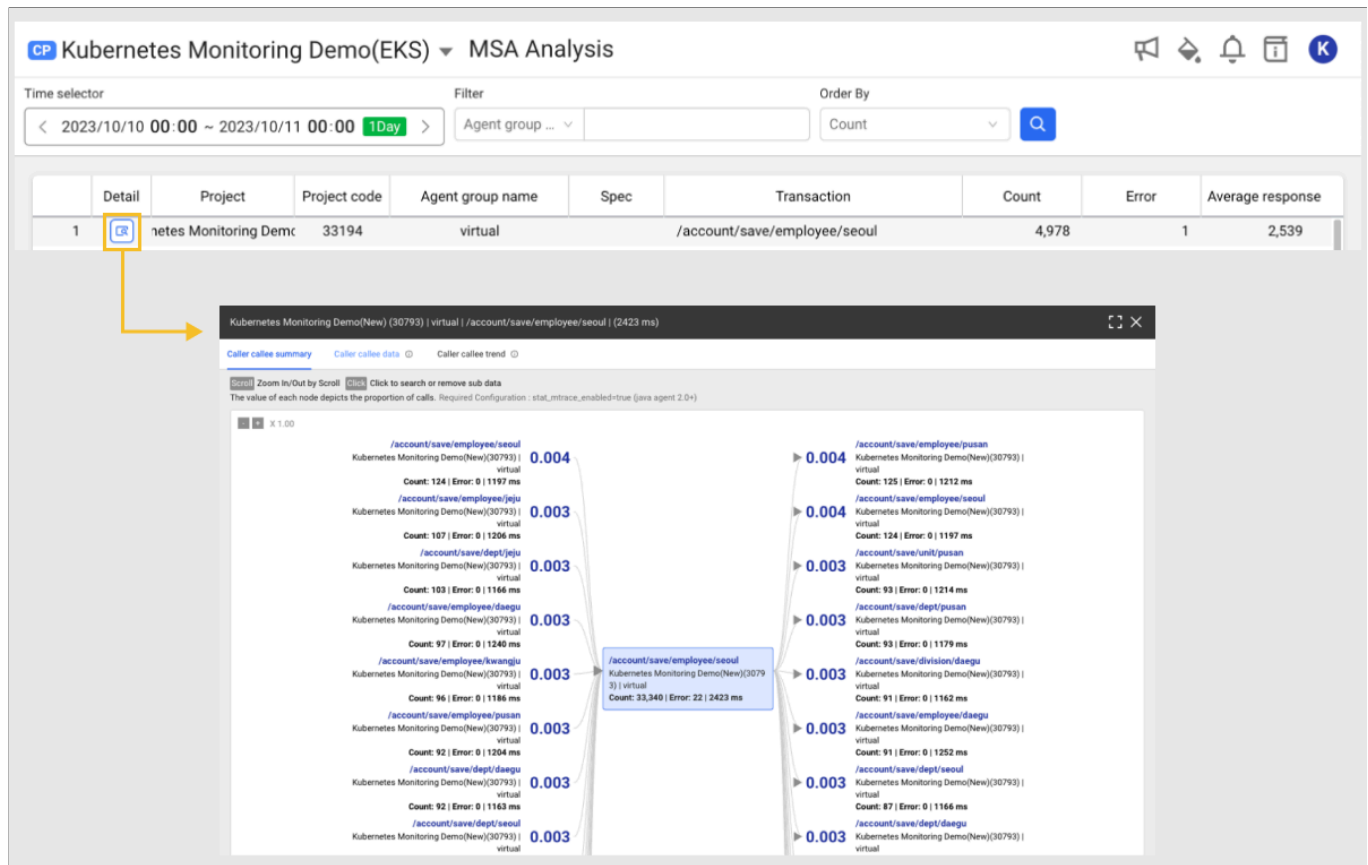
- **Time selector:** Select a desired search period. You can view a list of transactions including callers and callees on a daily basis.
- **Filter:** By filtering based on **Agent group name**, **Spec**, and **Transaction**, you can perform the search.
- **Order By:** Based on **Count**, **Error**, and **Time avg** (average response time), the list can be sorted.

- The time range that can be searched in **Time selector** is up to 1 day (24 hours). For more information on how to use the **Time selector** option.
- **Agent group name** is the group name classified as `whatap.okind` in the agent configuration options.
- After setting the conditions, select  to update the transaction list.

Analyzing in detail

When you select  for the item to view in the transaction list, the Display Detail window appears. You can view

the caller and callee for the transaction in the Display Detail window.



Each tab provides the following analysis features:

- **Caller callee summary:** It provides a visualized diagram for the relationship between the caller and callee of the transaction.
- **Caller callee data:** It provides the details of the transaction such as call count, error count, and response time.
- **Caller callee trend:** It provides a time series chart for performance changes in the number and rate of transaction calls over time.

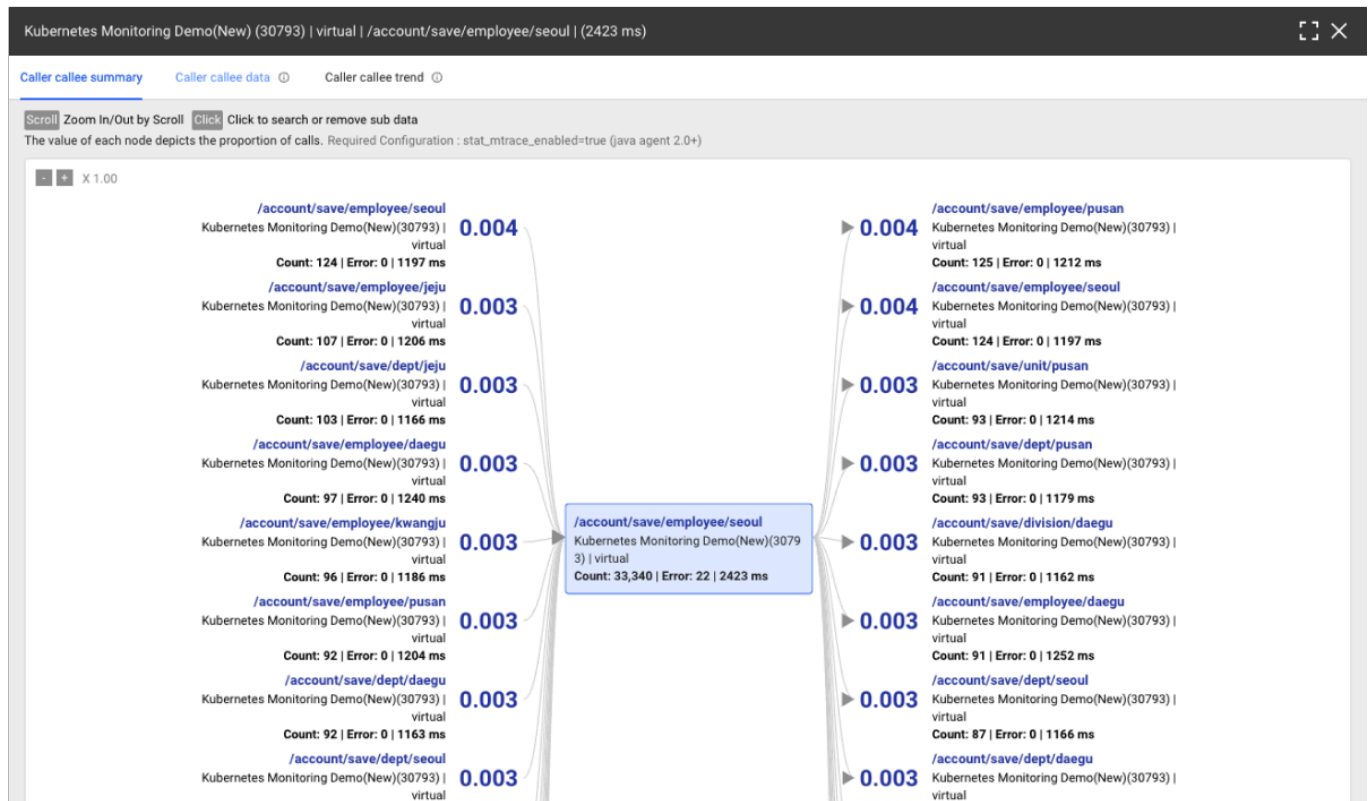


- You can view it in full screen when the button is selected on the upper right of the Display Detail window.
- To close the Display Detail window, select **Confirm** at the lower right or press the ESC key.

Caller callee summary

The **Caller callee summary** tab displays the dependency between the caller and callee of a transaction selected from the transaction list in a visualized diagram. You can intuitively understand the call relationships between services. Call paths and dependencies can be visually identified, which is advantageous for analyzing the causes of problems.

Analyzing call relationships



Caller and callee nodes display numerical data whose maximum is 1 such as 0.004 and 0.003, which represents its proportion among all callers. Likewise, the weight of each callee is displayed together on the right.

Through the above screen, you can analyze the dependencies between callers and callees based on the `/account/save/employee/seoul` transaction. You can visually see from and to which services the transactions are called.

Checking the performance metrics

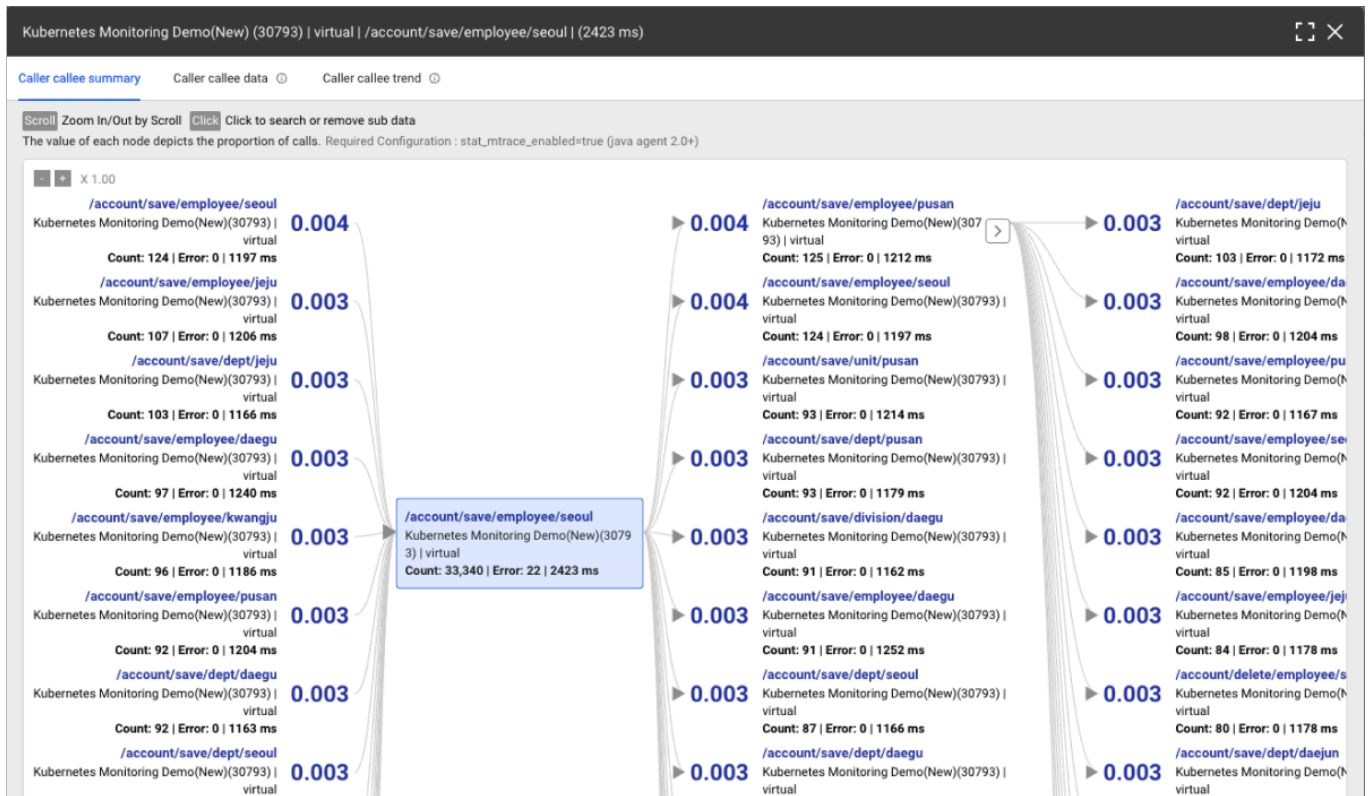


Each node consists of a transaction URL, project name, agent group name, and the following performance data:

- **Count:** Number of calls for the transaction
- **Error:** Number of errors occurred for calls
- **Average response time:** Average response time for the transaction, in milliseconds (ms)

Finding the transaction flow

When you hover your mouse over a caller or callee node, the `>` button appears. If you select this button, it expands into subnodes. By checking the entire flow of transactions, you can get answers in finding where problems occur.



- To close the subnodes, select > again.
- You can enlarge or reduce the screen using the scroll feature of the mouse. Or click the +/- buttons on the upper left of the screen.

Caller callee data

The **Caller callee data** tab displays the correlation between caller and callee for the reference URL (e.g. [/account/save/employee/seoul](#)) in a tabular format. You can specifically check the details of each transaction (call count, error count, average response time, etc.), enabling a precise, data-based analysis.

Kubernetes Monitoring Demo(EKS) (33194) | virtual | /account/save/employee/seoul | (2534 ms)

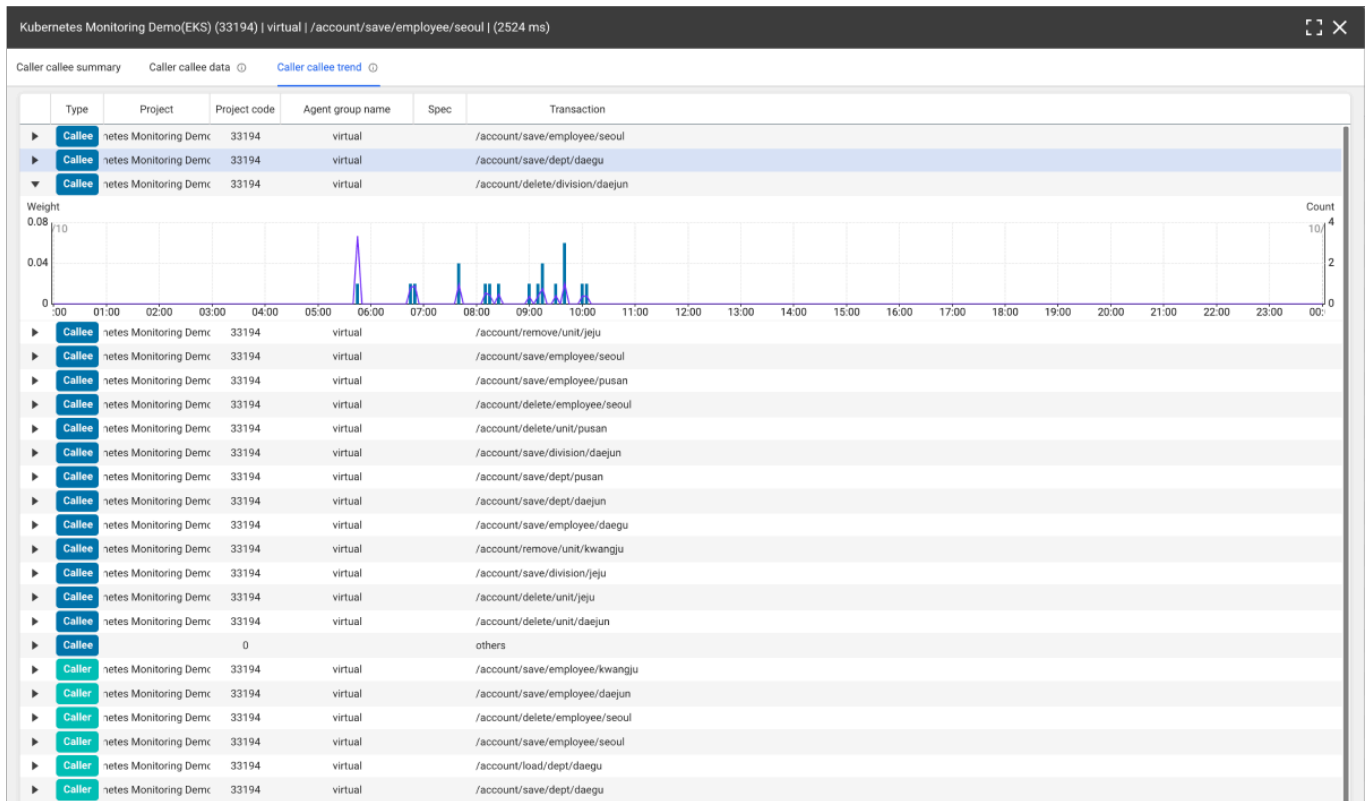
Caller callee summary [Caller callee data](#) [Caller callee trend](#) [Download CSV data](#)

| ID | Type | Project code | Project | Agent group na | Transaction | Depth | Count | Error | Average respor | Weight | 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-caller1-493697926 | 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... |

- It can be used to analyze performance issues by checking the number and rate of calls, number of errors, and average response time of each transaction.
- Through the **Type** column, you can easily distinguish between caller and callee and find the call relationship.
- By selecting ▶ for each transaction, you can additionally check the transactions related to the transaction.
- To save the data list in CSV format, select **Download CSV data** at the upper right.

Caller callee trend

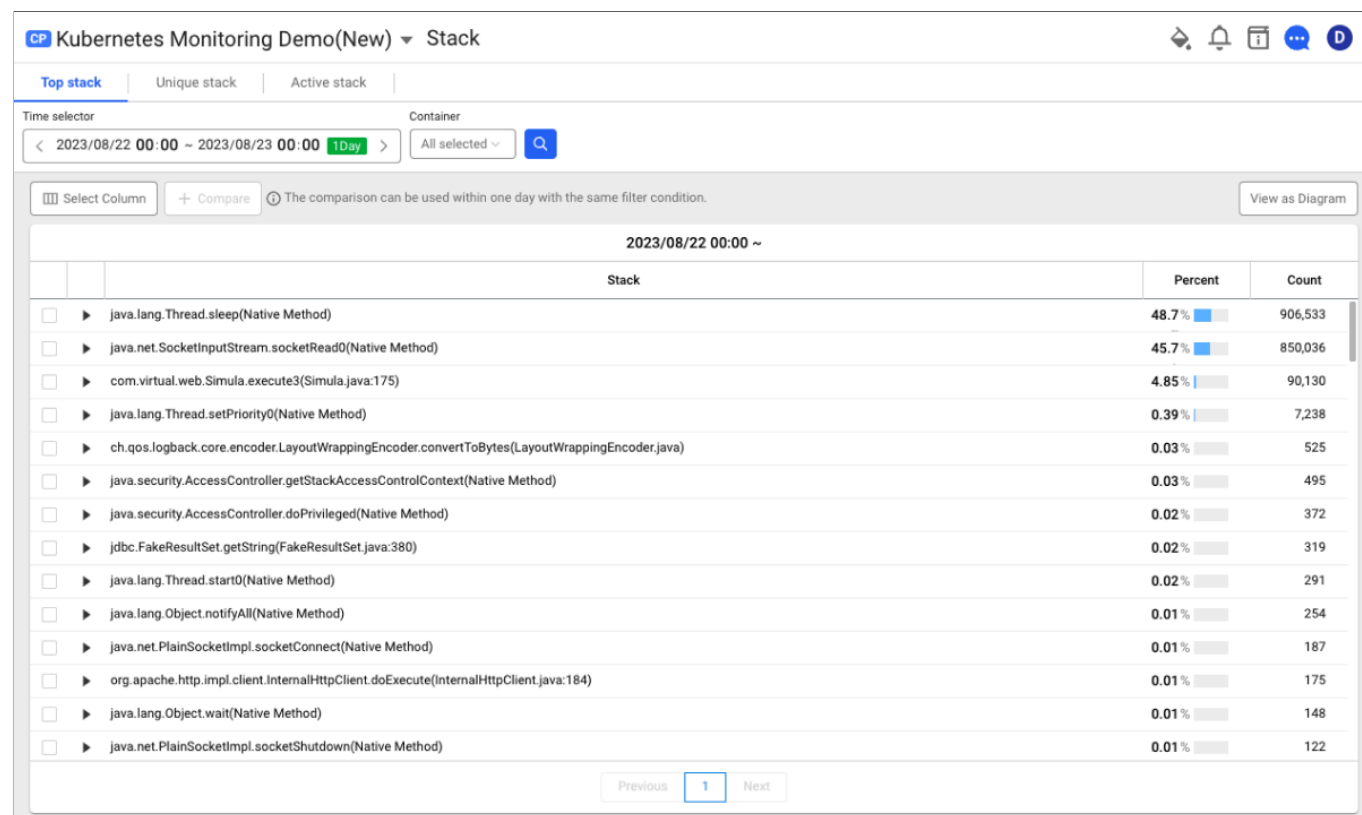
The **Caller callee trend** tab allows you to visually analyze the call relationships and performance data of the reference URL (</account/save/employee/seoul>) over time. This can help you find the pattern in the performance of transactions.



- You can see performance changes at a glance by visualizing the number and ratio of calls for specific transactions on a graph over time.
- Through the **Type** column, you can easily distinguish between caller and callee and find the call relationship.
- When you hover your mouse over the time series chart, you can see the numerical data through the tooltip.
- When you click a specific time zone on the chart, the **Analysis > Transaction Search** menu appears. You can view a list of transactions in the selected time zone.

Application Stack

Home > Select Project > [Analysis](#) > [Stack](#)



WhaTap can use the thread stacks collected every 10 seconds (default value) to analyze the performance delay sections in the method level. This feature allows you to check the [Top stack](#), [Unique stack](#), and [Active Stack](#) of each container.

! For more information about [top stack](#), [unique stack](#), and [active stack](#), see the following.

- [Java](#)
- [Python](#)

Metrics

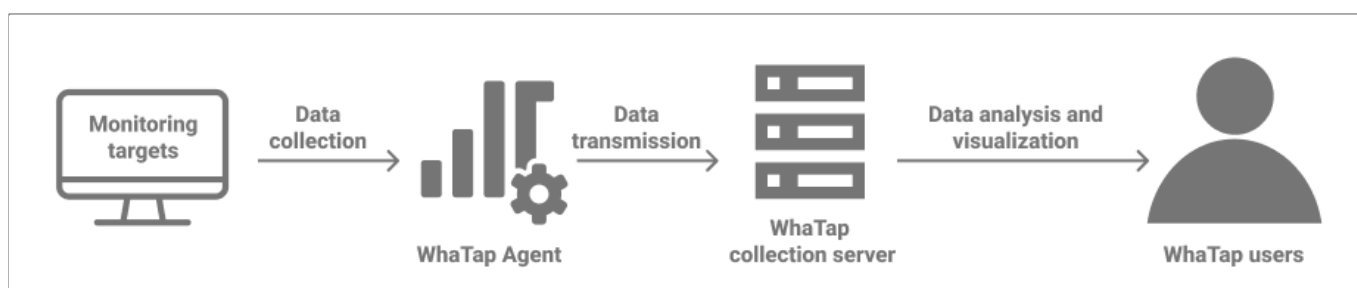
What is a metric?

A **metric** is time-series data that represents the state of a monitoring target as numerical values. It is the result of processing performance data collected by the WhaTap agent from monitoring targets.

Through metrics, you can view key performance indicators such as memory usage per server and DB connection time at a glance. After identifying a problem, you can perform detailed analysis through logs and traces.

Resource usage statistics help you estimate the required amount of resources, which can also be used for cost optimization.

How WhaTap collects metrics



To collect metrics, install the WhaTap agent on the monitoring target. Metrics sent by the agent are stored on the WhaTap collection server.

For example, to start **Kubernetes** monitoring, first [install the agent](#). See [Kubernetes metrics](#) for available metrics.

Metric components

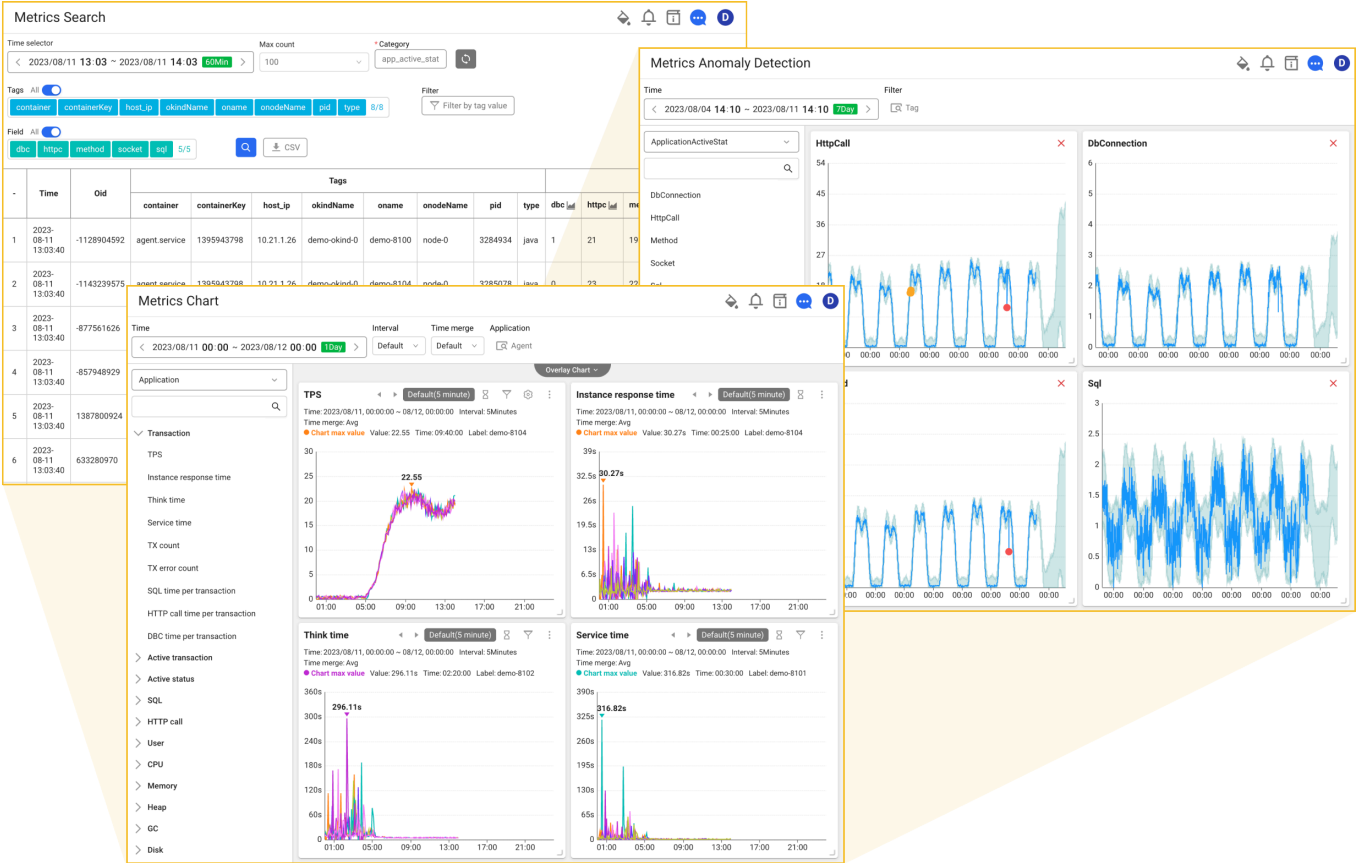
WhaTap **metrics** consist of the following components:

- **Category:** A name that groups related metrics.
- **Tags:** Labels that identify collection targets. Stores rarely-changing unique information such as IP, Oname, and Host. A metric can contain multiple tags.
- **Fields:** Stores actual measured values. A metric can contain multiple fields.
- **Time:** The time when the metric was collected.
- **Oid:** The unique number of the agent that collected the metric.
- **Oname:** The name of the agent that collected the metric.

Viewing and visualizing metrics

WhaTap provides menus to view collected metrics in various ways.

- **Metrics Search:** View the original metric data.
- **Metrics Chart:** View metric data through visualized charts.
- **Metrics Anomaly Detection:** Detect anomalies by comparing with metric patterns learned by AI.



Kubernetes Metrics

This page lists the metrics collected by Kubernetes monitoring, organized by category. Each category includes its Tags (dimensions) and Fields (measurements).

- Some metrics are also used in server monitoring and application monitoring.
- To search or chart metrics directly, see [Metrics Search](#) and [Metrics Chart](#).

Container(container) metrics

The `container` category collects all custom labels set on the container's pods as tags.

- Target: Cluster project, Namespace project
- Collection interval: 5 seconds
- Statistical data: 5 minutes

Tags

| Tag | Type | Unit | Description |
|--------------|------|------|---------------------------------------|
| agentOid | - | - | Node agent ID (unique) |
| agentPcode | - | - | Project code (unique) |
| command | - | - | Execution command |
| containerId | - | - | Container ID (unique) |
| containerKey | - | - | Container key |
| created | - | - | Time stamp generated by the container |
| image | - | - | Container image name |

| Tag | Type | Unit | Description |
|----------------|------|------|---|
| imageHash | - | - | Image hash value |
| imageId | - | - | Image ID |
| k8s-app | - | - | Value for the pod's label k8s-app |
| microOid | - | - | Unique ID of the WhaTap APM agent installed in the container |
| name | - | - | Container name |
| namespace | - | - | Namespace to which the container belongs |
| namespaceHash | - | - | Hash value of the namespace to which the container belongs |
| okind | - | - | Unique ID of OKIND specified in the WhaTap APM agent installed in the container |
| okindName | - | - | Name of OKIND specified in the WhaTap APM agent installed in the container |
| oname | - | - | Name of the WhaTap APM agent installed in the container |
| onode | - | - | Unique ID of the node agent on which the container is running |
| onodeName | - | - | Node name on which the container is running |
| podHash | - | - | Hash value of the container's Pod |
| podName | - | - | Container's Pod name |
| replicaSetHash | - | - | Hash value of the container's replica set |
| replicaSetName | - | - | Name of the container's replica set |
| whatap_project | - | - | Name of the WhaTap project to which the container belongs |

Fields

| Field | Type | Unit | Shortname/Name/Description |
|---------------|------|---------|--|
| blkio_rbps | - | byte | IoReadBytes |
| | | | Container Block I/O Read Byte |
| | | | Sum of bytes read per second across all block devices in the container |
| blkio_riops | - | count | IoReadIops |
| | | | Container Block I/O Read IOPS |
| | | | Sum of counts read per second across all block devices in the container |
| blkio_wbps | - | byte | IoWriteBytes |
| | | | Container Block I/O Write Byte |
| | | | Sum of bytes written per second across all block devices in the container |
| blkio_wiops | - | count | IoWriteIops |
| | | | Container Block I/O Write IOPS |
| | | | Sum of counts written per second across all block devices in the container |
| cpu_per_quota | - | percent | CpuByLimit |
| | | | Container CPU usage by limit (%) |
| | | | Container CPU utilization by limit |

| Field | Type | Unit | Shortname/Name/Description |
|----------------------|------|----------------|---|
| cpu_quota | - | millicores | CpuLimit |
| | | | Container CPU Limit (core) |
| | | | Container CPU Limit Quota
If the limit is not set, the total CPU cores of the node where the container is running appears in millicores |
| cpu_quota_percent | - | percent | CpuLimitByNode |
| | | | Container CPU Limit by Node(%) |
| | | | Container CPU Limit quota relative to node CPU
- If Limit is not set, the total CPU cores of the node where the container is running are displayed as a percentage |
| cpu_sys | - | percent | CpuSysByNode |
| | | | Container CPU Sys Usage by Node(%) |
| | | | Container CPU System Utilization against Node CPU |
| cpu_throttledperiods | - | count | CpuThrottledCnt |
| | | | Container CPU Throttling Count |
| | | | Container CPU Throttled Count |
| cpu_throttledtime | - | nanosecond(ns) | CpuThrottledTime |
| | | | Container CPU Throttling Time |
| | | | Container CPU Throttled Time |
| cpu_total | - | percent | CpuByNode |

| Field | Type | Unit | Shortname/Name/Description |
|-----------------|------|------------|---|
| | | | Container CPU Usage by Node(%) |
| | | | Container CPU Utilization against Node CPU |
| cpu_total_milli | - | millicores | CpuTotUsage |
| | | | Container CPU Usage (millicore) |
| | | | Container CPU Usage |
| cpu_user | - | percent | CpuUserByNode |
| | | | Container CPU User Usage by Node(%) |
| | | | Container CPU User Utilization against Node CPU |
| cpu_request | - | millicores | CpuRequest |
| | | | Container CPU Request (core) |
| | | | Container CPU Request |
| cpu_per_request | - | percent | CpuByRequest |
| | | | Container CPU Usage by Request(%) |
| | | | Utilization against Container CPU Request
= $\text{cpu_total_milli} / \text{cpu_request} * 100$ |
| mem_failcnt | - | count | MemFailCnt |
| | | | Container Memory Failure Count |
| | | | Container Memory Limit reached Count |
| mem_limit | - | byte | MemLimit |

| Field | Type | Unit | Shortname/Name/Description |
|------------------|------|---------|--|
| | | | Container Memory Limit (byte) |
| | | | Container Memory Limit Size |
| mem_maxusage | - | byte | MemMaxUsage |
| | | | Container Memory Max Usage (byte) |
| | | | Recorded maximum container memory usage
See the guide below for details. |
| mem_percent | - | percent | MemWsByLimit |
| | | | Container Memory Working Set by Limit(%) |
| | | | Working Set Usage based on Container Memory Limit
$= \text{mem_usage} / \text{mem_limit} * 100$ |
| mem_totalcache | - | byte | MemTotCache |
| | | | Container Memory Total Cache (byte) |
| | | | Container's Total Cache Size |
| mem_totalpgfault | - | count | MemTotPageFaultCnt |
| | | | Container Memory Total Page Fault Count |
| | | | Container's Page Fault Count |
| mem_totalrss | - | byte | MemTotRss |
| | | | Container Memory Total RSS (byte) |
| | | | Container's Total RSS Memory Size |

| Field | Type | Unit | Shortname/Name/Description |
|-------------------------|------|---------|--|
| mem_totalrss_percent | - | percent | MemTotRssByLimit |
| | | | Container Memory Total RSS By Limit (%) |
| | | | Container's Total RSS Memory Utilization |
| mem_totalunevictable | - | byte | MemTotUnevictable |
| | | | Container Memory Total Unevictable (byte) |
| | | | Container's Total Unevictable Memory Size |
| mem_usage | - | byte | MemUsage |
| | | | Container Memory Usage (byte) |
| | | | Container Memory Usage |
| mem_working_set | - | byte | MemWs |
| | | | Container Memory Working Set (byte) |
| | | | Container memory working set
= mem_usage - inactive file |
| mem_working_set_percent | - | percent | MemWsByLimit |
| | | | Container Memory Working Set by Limit (%) |
| | | | Working Set Usage based on Container Memory Limit
= mem_usage / mem_limit * 100 |
| mem_request | - | byte | MemRequest |
| | | | Container Memory Request (byte) |

| Field | Type | Unit | Shortname/Name/Description |
|------------------|------|---------|--|
| | | | Container Memory Request Size |
| mem_per_request | - | percent | MemWsByRequest |
| | | | Container Memory Working Set by Request (%) |
| | | | Working Set Usage based on Container Memory Request
= mem_working_set / mem_request * 100 |
| network_rbps | - | byte | NetRxBytes |
| | | | Container Network Receive Byte |
| | | | Sum of bytes read per second across all block devices in the container |
| network_rdropped | - | byte | NetRxDropped |
| | | | Container Network Receive Dropped |
| | | | Container Network Receive Dropped Count |
| network_rerror | - | byte | NetRxError |
| | | | Container Network Receive Error |
| | | | Container Network Receive Error Count |
| network_riops | - | byte | NetRxIops |
| | | | Container Network Receive IOPS |
| | | | Container Network Receive Error Count |
| network_wbps | - | byte | NetTxBytes |

| Field | Type | Unit | Shortname/Name/Description |
|------------------|------|---------|---|
| | | | Container Network Transmit Byte |
| | | | Container Network Transmit Data Size |
| network_wdropped | - | count | NetTxDropped |
| | | | Container Network Transmit Dropped |
| | | | Container Network Transmit Dropped Count |
| network_werror | - | count | NetTxError |
| | | | Container Network Transmit Error |
| | | | Container Network Transmit Error Count |
| network_wiops | - | count | NetTxIops |
| | | | Container Network Transmit IOPS |
| | | | Container Network Transmit Error Count |
| node_cpu | - | percent | ConNodeCpu |
| | | | Container Work Node CPU Usage (%) |
| | | | CPU Usage of the Node where the container is running |
| node_mem | - | percent | ConNodeMem |
| | | | Container Work Node Memory Usage (%) |
| | | | Memory Usage of the Node where the container is running |

| Field | Type | Unit | Shortname/Name/Description |
|---------------|---------|------|--|
| phase | string | - | Pod lifecycle
① PENDING
② RUNNING
③ SUCCEEDED
④ FAILED
⑤ UNKNOWN |
| restart_count | integer | - | ConRestartCnt |
| | | | Container Restart Count |
| | | | Number of container restarts |
| state | integer | - | ConState |
| | | | Container Current State |
| | | | Container State Code
① RUNNING = 114
② PAUSE = 112
③ RESTARTING = 101
④ OOMKILLED = 111
⑤ DEAD = 100
⑥ WAITING = 119 |
| status | string | - | ConStatus |
| | | | Container Current Status |
| | | | Container State Information
① running: Displays the uptime information
② waiting/terminated: Displays the reason of the state |

ⓘ `mem_maxusage` indicates the maximum memory usage recorded while the container was running. However, if the Linux kernel version is lower than 5.19, the raw data for this metric may not be

! supported. In this case, the value may be displayed as 0. Update the Linux kernel version to 5.19 or later to collect this metric.

Kubernetes node (kube_node) metrics

The kube_node category collects all custom labels set on the node as tags.

- Target: Cluster project, Namespace project
- Collection interval: 5 seconds
- Statistical data: 5 minutes, 1 hour

Tags

| Tag | Type | Unit | Description |
|----------|------|------|-------------|
| nodeName | - | - | Node name |

The following tag has been added.

| Tag | Type | Unit | Description |
|--------------------|--------|------|--|
| infrastructureType | string | - | Node infrastructure type
- baremetal , vm , unknown |

Fields

| Field | Type | Unit | Description |
|--------------------|------|------------|--|
| allocatable_cpu | - | millicores | CPU size that can be assigned to node |
| allocatable_memory | - | byte | Memory size that can be assigned to node |

| Field | Type | Unit | Description |
|------------------|---------|------------|---|
| allocatable_pods | integer | - | Number of Pods that can be assigned to node |
| limit_cpu | - | millicores | Sum of node CPU limits |
| limit_memory | - | byte | Sum of node memory limits |
| pods | integer | - | Total number of node Pods |
| request_cpu | - | millicores | Sum of node CPU requests |
| request_memory | - | byte | Sum of node memory requests |

The GPU quantity of a node is collected as parallel list fields by device type. `*_types` and `*_counts` correspond 1:1 by index, and hold `nvidia.com/gpu` (whole-GPU) and all MIG profiles (`nvidia.com/mig-*`) as individual entries.

| Field | Type | Unit | Description |
|------------------------|------|-------|--|
| gpu_capacity_types | list | - | List of device types of the node GPU capacity |
| gpu_capacity_counts | list | count | Quantity per type above for capacity (1:1 index correspondence) |
| gpu_allocatable_types | list | - | List of device types of the node GPU allocatable |
| gpu_allocatable_counts | list | count | Quantity per type above for allocatable (1:1 index correspondence) |

ⓘ Starting from `v1.9.14` , the whole-GPU-only scalar fields `capacity_gpu` and `allocatable_gpu` have been removed. This prevents the problem where, in MIG environments, the scalar fields hold only `nvidia.com/gpu` and omit MIG slices, making the aggregation a false total. The whole-GPU quantity is also checked from the `nvidia.com/gpu` entry in the list, and the total and per-type aggregations are the sum of `*_counts` .



| Category | Removed scalar field | Replacement list field |
|-----------|-----------------------------------|--|
| kube_node | capacity_gpu ,
allocatable_gpu | gpu_capacity_types / gpu_capacity_counts ,
gpu_allocatable_types / gpu_allocatable_counts |

Kubernetes node performance (kube_node_perf) metrics

The kube_node_perf category collects node-level performance metrics by summing up the resource usage and allocation of all Pods/Containers scheduled on the node.

- Target: Cluster project, Namespace project
- Collection interval: 5 seconds
- Statistical data: 5 minutes, 1 hour

Tags

| Tag | Type | Unit | Description |
|------------|------|------|--------------------|
| onode | - | - | Node hash value |
| onodeName | - | - | Node name |
| agentPcode | - | - | Agent project code |
| agentOid | - | - | Agent unique ID |

The following tag has been added.

| Tag | Type | Unit | Description |
|----------------|------|------|---|
| cgroup_version | - | - | cgroup mount mode of the node host
- unified (cgroup v2), hybrid, legacy (cgroup v1), unknown (inaccessible) |

The `/sys/fs/cgroup` mount mode of the node host is detected directly and collected as a tag. You can aggregate the cgroup version status per node on the server.

The node virtualization determination tags have been added.

| Tag | Type | Unit | Description |
|----------------|--------|------|--|
| virtualization | string | - | Whether the node is virtualized
- baremetal, vm, unknown |
| hypervisor | string | - | Hypervisor vendor name
- - for bare metal or when it cannot be determined |

The vendor is first cross-checked against the DMI information of the node host (`/sys/class/dmi/id/sys_vendor` , `product_name`). If no vendor matches, only whether it is a virtual machine is determined by the hypervisor flag in `/proc/cpuinfo` . Non-x86 nodes without DMI information are collected as `unknown` .

Vendors such as QEMU/KVM, VMware, Hyper-V, Xen, VirtualBox, Parallels, Amazon EC2, Google GCE, OpenStack, Nutanix AHV, oVirt/RHV, and Alibaba Cloud are recognized. The value does not change while the node is running, so it is read once and cached.

Constraint: Bare metal provided by a cloud provider, such as an Amazon EC2 `.metal` instance, is not distinguishable from a general instance by DMI information alone and is determined as `vm` .

Collected on node agent 1.9.21 or later.

Fields

| Field | Type | Unit | Description |
|------------|------|------------|---|
| cpuRequest | long | millicores | Total CPU Request sum of all containers in the node |

| Field | Type | Unit | Description |
|------------------|-------|------------|---|
| cpuLimit | long | millicores | Total CPU Limit sum of all containers in the node |
| cpuTotal | long | millicores | Total CPU usage sum of all containers in the node |
| cpuPerRequest | float | percent | Actual CPU utilization relative to node CPU Request |
| cpuPerLimit | float | percent | Actual CPU utilization relative to node CPU Limit |
| memoryRequest | long | byte | Total Memory Request sum of all containers in the node |
| memoryLimit | long | byte | Total Memory Limit sum of all containers in the node |
| memTotal | long | byte | Total Memory usage sum of all containers in the node |
| memoryPerRequest | float | percent | Actual Memory utilization relative to node Memory Request |
| memoryPerLimit | float | percent | Actual Memory utilization relative to node Memory Limit |

The following fields have been added.

| Field | Type | Unit | Description |
|-----------------|-------|---------|--|
| conntrack_count | long | count | Current number of conntrack entries |
| conntrack_max | long | count | Upper limit of conntrack entries (<code>nf_conntrack_max</code>) |
| conntrack_pct | float | percent | Usage against the upper limit |

- The number of entries is collected from the `/proc/1/net/stat/nf_conntrack` path to read the value based on the host network namespace.
- Nodes where the `nf_conntrack` module is not loaded do not send the fields. This behavior distinguishes the value `0` from non-collection.
- Collected on node agent 1.9.18 or later. You can turn the collection off with the `collect_conntrack_enabled` option.

Kubernetes event (kube_event) metrics

The `kube_event` category collects cluster-wide data for cluster projects, and collects data only for events that occurred in the namespace for namespace projects.

- Target: Cluster project, Namespace project
- Collection interval: 5 seconds
- Statistical data: 5 minutes, 1 hour

Tags

| Tag | Type | Unit | Description |
|------------|------|------|--|
| field_path | - | - | Field Path |
| kind | - | - | Object type on which the event occurred |
| name | - | - | Kubernetes object name on which the event occurred |
| namespace | - | - | Namespace on which the event occurred |
| reason | - | - | Event occurrence cause |
| type | - | - | Event type
- Warning or Normal |
| uid | - | - | UID
- Object where an event occurred |

Fields

| Field | Type | Unit | Description |
|--------|--------|------|-------------|
| action | string | - | Action name |

| Field | Type | Unit | Description |
|---------------------------|---------|-------|--|
| count | - | count | Event occurrence count |
| event_time | integer | - | Time stamp for the first event |
| first_timestamp | integer | - | First event occurrence time |
| last_timestamp | integer | - | Last event occurrence time |
| message | string | - | Event Message |
| reasonFiled | string | - | Event reason |
| reporting_component | string | - | Component that reports the current event |
| reporting_instance | string | - | Instance that reports the current event |
| series_last_observed_time | integer | - | Series last observed time |

Kubernetes Cluster (kube_stat) metrics

The kube_stat category collects all the clusters for the cluster project, and the namespace projects collects objects associated with the namespace.

- Target: Cluster project, Namespace project
- Collection interval: 5 seconds
- Statistical data: 5 minutes, 1 hour

Tags

| Tag | Type | Unit | Description |
|------|------|------|-------------------------|
| name | - | - | kube_stat (fixed value) |

The following tag has been added.

| Tag | Type | Unit | Description |
|-----------|------|------|---------------------------------------|
| namespace | - | - | Name of the namespace being collected |

Fields

| Field | Type | Unit | Description |
|-------------------------------|---------|------------|--|
| allocatable_cpu | - | millicores | Number of all cores in the cluster (cluster project only) |
| allocatable_ephemeral-storage | - | byte | Cluster-wide allocatable ephemeral storage (cluster project only) |
| allocatable_hugepages-1gi | - | byte | Cluster-wide allocatable hugepages-1Gi (cluster project only) |
| allocatable_hugepages-2mi | - | byte | Cluster-wide allocatable hugepages-2Mi (cluster project only) |
| allocatable_memory | - | byte | Total memory allocatable in the cluster (cluster project only) |
| allocatable_pods | integer | - | Number of pods that can be allocated |
| available_pod | integer | - | Number of pods whose phase is in Running state |
| desired_pod | integer | - | Sum of the number of pods deployed without <code>metadata.ownerReferences</code> and the number of desired pods defined in Kubernetes objects (ReplicaSet, Daemonset, StatefulSet) |
| | | | Same as the number of pods retrieved by <code>kubectl get pods -A</code> |
| nodes | integer | - | Number of nodes |
| pod_phase_Pending | integer | - | Number of pending pods |

| Field | Type | Unit | Description |
|------------------------|---------|------------|---|
| pod_phase_Running | integer | - | Number of running pods |
| running_containers | integer | - | Number of running containers |
| stopped_containers | integer | - | Number of stopped containers |
| total_available_cpu | integer | - | Total allocatable CPU |
| total_available_memory | integer | - | Total sum of allocatable memory |
| total_limit_cpu | - | millicores | Total sum of limit CPU |
| total_limit_memory | - | byte | Total sum of limit memory |
| total_request_cpu | - | millicores | Total sum of request CPU |
| total_request_memory | - | byte | Total sum of request memory |
| unavailable_pod | integer | - | Number of pods whose phase is not in Running state (Pending, Failed, Succeeded) |
| waiting_containers | integer | - | Waiting container count |

The upper limits and consumption of the ResourceQuota configured for a namespace are collected. The following fields have been added.

| Field | Type | Unit | Description |
|---------------------------|------|------------|---|
| quota_count | long | count | Number of ResourceQuotas configured for the namespace |
| quota_request_cpu_hard | long | millicores | Upper limit of the CPU Request quota |
| quota_request_cpu_used | long | millicores | Consumption of the CPU Request quota |
| quota_request_memory_hard | long | byte | Upper limit of the Memory Request quota |

| Field | Type | Unit | Description |
|--|------|------------|---|
| <code>quota_request_memory_used</code> | long | byte | Consumption of the Memory Request quota |
| <code>quota_limit_cpu_hard</code> | long | millicores | Upper limit of the CPU Limit quota |
| <code>quota_limit_cpu_used</code> | long | millicores | Consumption of the CPU Limit quota |
| <code>quota_limit_memory_hard</code> | long | byte | Upper limit of the Memory Limit quota |
| <code>quota_limit_memory_used</code> | long | byte | Consumption of the Memory Limit quota |
| <code>quota_pods_hard</code> | long | count | Upper limit of the Pod count quota |
| <code>quota_pods_used</code> | long | count | Consumption of the Pod count quota |

- The `status.hard` and `status.used` values calculated by Kubernetes are used as they are.
- When a namespace has several ResourceQuotas with different scopes, the values are summed per item.
- A namespace with no quota configured sends only `quota_count` as `0` and does not send the remaining fields. This behavior distinguishes no quota from an upper limit of `0`.
- Collected on master agent 1.9.17 or later.

Permission: The `list` permission for the core API `resourcequotas` is required in the ClusterRole. Without the permission, only the ResourceQuota collection is skipped and the rest of the `kube_stat` collection continues.

Pod (`kube_pod`) metrics

The `kube_pod` category collects all custom labels set on the Pod as tags.

- Target: Master (cluster) project, Namespace project
- Collection interval: 5 seconds
- Statistical data: 5 minutes

Tags

| Tag | Type | Unit | Description |
|--------------------|------|------|--|
| agentOid | - | - | Node agent ID (unique) |
| agentPcode | - | - | Project code (unique) |
| command | - | - | Execution command |
| containerIds | - | - | Container ID that belongs to the Pod |
| containerIdsCount | - | - | Number of containerIds |
| containerKeys | - | - | Hash value for the container ID that belongs to the pod |
| containerKeysCount | - | - | Number of containerKeys |
| DaemonSet | - | - | DaemonSet name of the pod |
| Deployment | - | - | Deployment |
| k8s-app | - | - | Value for the pod's label k8s-app |
| microOid | - | - | ID of the agent running on the applications inside the Pod's container. |
| microOids | - | - | Multiple IDs of the agents running on applications inside multiple containers in the pod |
| microOidsCount | - | - | Number of microOids |
| name | - | - | Pod Name |
| onames | - | - | Name of the agent running on the applications inside the Pod's container. |
| onamesCount | - | - | Number of onames |

| Tag | Type | Unit | Description |
|----------------|------|------|--|
| podName | - | - | Pod Name |
| namespace | - | - | Namespace to which the Pod belongs |
| namespaceHash | - | - | Hash value of the namespace to which the Pod belongs |
| replicaSetHash | - | - | Hash value of ReplicaSet of the Pod |
| replicaSetName | - | - | ReplicaSet name of the Pod |
| whatap_project | - | - | Name of the WhaTap project to which the Pod belongs |

Fields

| Field | Type | Unit | Shortname, Name, Description |
|-------------|------|-------|--|
| blkio_rbps | - | byte | IoReadBytes |
| | | | Pod Block I/O Read Byte |
| | | | Sum of bytes read per second across all block devices in the Pod |
| blkio_riops | - | count | IoReadIops |
| | | | Pod Block I/O Read IOPS |
| | | | Sum of cases read per second across all block devices in the Pod |
| blkio_wbps | - | byte | IoWriteBytes |
| | | | Pod Block I/O Write Byte |
| | | | Sum of bytes written per second across all block devices in |

| Field | Type | Unit | Shortname, Name, Description |
|----------------------|------|---------|--|
| | | | the Pod |
| blkio_wiops | - | count | IoWriteIops |
| | | | Pod Block I/O Write IOPS |
| | | | Sum of cases written per second across all block devices in the Pod |
| cpu_per_limit | - | percent | CpuByLimit |
| | | | Pod CPU Usage by Limit (%) |
| | | | Container CPU utilization by limit |
| cpu_per_request | - | percent | CpuByRequest |
| | | | Pod CPU Usage by Request (%) |
| | | | Total CPU utilization based on the CPU requests |
| cpu_quota_percent | - | percent | CpuLimitByNode |
| | | | Pod CPU Limit by Node (%) |
| | | | Pod CPU limit quota against the node
- If the limit is not set, the total CPU cores of the node where the Pod is running appears in percentage. |
| cpu_sys | - | percent | CpuSysByNode |
| | | | Pod CPU Sys Usage by Node (%) |
| | | | Pod CPU System Utilization against Node CPU |
| cpu_throttledperiods | - | count | CpuThrottledCnt |

| Field | Type | Unit | Shortname, Name, Description |
|-------------------|------|----------------|---|
| | | | Pod CPU Throttling Count |
| | | | Pod CPU Throttled Count |
| cpu_throttledtime | - | nanosecond(ns) | CpuThrottledTime |
| | | | Pod CPU Throttling Time |
| | | | Pod CPU Throttled Time |
| cpu_total | - | percent | CpuByNode |
| | | | Pod CPU Usage by Node (%) |
| | | | Pod CPU Utilization against Node CPU |
| cpu_total_milli | - | millicores | CpuTotUsage |
| | | | Pod CPU Usage (millicore) |
| | | | Pod CPU usage |
| cpu_user | - | percent | CpuUserByNode |
| | | | Pod CPU User Usage by Node (%) |
| | | | Pod CPU User Utilization against Node CPU |
| cpu_request | - | millicores | CpuRequest |
| | | | Pod CPU Request (core) |
| | | | Pod CPU Request |
| cpu_per_request | - | percent | CpuByRequest |

| Field | Type | Unit | Shortname, Name, Description |
|----------------------|------|---------|---|
| | | | Pod CPU Usage by Request (%) |
| | | | Utilization against Pod CPU Request
= $\text{cpu_total_milli}/\text{cpu_request} * 100$ |
| mem_totalcache | - | byte | MemTotCache |
| | | | Pod Memory Total Cache (byte) |
| | | | Total Pod Cache Size |
| mem_totalpgfault | - | count | MemTotPageFaultCnt |
| | | | Pod Memory Total Page Fault Count |
| | | | Pod's Page Fault Count |
| mem_totalrss | - | byte | MemTotRss |
| | | | Pod Memory Total RSS (byte) |
| | | | Pod's Total RSS Memory Size |
| mem_totalrss_percent | - | percent | MemTotRssByLimit |
| | | | Pod Memory Total RSS by Limit (%) |
| | | | Pod's Total RSS Memory Utilization |
| mem_totalunevictable | - | byte | MemTotUnevictable |
| | | | Pod Memory Total Unevictable (byte) |
| | | | Pod's Total Unevictable Memory Size |
| mem_usage | - | byte | MemUsage |

| Field | Type | Unit | Shortname, Name, Description |
|--------------------|------|---------|---|
| | | | Pod Memory Usage (byte) |
| | | | Pod Memory Usage |
| mem_working_set | - | byte | MemWs |
| | | | Pod Memory Working Set (byte) |
| | | | Pod Memory working set
= mem_usage - inactive file |
| memory_request | - | byte | MemRequest |
| | | | Pod Memory Request (byte) |
| | | | Pod memory requests |
| memory_limit | - | byte | MemLimit |
| | | | Pod Memory Limit (byte) |
| | | | Pod memory limit quota |
| memory_per_request | - | percent | MemByRequest |
| | | | Pod Memory Working Set by Request (%) |
| | | | Working Set usage based on the Pod memory request |
| memory_per_limit | - | percent | MemByLimit |
| | | | Pod Memory Working Set by Limit (%) |
| | | | Working Set usage based on the Pod memory limit |
| network_rbps | - | byte | NetRxBytes |

| Field | Type | Unit | Shortname, Name, Description |
|------------------|------|-------|--|
| | | | Pod Network Receive Byte |
| | | | Sum of bytes read per second across all block devices in the Pod |
| network_rdropped | - | byte | NetRxDropped |
| | | | Pod Network Receive Dropped |
| | | | Pod Network Receive Dropped Count |
| network_rerror | - | byte | NetRxError |
| | | | Pod Network Receive Error |
| | | | Pod's network receive error count |
| network_riops | - | byte | NetRxiops |
| | | | Pod Network Receive IOPS |
| | | | Pod Network Receive Count |
| network_wbps | - | byte | NetTxBytes |
| | | | Pod Network Transmit Byte |
| | | | Pod Network Transmit Data Size |
| network_wdropped | - | count | NetTxDropped |
| | | | Pod Network Transmit Dropped |
| | | | Pod Network Transmit Dropped Count |
| network_werror | - | count | NetTxError |

| Field | Type | Unit | Shortname, Name, Description |
|---------------|--------|-------|---|
| | | | Pod Network Transmit Error |
| | | | Pod Network Transmit Error Count |
| network_wiops | - | count | NetTxIops |
| | | | Pod Network Transmit IOPS |
| | | | Pod Network Transmit Count |
| phase | string | - | Phase |
| | | | Pod Current Phase |
| | | | Pod lifecycle
① PENDING
② RUNNING
③ SUCCEEDED
④ FAILED
⑤ UNKNOWN |

The following fields are reserved for internal use.

| Field | Type | Unit | Description |
|----------------------|------|------|--|
| kube_sless_normal | - | - | Number of Kubernetes informative events |
| kube_sless_warning | - | - | Number of Kubernetes warning events |
| micro_sful_critical | - | - | Number of APM events that are critical |
| micro_sful_info | - | - | APM informative event count |
| micro_sful_warning | - | - | APM warning event count |
| micro_sless_critical | - | - | Number of APM events that are not critical |

| Field | Type | Unit | Description |
|---------------------|------|------|---|
| micro_sless_info | - | - | Number of APM events that are not informative |
| micro_sless_warning | - | - | Number of APM events that are not for warning |
| sful_critical | - | - | Number of events that are critical in the metric |
| sful_info | - | - | Number of events that are informative in the metric |
| sful_warning | - | - | Number of events that are for warning in the metric |
| sless_critical | - | - | Number of events that are not critical in the metric |
| sless_info | - | - | Number of events that are not informative in the metric |
| sless_warning | - | - | Number of events that are not for warning in the metric |

The GPU Request and Limit quantities of a Pod are collected as parallel list fields by device type. `*_types` and `*_counts` correspond 1:1 by index, and hold `nvidia.com/gpu` (whole-GPU) and all MIG profiles (`nvidia.com/mig-*`) as individual entries.

| Field | Type | Unit | Description |
|--------------------|------|-------|--|
| gpu_request_types | list | - | List of device types of the Pod GPU Request |
| gpu_request_counts | list | count | Quantity per type above for Request (1:1 index correspondence) |
| gpu_limit_types | list | - | List of device types of the Pod GPU Limit |
| gpu_limit_counts | list | count | Quantity per type above for Limit (1:1 index correspondence) |

ⓘ Starting from `v1.9.14`, the whole-GPU-only scalar fields `gpu_request` and `gpu_limit` have been removed. This prevents the problem where, in MIG environments, the scalar fields hold only `nvidia.com/gpu` and omit MIG slices, making the aggregation a false total. The whole-GPU quantity is also checked from the `nvidia.com/gpu` entry in the list, and the total and per-type aggregations are

! the sum of *_counts .

Kubernetes Pod Statistics (kube_pod_stat) metrics

The kube_pod_stat category cluster project collects data for all clusters, and the namespace project collects data only for pods that belong to the namespace.

- Target: Cluster project, Namespace project
- Collection interval: 5 seconds
- Statistical data: 5 minutes, 1 hour

Tags

| Tag | Type | Unit | Description |
|------|------|------|--|
| kind | - | - | Type
- A cluster project has its fixed value, and a namespace project collects only the deployment or ReplicaSet. |
| name | - | - | Kubernetes resource name
- A cluster project has no name value and a namespace project has the name for Deployment or ReplicaSet. |

Fields

| Field | Type | Unit | Description |
|---------------|---------|------|---|
| available_pod | integer | - | Number of pods whose phase is in Running state |
| desired_pod | integer | - | Sum of the number of pods deployed without metadata.ownerReferences and the number of desired pods defined in Kubernetes objects (ReplicaSet, Daemonset, StatefulSet) |

| Field | Type | Unit | Description |
|-------------------|---------|------------|--|
| | | | Same as the number of pods retrieved by <code>kubectl get pods -A</code> |
| limit_cpu | - | millicores | CPU Limit Usage |
| limit_memory | - | byte | Memory Limit Usage |
| request_cpu | - | millicores | CPU Request Usage |
| request_memory | - | byte | Memory Request Usage |
| running_container | integer | - | Running Container Count |
| stopped_container | integer | - | Stopped Container Count |
| waiting_container | integer | - | Waiting container count |

Kubernetes Pod volume (`kube_pod_volume`) metrics

The `kube_pod_volume` category collects the capacity and i-node usage of volumes mounted on Pods running on a node. Since the directory name of an app Pod's PVC volume is the PV name, you can join it with the PVC's `spec.volumeName` using the `volumeName` tag.

Tags

| Tag | Type | Unit | Description |
|----------------|------|------|---|
| hostDevice | - | - | Host device where the volume is located |
| hostFileSystem | - | - | File system of the volume |
| hostMountPath | - | - | Volume mount path on the host |

| Tag | Type | Unit | Description |
|--------------|------|------|--|
| volumeName | - | - | Volume name
- For a PVC volume, the PV name (e.g. <code>pvc-...</code>). Can be joined with the PVC's <code>spec.volumeName</code> |
| volumePlugin | - | - | Volume plugin
- e.g. <code>kubernetes.io~csi</code> , <code>kubernetes.io~nfs</code> , <code>volume-subpaths</code> |

Fields

| Field | Type | Unit | Description |
|------------------|-------|---------|--|
| capacity | long | byte | Total volume capacity |
| available | long | byte | Available volume capacity |
| availablePercent | float | percent | Available volume capacity ratio |
| usedPercent | float | percent | Volume usage |
| free | long | byte | Free volume capacity |
| freePercent | float | percent | Free volume capacity ratio |
| totalInode | long | count | Total number of volume i-nodes |
| usedInode | long | count | Number of i-nodes in use in the volume |
| usedInodePercent | float | percent | Volume i-node usage |

`emptyDir` volume collection has been added. A default (disk) medium `emptyDir` is a directory of the node root file system rather than a separate mount, so it was not caught by the `df`-based collection. The node agent queries the kubelet volume directory directly and sends the data to the same `kube_pod_volume` category.

- Query path: `{HOST_PREFIX}{kubelet_root_dir}/pods/{podUid}/volumes/kubernetes.io~empty-dir/{volumeName}`

- Collection interval: 60 seconds. Because querying directories is expensive, it is separated from the 5-second collection interval.
- The `volumePlugin` tag value of an `emptyDir` volume is `kubernetes.io~empty-dir`.

The following tags have been added.

| Tag | Type | Unit | Description |
|------------------------------------|---------------------|------|---|
| <code>volumeType</code> | <code>string</code> | - | Volume type
- <code>EmptyDir</code> for an <code>emptyDir</code> volume |
| <code>emptyDirMedium</code> | <code>string</code> | - | Storage medium of the <code>emptyDir</code>
- <code>Memory</code> or <code>Disk</code> . Based on <code>emptyDir.medium</code> of the Pod spec; unspecified is <code>Disk</code> |
| <code>emptyDirSpecConfirmed</code> | <code>string</code> | - | Whether the Pod spec was checked
- <code>true</code> is a value confirmed by the spec, <code>false</code> is a value estimated from the kubelet path and the file system type |

The following fields have been added.

| Field | Type | Unit | Description |
|------------------------|----------------------|--------------------|--|
| <code>usedSpace</code> | <code>long</code> | <code>byte</code> | Total volume usage |
| <code>fileCount</code> | <code>long</code> | <code>count</code> | Number of files in the volume |
| <code>sizeLimit</code> | <code>long</code> | <code>byte</code> | <code>emptyDir.sizeLimit</code> of the Pod spec
- Not sent when it is not set |
| <code>truncated</code> | <code>integer</code> | - | Whether the collection limit was exceeded
- <code>1</code> means the value is smaller than the actual one because the limit was hit |

`capacity`, `available`, and `usedPercent` are sent only when the volume capacity can be determined.

- When `sizeLimit` exists, that value is used as the volume capacity.
- When there is no `sizeLimit` and the medium is `Memory`, the total tmpfs size created by kubelet is used as

the capacity. The default total tmpfs size is half of the node memory.

- A `Disk` medium without `sizeLimit` has no upper limit, so no capacity is created. In this case, compare it with the Pod's `container.ephemeral_storage_limit`.

All of the tags and fields above are sent on node agent 1.9.23 or later. For the collection activation and cost limit options, see [Configuring the options for the Kubernetes agent](#).

Kubernetes Horizontal Pod Autoscaler (HPA) (`kube_hpa_stat`) metrics

Metric collection starts only when HPA is added to the ClusterRole used by WhaTap.

- Target: Cluster project
- Collection interval: 5 seconds
- Statistical data: 5 minutes, 1 hour

Tags

| Tag | Type | Unit | Description |
|------|------|------|-------------|
| name | - | - | HPA name |

The following tags have been added.

| Tag | Type | Unit | Description |
|------------------------------------|---------------------|------|---|
| <code>scaleTargetKind</code> | <code>string</code> | - | Kind of the scale target object (<code>spec.scaleTargetRef.kind</code>) |
| <code>scaleTargetName</code> | <code>string</code> | - | Name of the scale target object (<code>spec.scaleTargetRef.name</code>) |
| <code>scaleTargetApiVersion</code> | <code>string</code> | - | API version of the scale target object |
| <code>creationTimeMillis</code> | <code>long</code> | - | HPA creation time (epoch ms)
- Not sent when there is no value |

The scale target of an HPA is in `spec.scaleTargetRef`. The `owner_kind`, `owner_name`, and `owner_id` tags based on `ownerReferences` are mostly empty for an HPA, so check the scale target with the `scaleTarget*` tags. The existing tags are kept for backward compatibility.

Collected on master agent 1.9.24 or later.

Fields

| Field | Type | Unit | Description |
|-----------------|---------|-------|-----------------------|
| currentReplicas | integer | count | Current Replica Count |
| desiredReplicas | integer | count | Desired Replica Count |
| lastScaleTime | integer | count | Last scaled TimeStamp |
| maxReplicas | integer | count | Maximum Replica Count |
| minReplicas | integer | count | Minimum Replica Count |

Kubernetes HPA list (`kube_hpa`) metrics

The `kube_hpa` category collects HPA list data equivalent to the result of `kubectl get hpa -o wide`. It is a separate category from `kube_hpa_stat`, which is a time series metric.

- Target: Cluster project
- Collection interval: 30 seconds
- Required version: Master agent 1.9.24 or later

Tags

| Tag | Type | Unit | Description |
|-----------------------|--------|------|--|
| hpaUid | string | - | HPA UID (unique value) |
| hpaName | string | - | HPA name |
| namespace | string | - | Namespace the HPA belongs to |
| creationTimeMillis | long | - | HPA creation time (epoch ms) |
| scaleTargetKind | string | - | Kind of the scale target object |
| scaleTargetName | string | - | Name of the scale target object |
| scaleTargetApiVersion | string | - | API version of the scale target object |

Fields

| Field | Type | Unit | Description |
|------------------|---------|-------|---|
| minReplicas | integer | count | Minimum replica count |
| maxReplicas | integer | count | Maximum replica count |
| currentReplicas | integer | count | Current replica count |
| desiredReplicas | integer | count | Desired replica count |
| targetMetricType | list | - | List of metric types used as the scaling criteria |
| targetValue | list | - | List of target values per metric |
| currentValue | list | - | List of current values per metric |

| Field | Type | Unit | Description |
|--------------------------------------|--------|------|---|
| <code>conditionAbleToScale</code> | string | - | Status of the <code>AbleToScale</code> condition |
| <code>conditionScalingActive</code> | string | - | Status of the <code>ScalingActive</code> condition |
| <code>conditionScalingLimited</code> | string | - | Status of the <code>ScalingLimited</code> condition |

`targetMetricType` , `targetValue` , and `currentValue` are parallel lists where the same index corresponds to one metric. An HPA can have CPU, memory, and custom metrics at the same time, so it cannot be expressed with a single value.

- The values are strings in the same notation as `kubectl` (for example, `50%` , `100m`).
- When the current value cannot be obtained because metrics-server is absent, it is collected as `<unknown>` .
- Besides Resource metrics, Pods, External, and Object metrics are stored with the `pods:` , `external:` , and `object:` prefixes respectively.
- `autoscaling/v2` is used first, and clusters that do not support it fall back to `autoscaling/v1` .

The `horizontalpodautoscalers` permission is already included in the ClusterRole, so no additional permission setting is required.

Process (`kube_process`) metrics

ⓘ Kubernetes agent 1.7.12 or later is required. For more information about agent updates, see [the following](#).

Kubernetes-related processes that exist in the node are collected during monitoring.

- Target: Cluster project, Namespace project
- Collection interval: 5 seconds
- Statistical data: 5 minutes

Tags

| Tag | Type | Unit | Description |
|------------|-----------|------|--|
| ppid | string | - | Parent process ID
<code>/proc/[pid]/status::PPid</code> |
| pid | string | - | Process ID
<code>/proc/[pid]/status::Pid</code> |
| cmd1 | string | - | Command name
<code>/proc/[pid]/status::Name</code> |
| cmd2 | string | - | Command line (all commands and arguments)
<code>/proc/[pid]/cmdline</code> |
| user | string | - | User ID or user name
<code>/proc/[pid]/status::Uid</code> |
| onodeName | string | - | Node name of the process
Container system's environment variable (<code>NODE_IP</code>) |
| createTime | TimeStamp | - | Process start time
Field calculated through <code>/proc/uptime</code> |

Fields

| Field | Type | Unit | Description |
|--------|-------|------------|---|
| cpu | float | percent(%) | CPU utilization
- Field calculated through <code>/proc/[pid]/stat</code> |
| memory | float | percent(%) | Memory utilization
- Field calculated through <code>/proc/[pid]/statm</code> |
| rss | long | byte | Actual memory usage (Resident Set Size) |

| Field | Type | Unit | Description |
|---------------------|---------|------|--|
| | | | - VmRSS of <code>/proc/[pid]/status</code> |
| uid | string | - | User ID or name
- Uid of <code>/proc/[pid]/status</code> |
| state | string | - | Process state
- State of <code>/proc/[pid]/status</code> |
| sharedMemory | long | byte | Shared memory size
- Field calculated through <code>/proc/[pid]/statm</code> |
| openFileDescriptors | integer | - | Number of file descriptors that the process has open
- Field calculated through <code>/proc/[pid]/fd</code> |
| vmSize | long | byte | Virtual memory size
- VmSize of <code>/proc/[pid]/status</code> |
| threads | integer | - | Number of threads created by processes
- Threads of <code>/proc/[pid]/status</code> |

Linux process status in the Kubernetes environment

On Linux, the `State` field in the `/proc/[pid]/status` file displays the current state of the process. The meanings of each status are as follows:

| Code | Description |
|----------------|---|
| R (Running) | The process is running or ready to run. |
| S (Sleeping) | Interruptible sleep state, waiting for an event. |
| D (Disk Sleep) | Non-interruptible sleep state, waiting for an I/O operation. |
| R (Zombie) | The process has been terminated, but the parent process has not yet collected its termination status. |

| Code | Description |
|------------------|--|
| T (Stopped) | The process is stopped by a job control signal (such as SIGSTOP) or debugger. |
| t (Tracing stop) | Tracing stopped
- State being traced by the debugger (indicated by a lowercase t) |
| X (Dead) | Dead state
- The process is dead (commonly unseen) |
| x (Dead) | Dead state
- The kernel thread is dead (commonly unseen) |
| K (WakeKill) | Forcibly terminated
- It ignores any wakeup signal and is immediately dead. |
| W (Waking) | Waking up
- State of being woken up after receiving a wake-up signal |
| I (Idle) | Kernel thread is idle (usually invisible to user space processes). |

❗ Because Kubernetes manages the resources of containers and nodes efficiently, many processes running in containers are actually in waiting state. As a result, most processes may be in `Sleeping` state.

Agent status (`agent_status_summary`) metrics

This category collects metrics related to agent status every 10 seconds.

Fields

| Field | Type | Unit | Description |
|--------------------------|---------|-----------------|---|
| <code>inActTime</code> | - | millisecond(ms) | Amount of time the agent remains inactive |
| <code>isActive</code> | Boolean | - | Whether the current agent is active or not (<code>true</code> / <code>false</code>) |
| <code>isRestart</code> | Boolean | - | Whether the agent was restarted (<code>true</code> / <code>false</code>) |
| <code>lastActTime</code> | - | millisecond(ms) | Last time when the agent was activated
- <code>0</code> : Disabled |
| <code>oid</code> | - | - | Unique IDs for each agent in the project |
| <code>oType</code> | - | - | Agent type
- <code>1</code> : Application agent
- <code>2</code> : See <code>subType</code> |
| <code>startTime</code> | - | millisecond(ms) | Timestamp indicating the time when the agent was started |
| <code>subType</code> | - | - | Agent type
- <code>9</code> : Node agent
- <code>10</code> : Master agent |

Ingress (`kube_ingress`) metrics

ⓘ Kubernetes agent 1.7.13 or later is required. For more information about agent updates, see [the following](#).

It is collected when monitoring metadata and the related information for Ingress resources.

- Target: Cluster project, Namespace project

- Collection interval: 30 seconds
- Statistical data: 5 minutes

Tags

| Tag | Type | Unit | Description |
|-------------------------------------|--------|-----------------|--------------------------------------|
| <code>ingressUid</code> | string | - | Unique ID of the Ingress resource |
| <code>ingressName</code> | string | - | Name of the Ingress resource |
| <code>ingressNamespace</code> | string | - | Namespace of the Ingress resource |
| <code>creationTimeMillis</code> | long | millisecond(ms) | Created time of the Ingress resource |
| <code>ingressClassName</code> | string | - | Name of the Ingress class |
| <code>ingressLoadBalancerIps</code> | List | - | IP of the Ingress load balancer |

Fields

| Field | Type | Unit | Description |
|---------------------------------|------|------|--|
| <code>host</code> | List | - | Host name that the Ingress resource listens to (if <code>*</code> , it applies to all hosts) |
| <code>path</code> | List | - | Request path under a specific host |
| <code>backendServiceName</code> | List | - | Name of the service passed to the backend |
| <code>backendServicePort</code> | List | - | Port number passed to the backend |
| <code>backendServiceUid</code> | List | - | URL of the service passed to the backend |
| <code>pathType</code> | List | - | Path matching method (e.g. <code>Prefix</code> , <code>Exact</code>) |

Kubernetes cron job (kube_cronjob) metrics

It collects metadata and status information for deployment resources by using the tags.

Tags

| Tag | Type | Unit | Description |
|-------------|--------|------|--|
| cronJobName | string | - | CronJob name (metadata.name) |
| namespace | string | - | CronJob namespace (metadata.namespace) |
| cronJobUid | string | - | CronJob UID (metadata.uid) |

Fields

| Field | Type | Unit | Description |
|----------------------------|---------|-------|---|
| lastScheduleTime | string | - | Last time when the CronJob was scheduled (status.lastScheduleTime) |
| lastSuccessfulTime | string | - | Last time when the CronJob succeeded (status.lastSuccessfulTime) |
| cronJobSpecSchedule | string | - | Scheduling cycle of the CronJob (spec.schedule , e.g.: "0 */5 * * *") |
| successfulJobsHistoryLimit | integer | count | Maximum number of successful jobs to keep (spec.successfulJobsHistoryLimit) |
| failedJobsHistoryLimit | integer | count | Maximum number of failed jobs to keep (spec.failedJobsHistoryLimit) |
| concurrencyPolicy | string | - | Concurrency policy (Allow , Forbid , Replace) (spec.concurrencyPolicy) |

| Field | Type | Unit | Description |
|--------------------------------------|---------|--------|---|
| <code>startingDeadlineSeconds</code> | integer | second | Maximum delay time when the job can be run after its scheduled time (<code>spec.startingDeadlineSeconds</code>) |

Kubernetes deployment (`kube_deployment`) metrics

It collects metadata and status information for deployment resources by using the tags.

Tags

| Tag | Type | Unit | Description |
|-------------------------|--------|------|--|
| <code>deployName</code> | string | - | Deployment name (<code>metadata.name</code>) |
| <code>namespace</code> | string | - | Deployment namespace (<code>metadata.namespace</code>) |
| <code>deployUid</code> | string | - | Deployment UID (<code>metadata.uid</code>) |

Fields

| Field | Type | Unit | Description |
|--------------------------------------|---------|-------|--|
| <code>deployReadyReplicas</code> | integer | count | Number of Pods that are in ready status (<code>status.readyReplicas</code>) |
| <code>deployTotalReplicas</code> | integer | count | Total number of set Pods (<code>spec.replicas</code>) |
| <code>deployUpdatedReplicas</code> | integer | count | Number of Pods updated to the latest version (<code>status.updatedReplicas</code>) |
| <code>deployAvailableReplicas</code> | integer | count | Number of Pods in available state (<code>status.availableReplicas</code>) |

| Field | Type | Unit | Description |
|--|---------|-----------------|---|
| <code>deployUnavailableReplicas</code> | integer | count | Number of Pods in unavailable state (<code>status.unavailableReplicas</code>) |
| <code>deployCreationTime</code> | integer | millisecond(ms) | Deployment creation time (Unix Epoch milliseconds, <code>metadata.creationTimestamp</code>) |
| <code>deploySelector</code> | string | - | Selector label of the deployment (<code>spec.selector.matchLabels</code>) |
| <code>deployStrategy</code> | string | - | Deployment strategy (<code>spec.strategy.type</code> , e.g. <code>RollingUpdate</code> , <code>Recreate</code>) |

Kubernetes daemonset (`kube_daemonset`) metrics

It collects metadata and status information for daemonset resources by using the tags.

Tags

| Tag | Type | Unit | Description |
|----------------------------|---------|-----------------|---|
| <code>daemonSetName</code> | string | - | DaemonSet name (<code>metadata.name</code>) |
| <code>namespace</code> | string | - | DaemonSet namespace (<code>metadata.namespace</code>) |
| <code>daemonSetUid</code> | string | - | DaemonSet UID (<code>metadata.uid</code>) |
| <code>creationTime</code> | integer | millisecond(ms) | DaemonSet creation time (<code>metadata.creationTimestamp</code>) |
| <code>age</code> | string | - | Elapsed time since DaemonSet creation |

Fields

| Field | Type | Unit | Description |
|-------------------------------------|---------|-------|--|
| <code>currentNumberScheduled</code> | integer | count | Number of scheduled Pods (<code>status.currentNumberScheduled</code>) |
| <code>desiredNumberScheduled</code> | integer | count | Number of desired Pods (<code>status.desiredNumberScheduled</code>) |
| <code>numberReady</code> | integer | count | Number of Pods in ready state (<code>status.numberReady</code>) |
| <code>numberAvailable</code> | integer | count | Number of available Pods (<code>status.numberAvailable</code>) |
| <code>numberMisscheduled</code> | integer | count | Number of wrongly scheduled Pods
(<code>status.numberMisscheduled</code>) |
| <code>updatedNumberScheduled</code> | integer | count | Number of updated Pods (<code>status.updatedNumberScheduled</code>) |
| <code>daemonSetSelector</code> | string | - | Selector label (<code>spec.selector.matchLabels</code>) |

Kubernetes replica set (`kube_replicaset`) metrics

It collects metadata and status information for ReplicaSet resources with the tags.

Tags

| Tag | Type | Unit | Description |
|-----------------------------|--------|------|--|
| <code>replicaSetName</code> | string | - | ReplicaSet name (<code>metadata.name</code>) |
| <code>namespace</code> | string | - | ReplicaSet namespace (<code>metadata.namespace</code>) |
| <code>replicaSetUid</code> | string | - | ReplicaSet UID (<code>metadata.uid</code>) |

| Tag | Type | Unit | Description |
|--------------|---------|-----------------|--|
| ownerKind | string | - | Type of the upper layer object (e.g. Deployment)
(metadata.ownerReferences.kind) |
| ownerName | string | - | Name of the upper layer object
(metadata.ownerReferences.name) |
| ownerUid | string | - | UID of the upper layer object (metadata.ownerReferences.uid) |
| creationTime | integer | millisecond(ms) | ReplicaSet creation time (metadata.creationTimestamp) |
| age | string | - | Elapsed time since creation of ReplicaSet |

Fields

| Field | Type | Unit | Description |
|-----------------------------|---------|-------|--|
| replicaSetReplicas | integer | count | Number of set Pods (spec.replicas) |
| replicaSetReadyReplicas | integer | count | Number of Pods in ready state (status.readyReplicas) |
| replicaSetAvailableReplicas | integer | count | Number of available Pods (status.availableReplicas) |
| replicaSetSelector | string | - | Selector label (spec.selector.matchLabels) |

Kubernetes stateful set (kube_statefulSet) metrics

It collects metadata and status information for Stateful resources with the tags.

Tags

| Tag | Type | Unit | Description |
|-----------------|---------|-----------------|--|
| statefulSetName | string | - | Name of StatefulSet (<code>metadata.name</code>) |
| namespace | string | - | Namespace to which the StatefulSet belongs (<code>metadata.namespace</code>) |
| statefulSetUid | string | - | UIO of StatefulSet (<code>metadata.uid</code>) |
| creationTime | integer | millisecond(ms) | StatefulSet creation time (<code>metadata.creationTimestamp</code>) |
| age | string | - | Elapsed time since StatefulSet creation |

Fields

| Field | Type | Unit | Description |
|---------------------|---------|-------|---|
| replicas | integer | count | Number of set Pods (<code>spec.replicas</code>) |
| readyReplicas | integer | count | Number of Pods in ready state (<code>status.readyReplicas</code>) |
| currentReplicas | integer | count | Number of created Pods (<code>status.currentReplicas</code>) |
| updatedReplicas | integer | count | Number of updated Pods (<code>status.updatedReplicas</code>) |
| statefulSetSelector | string | - | Selector label (<code>spec.selector.matchLabels</code>) |

Kubernetes job (`kube_job`) metrics

It collects metadata and status information for deployment resources by using the tags.

Tags

| Tag | Type | Unit | Description |
|------------------------|--------|------|--|
| <code>jobName</code> | string | - | Job name (<code>metadata.name</code>) |
| <code>namespace</code> | string | - | Namespace of the Job (<code>metadata.namespace</code>) |
| <code>jobUid</code> | string | - | Job UID (<code>metadata.uid</code>) |
| <code>ownerKind</code> | string | - | Parent object kind (e.g. <code>CronJob</code>) |
| <code>ownerName</code> | string | - | Parent object name |
| <code>ownerUid</code> | string | - | Parent object UID |

Fields

| Field | Type | Unit | Description |
|------------------------------------|---------|-------------|---|
| <code>succeededPods</code> | integer | count | Number of successful jobs (<code>status.succeeded</code>) |
| <code>failedPods</code> | integer | count | Number of failed Pods (<code>status.failed</code>) |
| <code>completionTime</code> | string | - | Job completion time (<code>status.completionTime.toString()</code>) |
| <code>parallelism</code> | integer | count | Number of Pods set to run concurrently (<code>spec.parallelism</code>) |
| <code>completions</code> | integer | count | Number of Pod executions required for a job to be completed (<code>spec.completions</code>) |
| <code>backoffLimit</code> | integer | count | Maximum number of possible retries upon failure (<code>spec.backoffLimit</code>) |
| <code>activeDeadlineSeconds</code> | integer | second(sec) | Maximum time for a job to run (<code>spec.activeDeadlineSeconds</code>) |

Kubernetes NetworkPolicy (kube_network_policy) metrics

The kube_network_policy category collects the NetworkPolicy objects of the cluster on an event basis. Instead of periodic collection, a snapshot of that point in time is sent whenever an object is added, changed, or deleted.

- Target: Cluster project, namespace project
- Collection method: Event based

Tags

| Tag | Type | Unit | Description |
|-------------------|--------|------|--|
| networkPolicyUid | string | - | NetworkPolicy UID (unique value) |
| networkPolicyName | string | - | NetworkPolicy name |
| namespace | string | - | Namespace the NetworkPolicy belongs to |
| resourceVersion | string | - | Object revision (metadata.resourceVersion) |

Fields

| Field | Type | Unit | Description |
|--------------------|------|------|--|
| generation | long | - | Object generation (metadata.generation) |
| creationTimeMillis | long | - | Object creation time (epoch ms) |
| policyTypes | list | - | List of policy types
- Ingress , Egress . An empty list means Ingress , the API default |
| selectedPodUids | list | - | List of Pod UIDs selected by the policy |
| podSelector | map | - | Pod selector of the policy target (matchLabels , matchExpressions) |

| Field | Type | Unit | Description |
|--------------------------------|----------------------|------|---|
| <code>ingressRules</code> | <code>list</code> | - | List of Ingress rules
- Includes <code>from</code> and <code>ports</code> per rule |
| <code>egressRules</code> | <code>list</code> | - | List of Egress rules
- Includes <code>to</code> and <code>ports</code> per rule |
| <code>specSchemaVersion</code> | <code>integer</code> | - | Schema version of the spec fields |
| <code>specHash</code> | <code>long</code> | - | Hash of the normalized spec
- Used to determine whether the content changed |

The rules are sent in a nested structure to preserve the meaning of the Kubernetes API as it is.

- `namespaceSelector` and `podSelector` within one peer are in an AND relationship.
- The peer list and the rule list are in an OR relationship.
- An empty `from` or `to` in a rule means all peers, and empty `ports` means all ports.
- The `ipBlock` of a peer contains `cidr` and `except`.
- A port contains `protocol`, `portType`, `portNumber` or `portName`, and `endPort`.

The `networkpolicies` view permission is required in the ClusterRole. You can turn the collection off with the `collect_network_policy_enabled` option.

GPU Metrics

The following NVIDIA GPU metrics are collected through DCGM Exporter. For details about how GPU metrics are collected and the sidebar pattern, see the [Supported environment > NVIDIA GPU](#) document.

| Metric | Type | Unit | Description |
|---------------------------------|---------|-------|--|
| DCGM_FI_DEV_GPU_UTIL | gauge | % | Percentage of time the GPU cores were actively performing computation. |
| DCGM_FI_PROF_GR_ENGINE_ACTIVE | gauge | ratio | Fraction of time the graphics engine (3D/Compute) was active. |
| DCGM_FI_PROF_SM_ACTIVE | gauge | ratio | Fraction of cycles in which one or more warps were active. |
| DCGM_FI_PROF_SM_OCCUPANCY | gauge | ratio | SM (Streaming Multiprocessor) occupancy (resident warps per SM). |
| DCGM_FI_PROF_PIPE_TENSOR_ACTIVE | gauge | ratio | Fraction of cycles in which the tensor core (HMMA) pipe was active. |
| DCGM_FI_PROF_DRAM_ACTIVE | gauge | ratio | Fraction of cycles in which the memory interface was active (sending or receiving data). |
| DCGM_FI_DEV_FB_USED | gauge | MiB | Used framebuffer (VRAM) memory. |
| DCGM_FI_DEV_FB_FREE | gauge | MiB | Available framebuffer (VRAM) memory. |
| DCGM_FI_DEV_FB_TOTAL | gauge | MiB | Total framebuffer (VRAM) memory. |
| DCGM_FI_DEV_FB_USED_PERCENT | gauge | % | Percentage of used framebuffer (VRAM) memory. |
| DCGM_FI_DEV_FB_RESERVED | gauge | MiB | Reserved framebuffer memory. |
| DCGM_FI_PROF_PCIE_TX_BYTES | counter | byte | Total bytes transmitted (Host → Device) over the |

| Metric | Type | Unit | Description |
|-------------------------------|---------|---------|---|
| | | | PCIe bus. |
| DCGM_FI_PROF_PCIE_RX_BYTES | counter | byte | Total bytes received (Device → Host) over the PCIe bus. |
| DCGM_FI_DEV_POWER_USAGE | gauge | W | Current power consumption of the GPU. |
| DCGM_FI_DEV_GPU_TEMP | gauge | °C | Current temperature of the GPU chip. |
| DCGM_FI_DEV_SM_CLOCK | gauge | MHz | SM (Streaming Multiprocessor) clock frequency. |
| DCGM_FI_DEV_MEM_CLOCK | gauge | MHz | Memory clock frequency. |
| DCGM_FI_DEV_PSTATE | gauge | P-State | Current performance state of the GPU (P0 is the highest). |
| DCGM_FI_DEV_ECC_SBE_AGG_TOTAL | counter | count | Aggregated count of single-bit persistent ECC errors. |
| DCGM_FI_DEV_ECC_DBE_AGG_TOTAL | counter | count | Aggregated count of double-bit persistent ECC errors. |

The following metric has been added.

| Metric | Type | Unit | Description |
|-------------------------------|-------|-------|--|
| DCGM_FI_DEV_WEIGHTED_GPU_UTIL | gauge | ratio | GPU utilization weighted by the slice ratio the device occupies. |

This metric lets you aggregate GPU utilization of MIG and non-MIG environments on the same basis. Because a MIG profile occupies only part of a physical GPU, simply adding up utilization values does not match the GPU-level total.

The calculation method is distinguished by the `calculation_method` label.

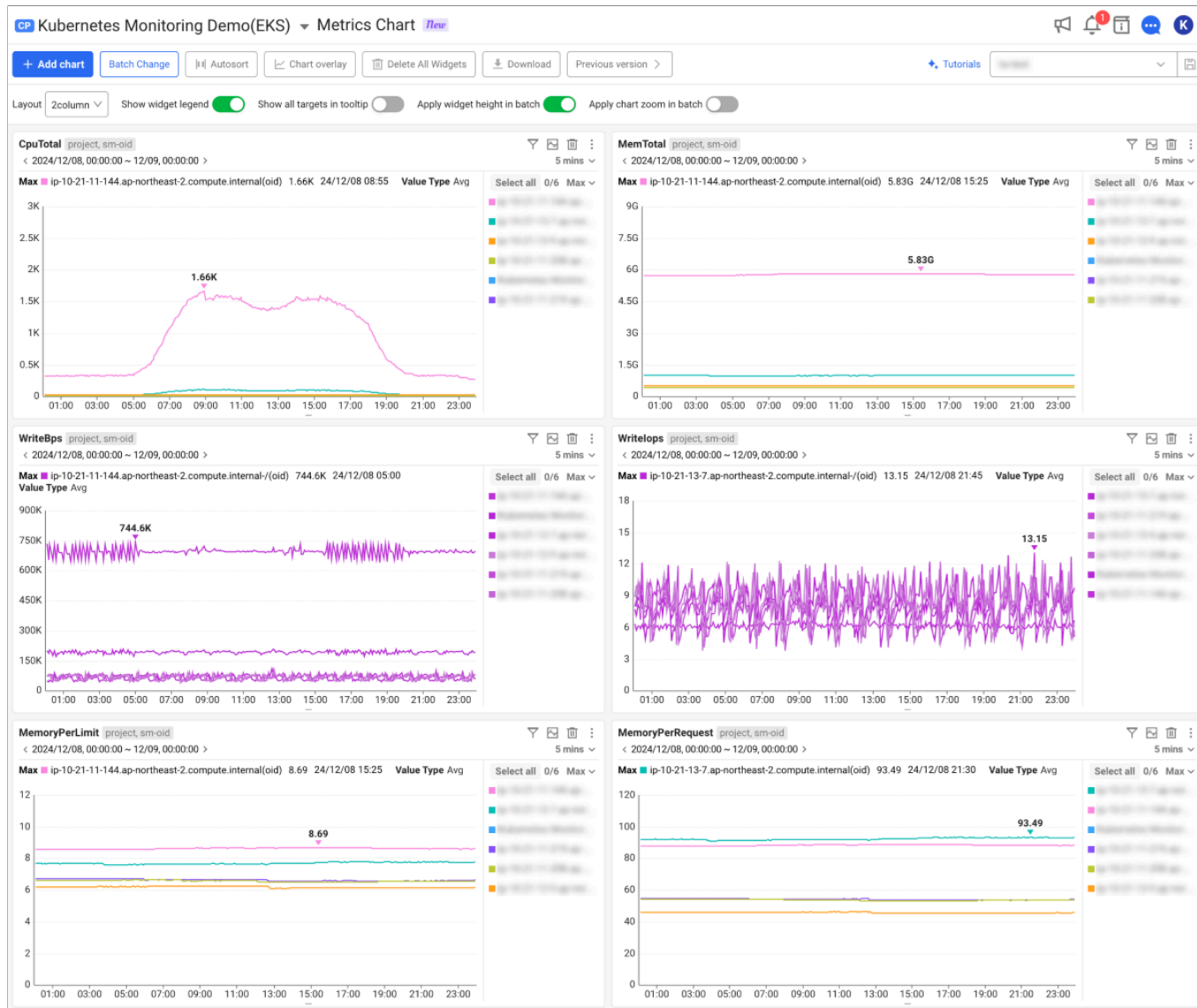
| Value | Description |
|-----------------------|---|
| weighted_sum | MIG instance. DCGM_FI_PROF_GR_ENGINE_ACTIVE multiplied by the slice ratio (compute_slices / DCGM_FI_DEV_MIG_MAX_SLICES) and then summed |
| direct | Non-MIG GPU. DCGM_FI_DEV_GPU_UTIL divided by 100 |
| prof_gr_engine_active | Non-MIG GPU of the H100, H200, H800, and H20 series.
DCGM_FI_PROF_GR_ENGINE_ACTIVE used as is |

- The value ranges from 0 to 1 .
- Abnormal DCGM_FI_DEV_GPU_UTIL values over 100 are capped at 100 before the calculation.
- The GPU UUID is provided together in the DCGM_FI_DEV_UUID label.

Metrics Chart

Home > Select Project > **Analysis** > **Metrics Chart** *New*

The **Metrics Chart** *New* menu is a core feature of the WhaTap monitoring solution, which provides the data collected from various monitoring targets in a visualized time-series chart. This allows users to effectively monitor the system performance, quickly identify anomalies, and gain insights for resolving issues.



- You can visually check various metrics via time series charts, and analyze multiple metrics upon failure at a time.
- You can select a specific time range to view past data and analyze the performance at that period.
- You can filter data based on specific time zones or conditions for more accurate analysis.
- Users can place their desired charts on the screen and freely adjust the layout.
- Users can arrange desired metrics and save them as presets for loading at any time.

! Difference from the **Metrics Chart** menu

The following are the changes since the [Service 2.8.0](#) update. To check the changes on the screen after the update, select **Tutorials** on the upper right.

- User experience (UX) has been improved so that users can set the time, target, and interval options, and then specify the metrics to monitor and add charts.
- You can change the time range of a chart view without deleting the widgets placed on the dashboard.
- It provides a batch **Batch Change** feature that allows you to apply options to all widgets placed on the dashboard at once.
- The preset feature allows you to save and load the widgets placed on the dashboard.
- You can adjust the layout of the dashboard and the functionality of the widgets through various options provided.

Adding a chart widget

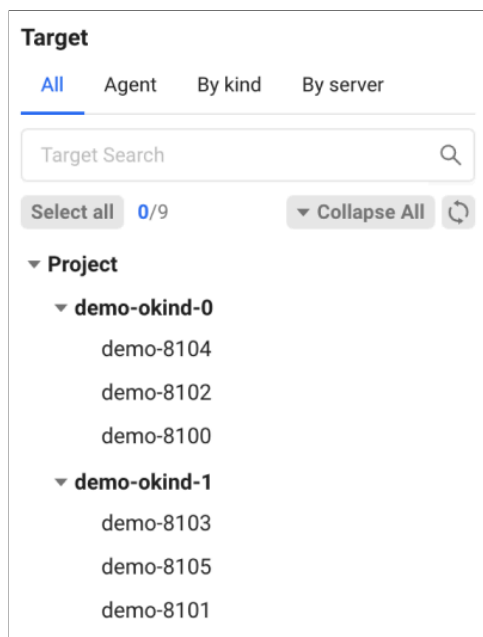
To add a metric as a widget to the dashboard, select **Add chart** on the upper left.

Setting the lookup time

In **Time**, set the date and time to view the metric.

Selecting a target

Select a desired target from the agents in the project. You can select an individual agent or a group classified by the agent configuration.

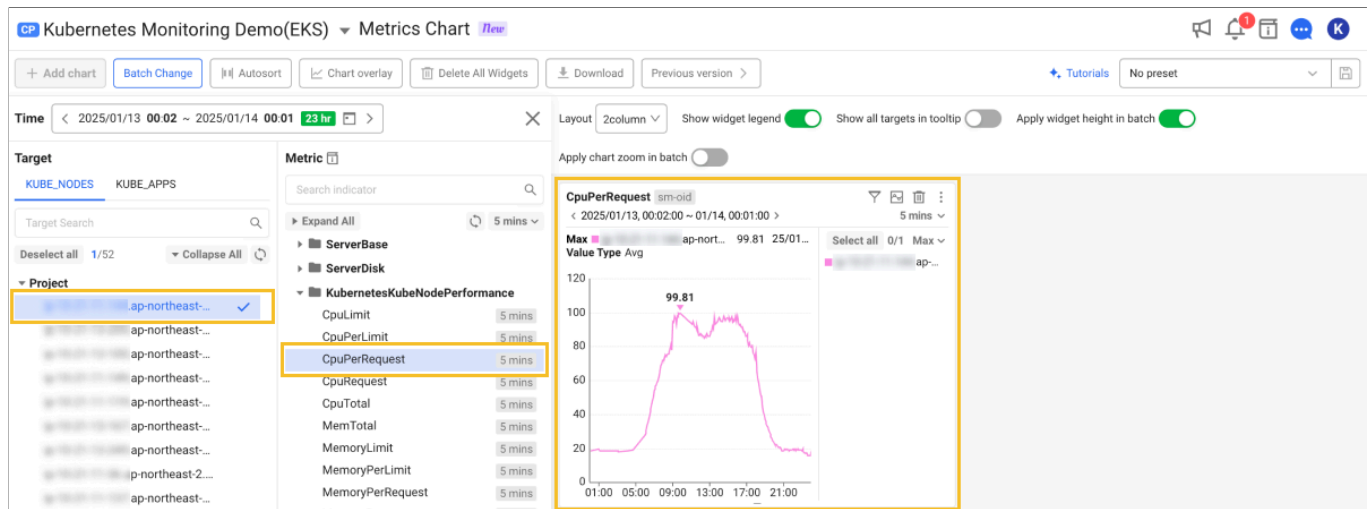


- **Agent:** You can select an individual agent or all agents.
- **By kind:** Agents can be monitored in each group classified by `whatap.okind` in the agent configuration.
- **By server:** Agents can be monitored in each group classified by `whatap.onode` in the agent configuration.

- ⓘ • Depending on the set time and date, available targets may change.
- You can select or deselect targets by clicking in the agent list. You can also select multiple agents at once by using the `Shift` key.
- By clicking **Select all** or **Deselect all**, you can select or deselect related sub-targets at once.
- If a desired agent does not appear in the list, select **Refresh**.
- By selecting **Collapse All** or **Expand All**, you can hide or expand the list by group.

Selecting a metric

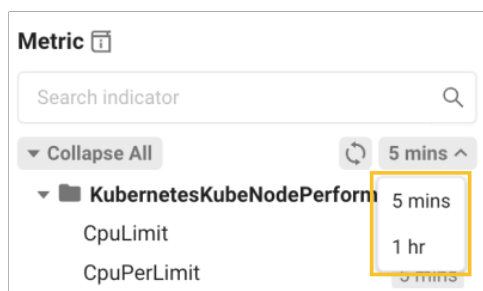
Select a target agent to monitor. Then the list that can be selected on the **Metric** panel appears. Select a desired metric in the list.



A chart widget for the selected metric is placed on the dashboard on the right. You can add chart widgets by selecting additional metrics. To enlarge the chart area, select **×** on the right of the **Time** option. The **Target** and **Metric** panels disappear, leaving only the chart area visible on the screen.

- If you enter a string in the **Search indicator** field, you can quickly search for a desired metric.
- You can also search by metric category and field name. For more information about the categories and fields, see [the following](#).
- By selecting **Collapse All** or **Expand All**, you can hide or expand the metrics list by category.
- If a desired metric does not appear in the list, select **Refresh**.

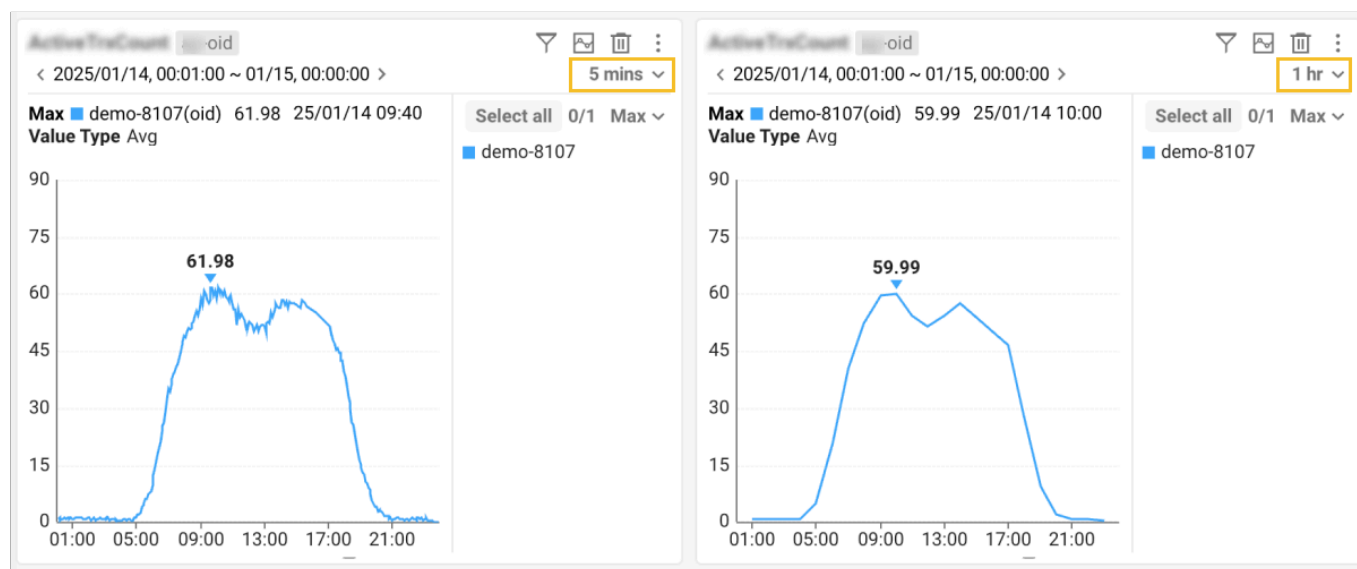
Setting the interval



You can set intervals for each chart, allowing you to adjust the aggregation unit of data to visualize at different time intervals. If you select ⌚ on the upper right of the **Metric** panel, you can select an interval option.

Each metric has a different default interval time, and the interval time changes depending on the query time range.

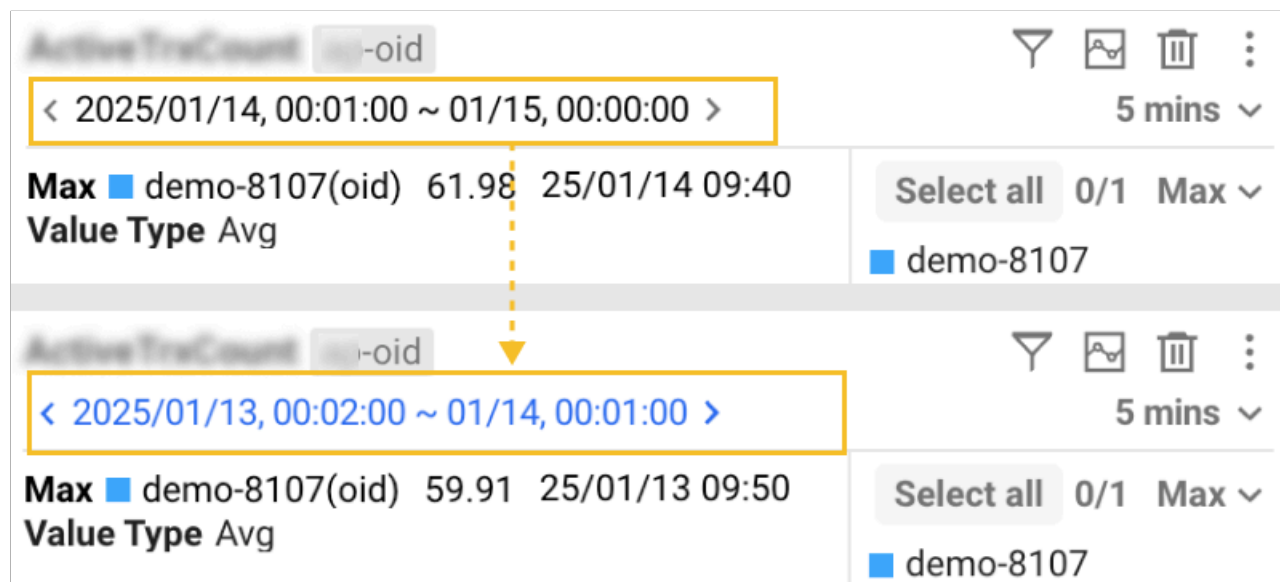
For example, the **5 min** interval displays average values for 5-minute intervals, and the **1 hour** interval displays average values for every one hour.



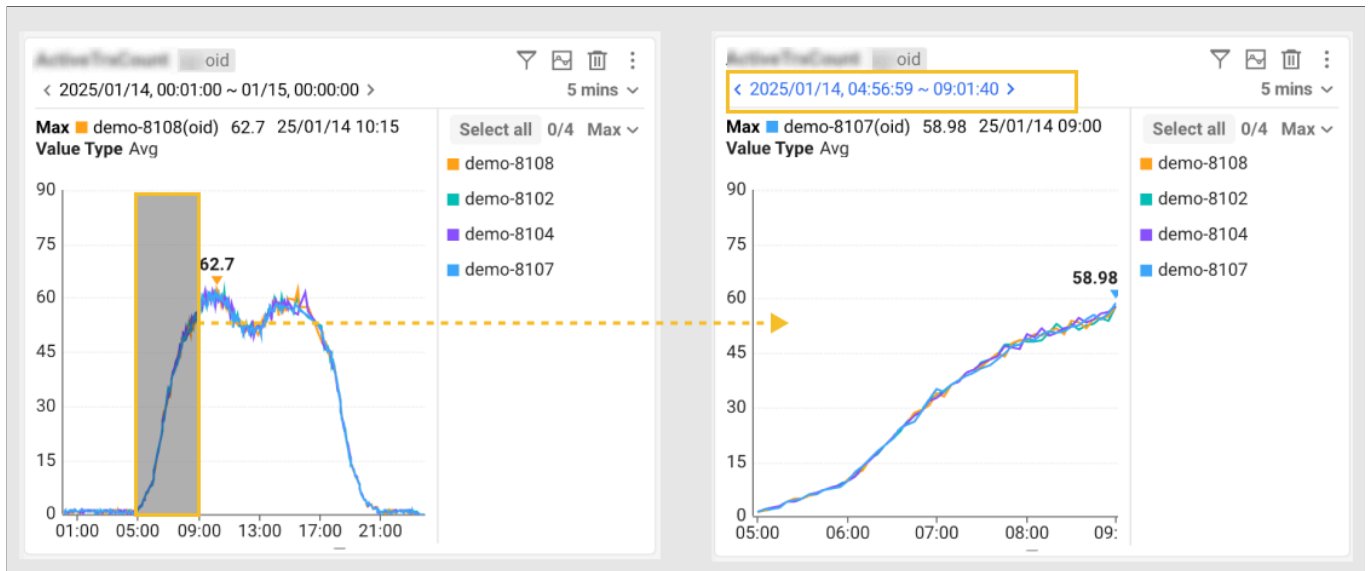
- ✓ By adjusting the interval option, you can analyze the data from different perspectives, from detailed data over a short time to overall trend over a long time.

Changing the query time

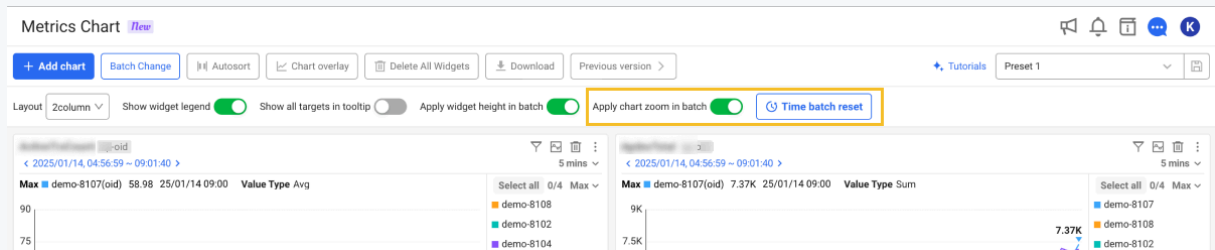
To change the chart query time range for an individual widget, select **<** or **>** at the top of the widget. The time change range is the same as the time reference set in the **Time** option. For example, if the **Time** option is set to **1 day** and you select **<** or **>**, you can change the time to a daily interval.



Alternatively, by dragging a specific area of the chart, you can change the time according to the dragged range for a view.



- ✓ You can change the time for all widgets by changing for each widget with the **Apply chart zoom in batch** option enabled at the top of the screen.



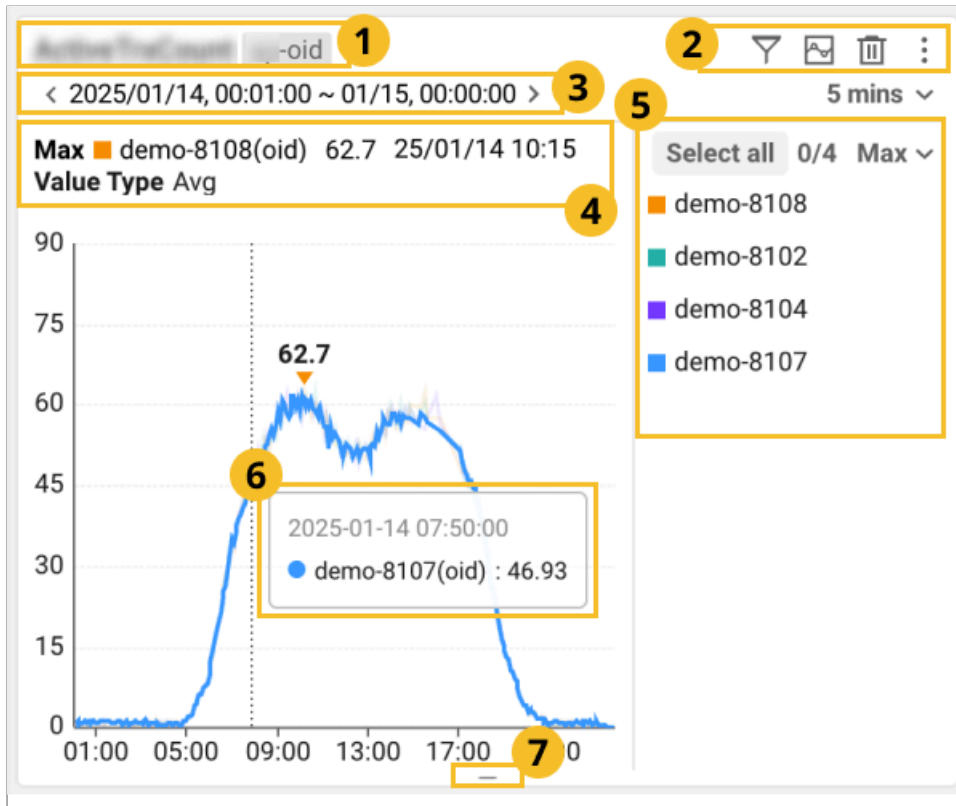
- To change the query time range and then change it back to a recent time, select .

If the **Apply chart zoom in batch** option is enabled at the top of the screen, the  **Time batch reset** button appears. By selecting this button, you can reset the time for all widgets.

- ! By using the **Batch Change** feature, you can change the time for all widgets and change the monitoring target and interval. For more information, see [the following](#).

Chart details

See details for each item displayed in the chart widget.



- **1 Metric name and category:** It displays the metric name of the chart widget. On the right of the metric, the categories of the query targets are displayed in tag format. The category name may differ depending on the product.
 - `*oid` : It is displayed when an individual agent is selected.
 - `project` : It is displayed when the target is selected on a project basis.
 - `okind` : It is displayed when a group classified as `whatap.okind` is selected in the agent configuration.
 - `onode` : It is displayed when a group classified as `whatap.onode` is selected in the agent configuration.
- **2 Widget option:** Option menu of the chart widget. For more information, see [the following](#). For more information about the interval options, see [the following](#).
- **3 Time:** It displays the time range for which data in the chart is viewed. By selecting `<` or `>`, you can adjust

the time range for a query.

- **4 Data:** You can check the maximum value (**Max**) of the data viewed from the monitoring targets and the data merge method (**Value Type**).
- **5 Legend:** It displays the monitoring targets as a legend. For more information about the display of legend, see [the following](#).
- **6 Chart:** It displays the viewed data in a time series graph. If you hover your mouse pointer over the graph, you can see detailed information through a tooltip. By dragging a specific area of the chart, you can change the time according to the dragged range for a view.
- **7 Resize:** You can adjust the size by dragging the center of the bottom of the chart widget.

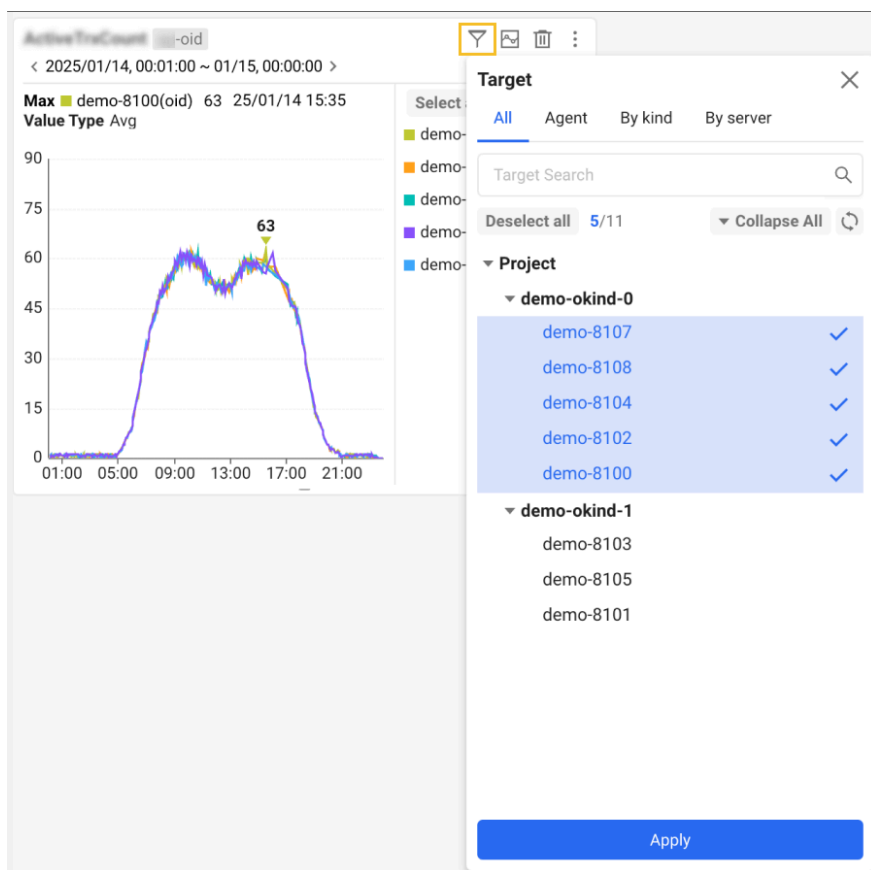
Using the chart widget options

Various features can be used through the option buttons on the upper right of the chart widget.

Filtering targets

You can add or remove monitoring targets to/from the chart widget added to the screen.

1. If you select , a pop-up menu appears for you to select monitoring targets.



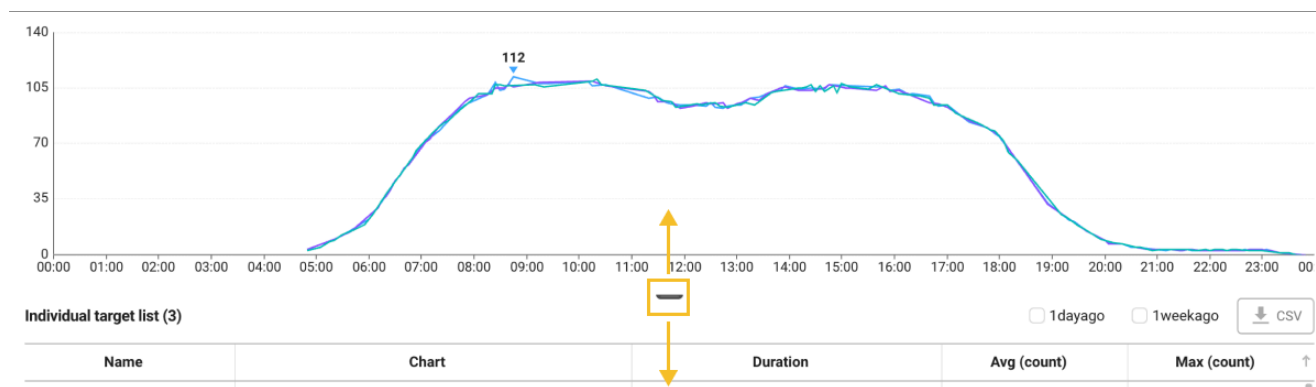
2. Select the desired targets according to the instructions on the screen.
3. Select **Apply**.

Viewing the details

To check the detailed information about the chart data in a wider screen, select . The **View details** window appears. You can view charts, averages, and maximum values for individual targets.



- **1dayago:** You can compare the data with that from the previous day.
- **1weekago:** You can compare the data with that of the previous week.
- **↓ CSV:** The viewed data can be downloaded as a CSV file.
- You can resize the chart by dragging the icon at the bottom of the graph.

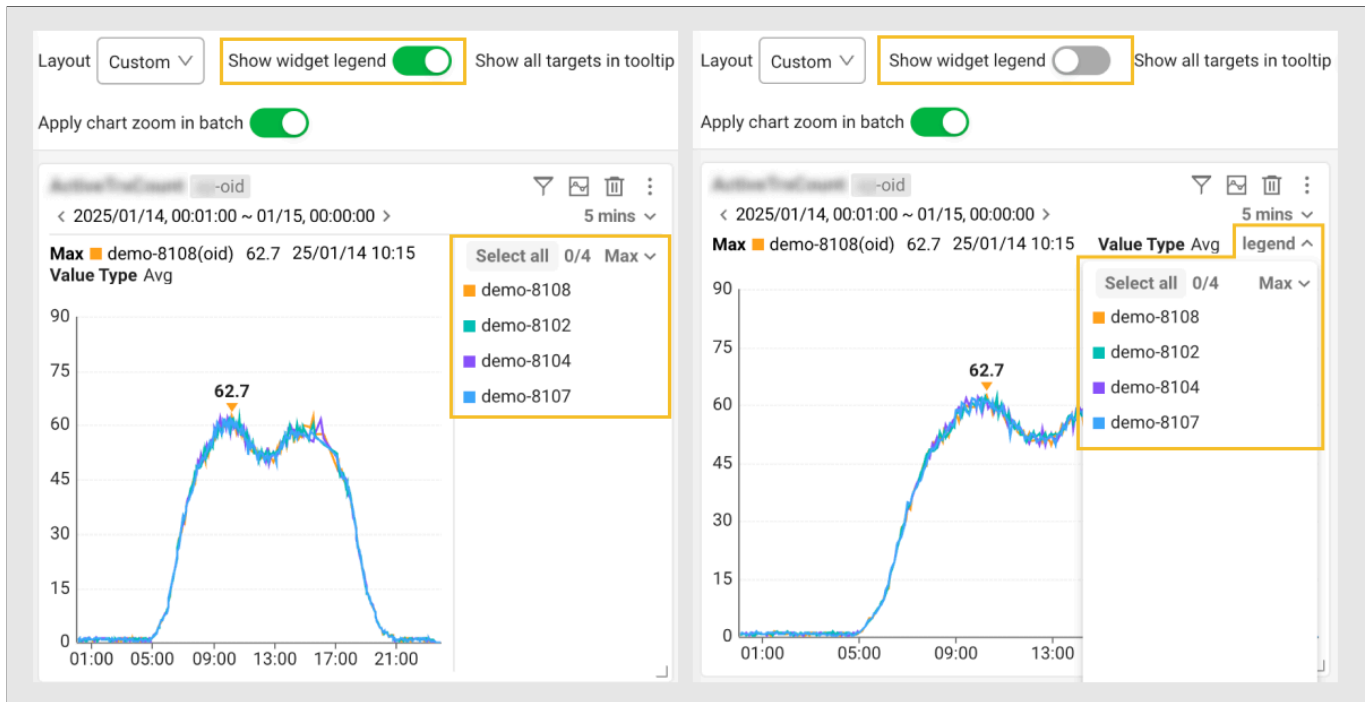


Removing the chart widget

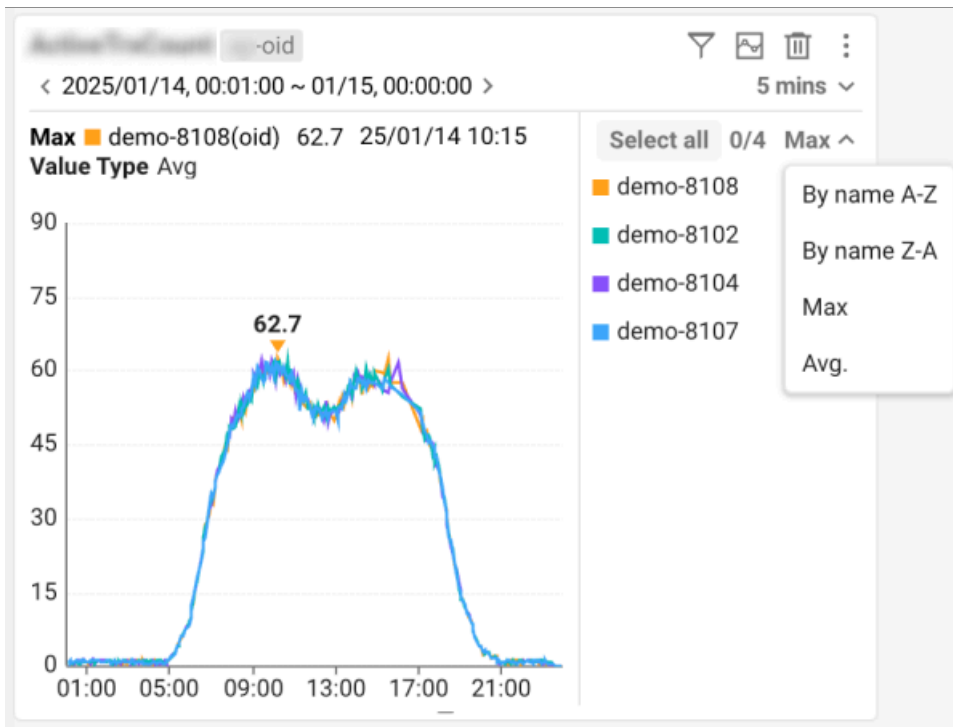
To remove any chart widgets placed on the screen, select .

Using the legend

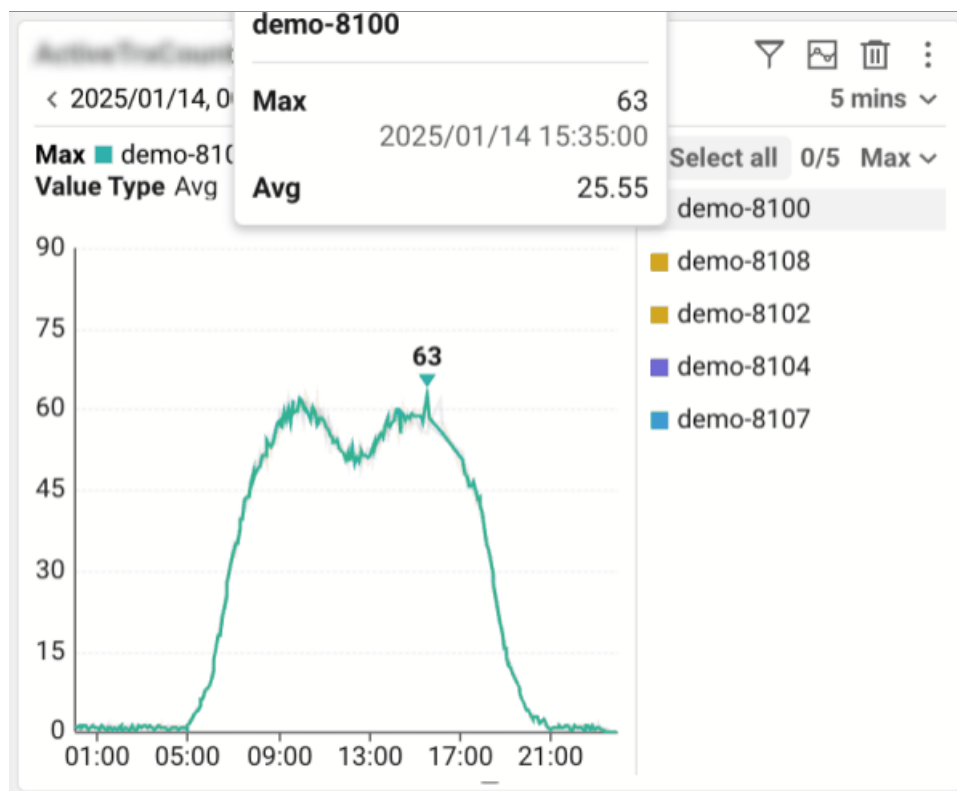
You can check the monitoring targets displayed on the chart through the legend and then sort or filter them by desired criteria. You can show or hide the legend on the right of the widget's chart by turning on or off the **Show widget legend** option at the top of the screen. With the option turned off, you can check and select the monitoring targets by clicking **legend**.



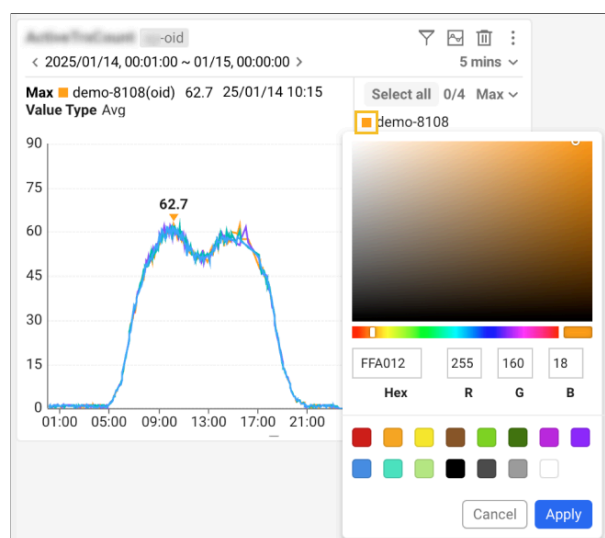
- You can select or unselect monitoring targets by selecting **Select all/Deselect all**.
- In the legend list, you can sort monitored objects by name, maximum value, or average value.



- If you hover the mouse over each item in the legend list, the highlighted graph for the monitoring targets appears and a tooltip indicating the specific time at which the maximum value is recorded. Alternatively, you can filter by selecting a target from the legend list.



Changing the legend color



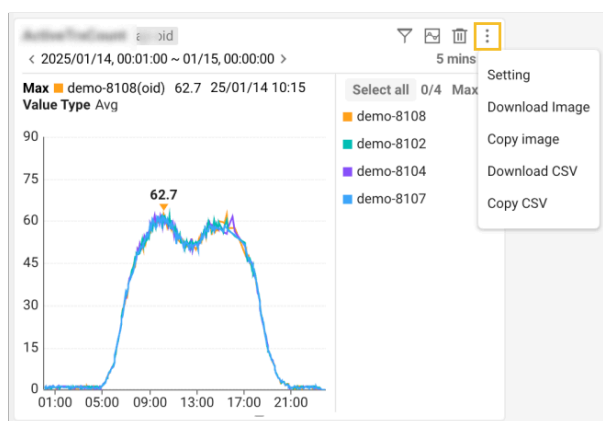
You can change the graph color corresponding to a monitoring target. In the legend list, select the square box

on the left of the target to change its color. Select a desired color and then click **Apply**.

! The set colors are not saved via the preset feature. This feature is to be provided in a future update.

Using additional options

If you select  on the upper right of the chart widget, you can select the following additional options.



- **Setting**
 - **Use Y axis automatically:** The size of the Y-axis can be automatically adjusted within the searched range.
 - **Show maximum value:** You can display the maximum value in the graph area of the chart.
- **Download Image:** You can download the chart as an image file.
- **Copy image:** You can copy the chart to the clipboard as an image format.
- **Download CSV:** You can download the data searched from the chart as a CSV file.
- **Copy CSV:** You can copy the data viewed in the chart to the clipboard in a CSV format.

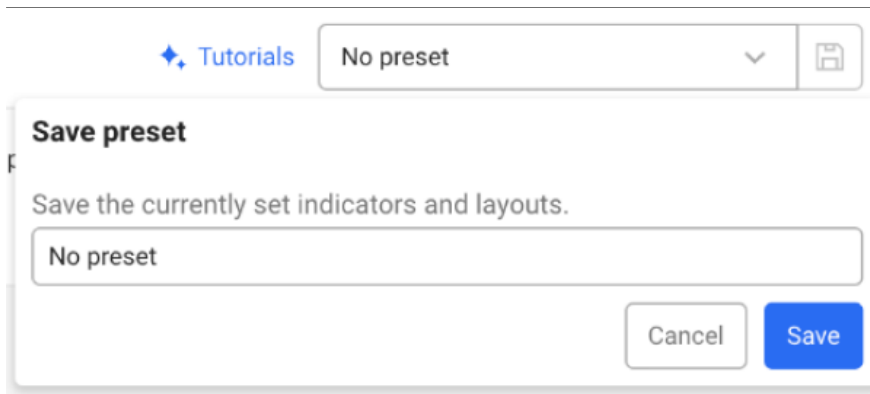
Setting presets

You can save and load the settings and layouts for widgets on the dashboard. You can create a new preset by adjusting its size and placing it on the desired position.

- ! You can save or modify the presets with the **Edit project** role. For more information about the member roles, see [the following](#).

Creating a new preset

1. Place the widgets in the desired format on the dashboard. You can also resize and place only the widgets to be frequently checked.
2. On the upper right of the screen, select .
3. Enter a new preset name.



4. Select **Save**.

You can see the newly saved preset from the preset list.

- ! • If any changes are made to the newly created preset, save the preset again. Select  and then save the preset with the same name. Any changes are overwritten in the existing preset.
- If you go to another menu without saving changes on the dashboard, the changes are not saved.
 - Presets are saved on a project basis and can be shared with other users.
 - The targets saved in the preset are the layout of the dashboard, the widgets placed on the dashboard, and the options (**Target**, **Interval**) modified in the widgets. The **Time** option at the top of the screen is not saved.
 - To change the widget query time after loading a preset, use the **Batch Change** feature. For more information, see [the following](#).

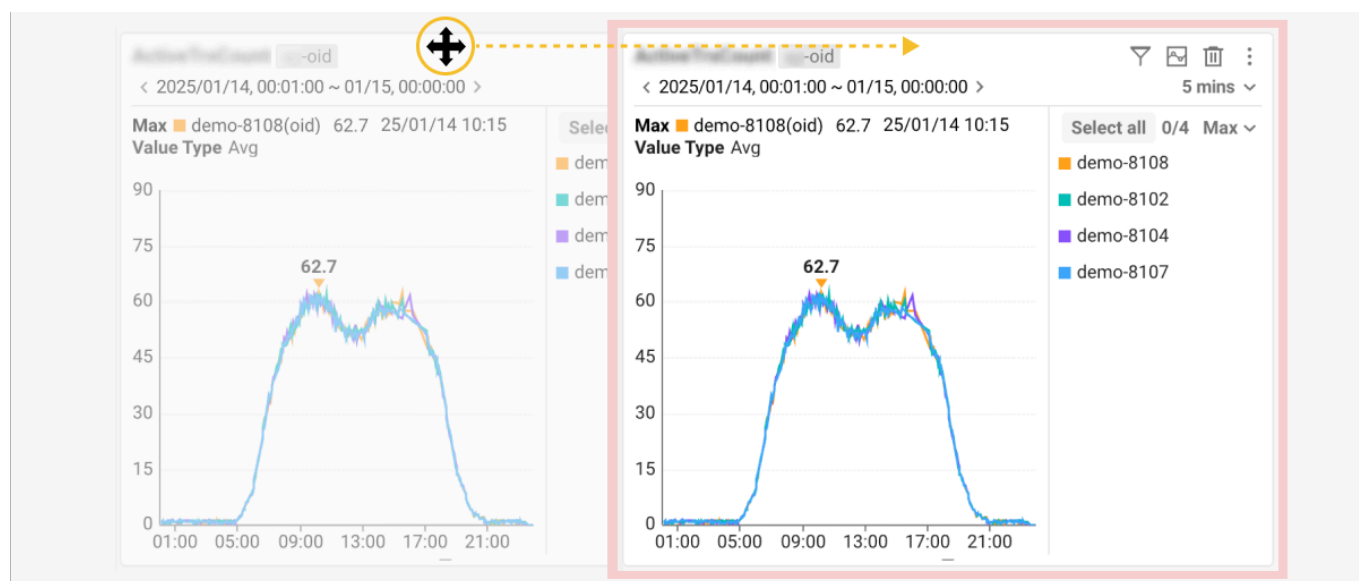
Deleting presets

If you have unused presets, you can delete them from the preset list. Select **×** on the right of the item to delete from the preset list.

Editing a dashboard widget

The widgets placed on the dashboard can be placed and resized to the user's desired location.

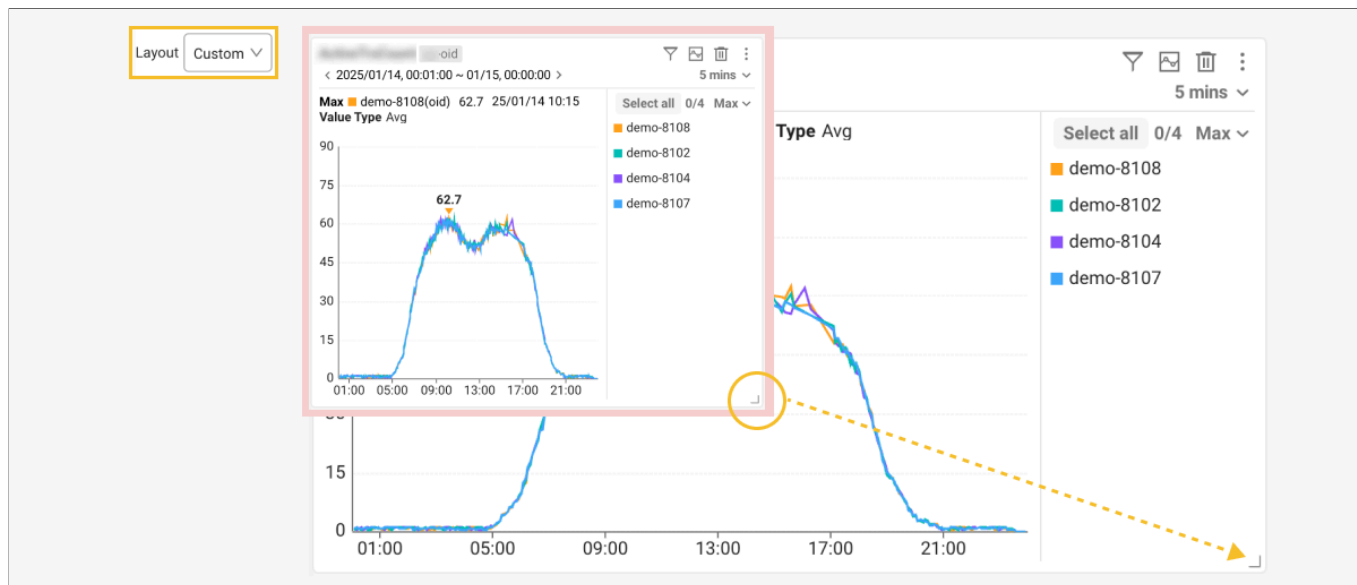
Moving a widget



When you move the top part of the widget by using the mouse pointer, it changes to the **+** shape. At this time, you can move the widget by dragging with your mouse to the desired position.

Resizing a widget

If **Layout** among the dashboard options is set to **Custom**, you can resize the widget.

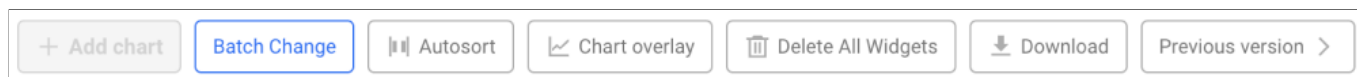


Click and hold the mouse on the  element at the lower right of the widget, and then drag it to the desired size.

ⓘ If **Layout** is not set to **Custom**, you can adjust only the height of the widget. Resize the widget with the bottom center of the widget clicked with your mouse.

Using the dashboard options

You can apply various settings to the chart widget at once using the Option menu located at the top of the screen. You can also customize the dashboard layout to your desired format.



Batch update of the chart widget options

It provides the feature to apply detailed options to the chart widgets on the screen at once. On the upper left of the screen, select **Batch Change**. If the **Batch Change** window appears, set the desired options and then select **Apply**.

Batch Change
×

Time
< 2025/01/15 00:00 ~ 2025/01/15 23:59 23 hr

Target

KUBE_NODES
KUBE_APPS

Target Search

Deselect all 1/6
Collapse All

▼ Project

ap-northeast-2.compute.internal ✓

ap-northeast-2.compute.internal

ap-northeast-2.compute.internal

ap-northeast-2.compute.internal

ap-northeast-2.compute.internal

Apply widget

Select all

☐ CpuTotal
2024/12/08, 00:00:00 ~ 12/09, 00:00:00 project, sm-oid 5 mins

☐ MemTotal
2024/12/08, 00:00:00 ~ 12/09, 00:00:00 project, sm-oid 5 mins

☐ WriteBps
2024/12/08, 00:00:00 ~ 12/09, 00:00:00 project, sm-oid 5 mins

☐ Writelops
2024/12/08, 00:00:00 ~ 12/09, 00:00:00 project, sm-oid 5 mins

☐ MemoryPerLimit
2024/12/08, 00:00:00 ~ 12/09, 00:00:00 project, sm-oid 5 mins

☐ MemoryPerRequest
2024/12/08, 00:00:00 ~ 12/09, 00:00:00 project, sm-oid 5 mins

Interval
no change 5 mins 1 hr

Cancel Apply

- **Time:** You can set the data query range for the chart.
- **Target:** Select a monitoring target. You can select an individual agent or a group classified by the agent configuration.
 - **Agent:** You can select an individual agent or all agents.
 - **By kind:** Agents can be monitored in each group classified by `whatap.okind` in the agent configuration.
 - **By server:** Agents can be monitored in each group classified by `whatap.onode` in the agent configuration.
- **Interval:** You can adjust the data aggregation unit to various time intervals.

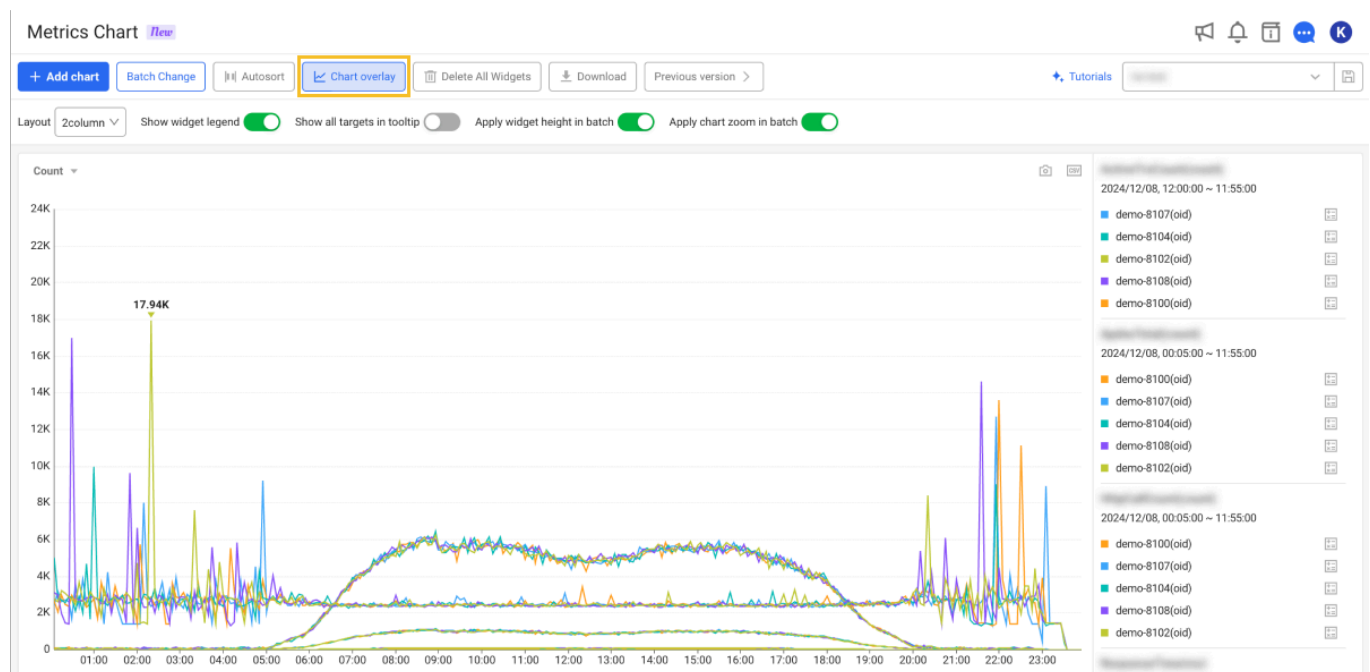
- **Apply widget:** Select a chart widget to apply the options to change. You can apply changed options to the selected widget at once.

Auto-aligning the dashboard

To auto-align the widgets placed on the dashboard, select **Autosort**. It auto-aligns widgets in order of their width and height, from smallest to largest. If the widths are the same, it aligns them in order of height, from smallest to largest.

Comparing multiple charts

This feature allows you to view data from multiple charts placed on the dashboard in a single chart. This feature allows you to compare multiple metrics from the same time period with others. Select **Chart overlay**. You can place overlay charts on the screen.



- If you select **Count** on the upper left of the overlay chart, a list appears where you can change the data criteria for the x-axis chart. You can compare the data of the chart by selecting a desired item.
- : You can download the chart as an image file based on the current view.
- : You can download the chart as a CSV file based on the current view.

- In the right section, you can see a list of agents for each metric. The colors of individual agents match the graph colors on the chart.

Batch deletion of placed widgets

You can delete multiple widgets placed on the dashboard all at once. Select **Delete All Widgets**. When the confirmation message appears, select **Delete**.

Downloading as a CSV file

You can download data from the widgets placed on the dashboard as a CSV file. Select **Download > Download CSV**. You can check the data of metrics for each time zone.

Downloading as an image file

You can download all the widgets placed on the dashboard as a single image file. Select **Download > Download Image**. You can save a captured image of the dashboard screen as is.

Using the previous version

To use the **Metrics Chart** menu of the previous version, select **Previous version**. Alternatively, you can see in the  **Sitemap > Analysis > Metrics Chart *Old*** path.

 For more information about **Metrics Chart *Old***, see [the following](#).

Using other options

Other options below allow you to adjust the layout of the dashboard or easily and conveniently batch set the applied options to the widgets.



- **Layout:** You can set the number of columns in the dashboard (or the number of widgets that can be placed in

a row). You can set up to 4 columns. If **Custom** is selected, widgets can be placed regardless of the number of columns.

❗ In **Custom**, you can freely adjust the width and height of the widget. If you select another option, you can only adjust the height of the widget.

- **Show widget legend:** Turning the option on or off shows or hides the legend on the right of the widget's chart. With the option turned off, you can check and select the monitoring targets by clicking **legend**. For more information about the use of legend, see [the following](#).
- **Show all targets in tooltip:** When you hover your mouse pointer over the chart, you can see the value for the point on the graph through a tooltip. Turn on the option to view data for all targets in the time zone where the mouse pointer is located through the tooltip.
- **Apply widget height in batch:** Enabling this option forces all widgets to have the same height.
- **Apply chart zoom in batch:** When changing the data query time of an individual chart widget, turn on this option to also change the chart times for other widgets.

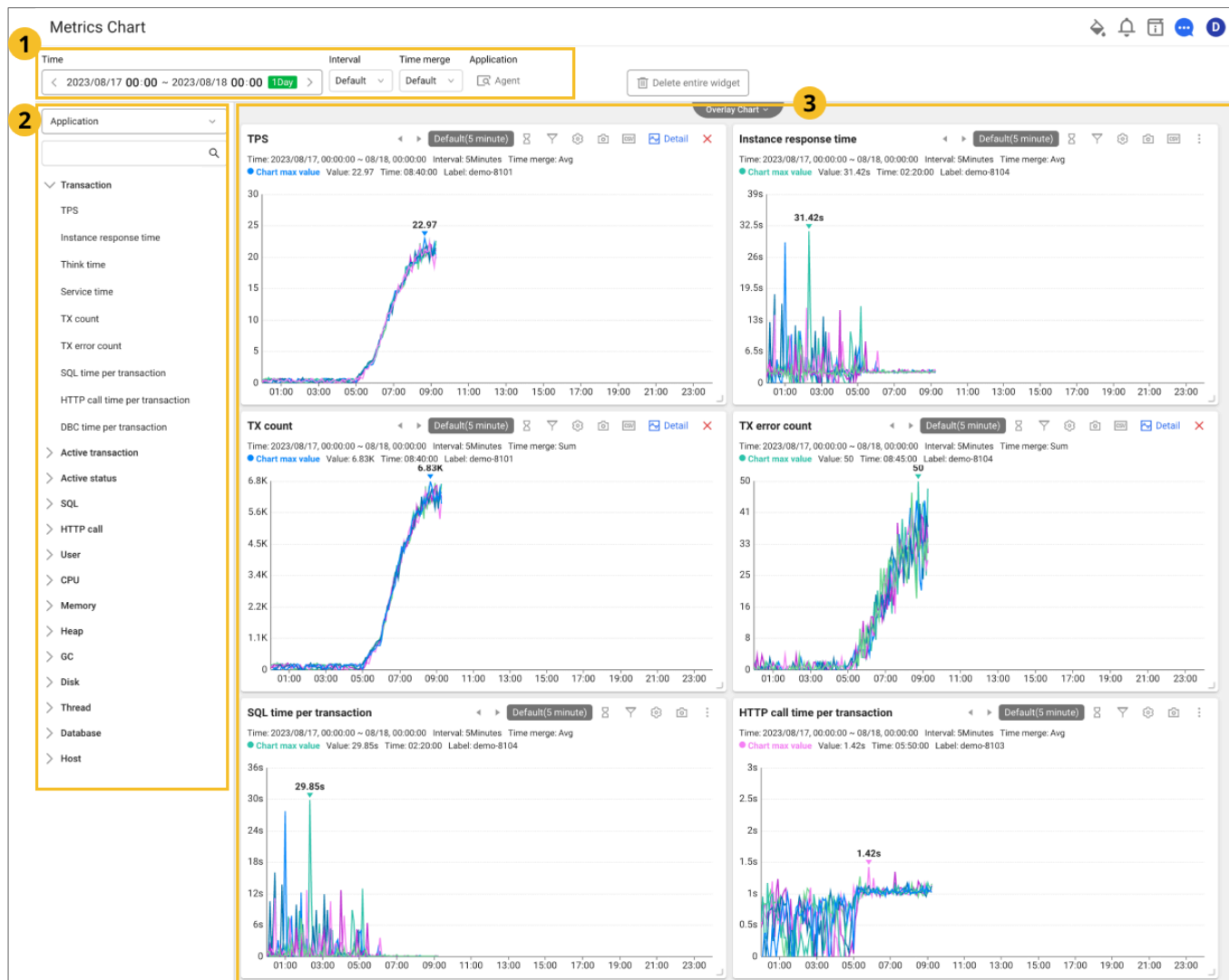
Using the tutorial

When you select **Tutorials** on the upper right of the screen, you can check the changes after the [Service 2.8.0](#) update.

Metrics Chart

[Sitemap](#) > [Analysis](#) > [Metrics Chart](#) Old

In **Metrics Chart**, you can view the metrics data collected from the monitoring targets in a chart as follows. Specifying the time and metrics is required.



Options at the top

In the **1** area, you can specify the time range of the chart and the agent to be monitored through the options at the top of **Metrics Chart**.

- **Time:** You can specify the start and end of the X-axis as the total range in time.

① For more information on how to use the **Time** option.

- **Interval:** You can specify the X-axis data interval as the time interval.
- **Time merge:** As one of the data merging methods, you can merge data within the time specified by intervals.

For example, **Avg** indicates the average value of data within 1 hour.

- **Agent:** You can specify the agents to query. If not specified, all items are queried.

Metrics list

The **2** area is a list of metrics to query for options. First select a **Category**. After querying the metrics under the selected **Category**, select a desired metric. If you select the **Category** and metric, Based on the data within the time range specified in the top menu of the area **1**, you can see the chart widgets in the **3** area.

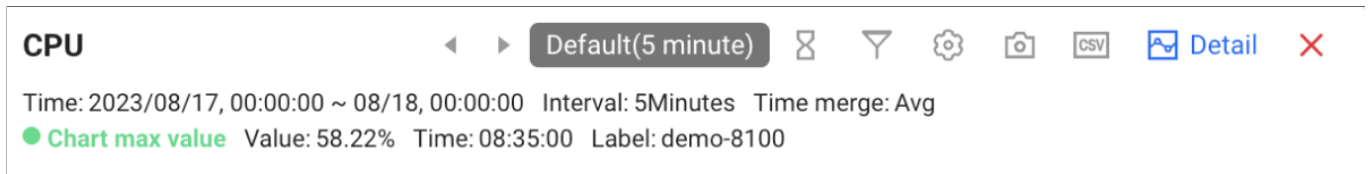
① Data Merge

Data merge provides the following methods: **Object Merge** and **Time merge**.

- **Time merge** merges data with the same field values at regular intervals in the original data.
- **Object Merge** merges data with matching tags among data with different field values.

Chart widget

On the top left of **3**, you can see the metric name. On the top right of the chart widget, you can see the following options:



- Time movement: ◀ ▶ Using the left arrow button or right arrow button, it can be moved by -1 or +1 as much as the selected time range.

For example, if the time range is between 00:00 on February 13 and 00:00 on February 14, when you select the ◀ left arrow button, you can see the data within the time range between 00:00 on February 12 and 00:00 on February 13.

- Interval/Time Merge: In the menu at the top of ❶, you can modify the specified interval and time merge.
- By selecting the monitoring target: ⌵, you can specify the monitoring targets. If not selected, it searches for all.
- Time comparison: If you select ⌛, you can compare the current trend of the same metric with that of the previous time zone.
- By selecting the snapshot: 📷 icon, you can snapshot charts except for widget options.
- By selecting the CSV: CSV icon, you can download the plot data on the chart as a CSV file.
- **Detail:** By selecting the 📊 icon, the detailed view can be seen. When there are many monitoring targets, you can see the metrics trends for each monitoring target individually.

❗ If you do not see any options at the top of the metrics chart widget, select the ⋮ icon.

Deleting a widget

To delete all widgets placed on the screen, select 🗑 **Delete All Widgets** on the screen.

Metric Anomaly Detection

Home > Select Project > **Analysis** > **Metrics Anomaly Detection**

You can compare the patterns of various metrics with the expected patterns learned by the AI. Anomaly detections that deviate from the expected patterns can be viewed through the graph chart. You can check recurring patterns based on the past data and use them to predict future metrics values.

- ❗ The widgets placed on the screen are not saved and reset when you go to another menu.
- It is similar to **Analysis** > **Metrics Chart** except for the pattern and anomaly display.
- The alert, **Anomaly detection** is based on the **Metric anomaly detection**. For more information about the **Anomaly detection** alert setting, see [the following](#).

Checking the widgets



- ❶ The bright color areas of the graph are the estimated patterns analyzed by the AI.
- ❷ The blue graph is the metrics trend of the project.
- If it deviates from the expected pattern analyzed by the AI, it displays dots on the graph in Orange and Red steps. The values that are far from the expected pattern range appear in red.

Placing the widgets



1. In **1 Time**, set a desired time interval. You can set the interval up to 1 month.
2. In **2 Filter**, select a range of the metrics.
3. In the following list, select a metric for monitoring.

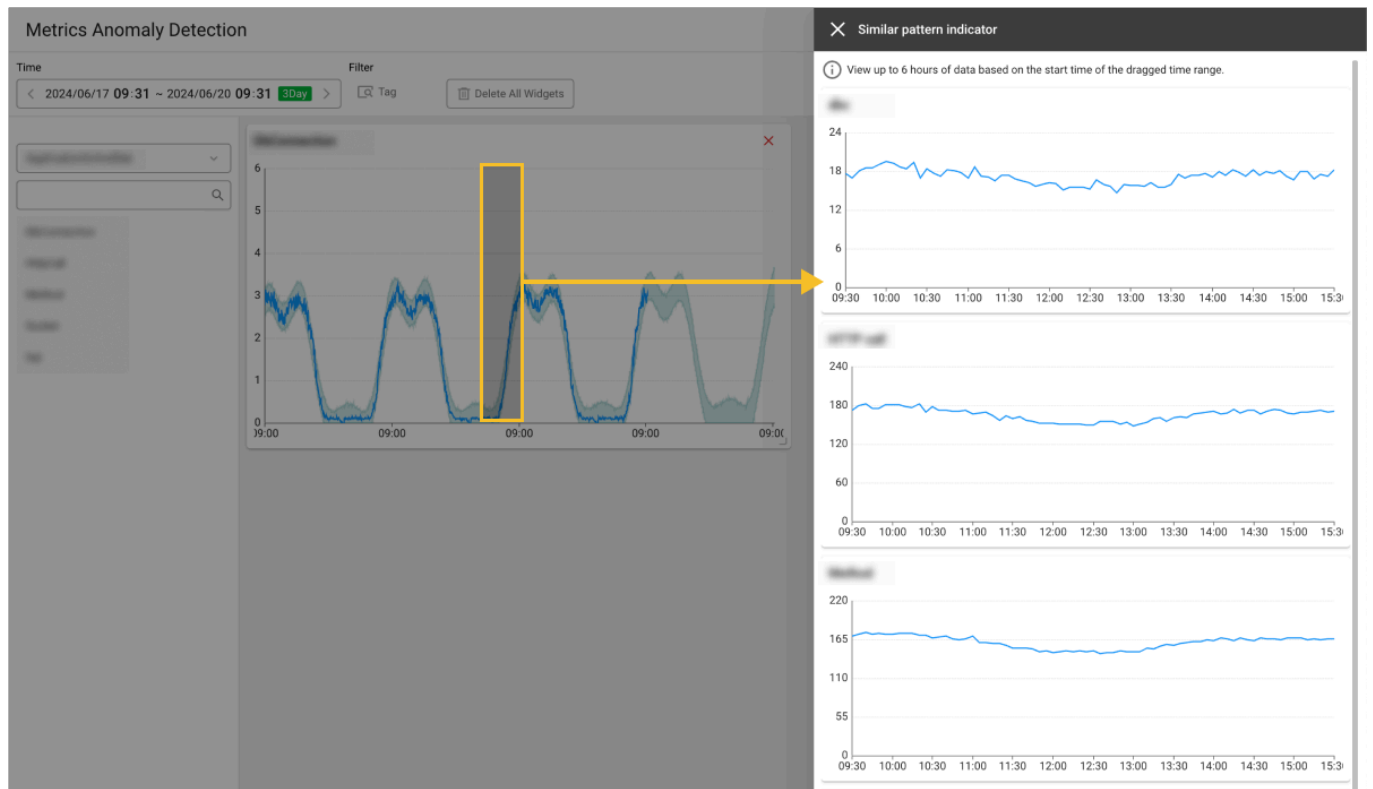
Place the selected metric to the right of the screen.

- Select **2 Tag** to check the metrics separately by selecting a monitoring target. A tag selection list appears on the right of the screen. Select desired items, and then add widgets.
- The widgets placed on the screen are not reflected in the chart even if the **Time** or **Tag** values have been changed.
- Select **X** on the upper right of the widget on the screen to delete.
- You can reposition a widget by dragging it while selecting the upper left part of the widget.
- You can resize a widget by selecting the upper right part of the widget and then dragging it.

Deleting a widget

To delete all widgets placed on the screen, select **Delete All Widgets** on the screen.

Checking the related metrics



Drag part of the chart in the widget. The **Similar pattern indicator** window appears on the right of the screen. You can search the related metrics for up to 6 hours based on the start time of the time range you dragged.

Metrics Search

Home > Select Project > **Analysis** > **Metrics Search**

You can query specific metrics based on tags in **Metrics Search**.

1 Metrics Search

Time selector: < 2023/08/17 07:57 ~ 2023/08/17 08:57 60Min > Max count: 100 Category: app_counter

2 Tags: All ☒ Filter: Filter by tag value

Field: All ☒ active_tx_3 active_tx_8 active_tx_count apex_satisfied apex_tolerated apex_total arrival_rate httpc_count httpc_error httpc_time meterlog resp90 resp95 resp_time sql_count sql_error sql_fetch_count sql_fetch_time sql_time tps tx_count tx_doc_time tx_error tx_http_time tx_sql_time tx_time tx_time_agg_sum 5/29

3

| | Time | Oid | Tags | | | | | | | | Fields | | | | |
|---|---------------------|------------|---------------|--------------|------------|--------------|-----------|-----------|---------|------|-------------|--------------|-----------|-----------|----------|
| | | | container | containerKey | hostIp | okindName | oname | onodeName | pid | type | active_tx_3 | arrival_rate | sql_time | tps | tx_error |
| 1 | 2023-08-17 07:57:40 | 633280970 | agent.service | 1395943798 | 10.21.1.26 | demo-okind-1 | demo-8103 | node-1 | 3285013 | java | 10 | 20.57942 | 7.822286 | 20.94554 | 0 |
| 2 | 2023-08-17 07:57:40 | -857948929 | agent.service | 1395943798 | 10.21.1.26 | demo-okind-1 | demo-8105 | node-1 | 3284721 | java | 10 | 17.974834 | 10.626543 | 20.609276 | 0 |

1 Time and category selection

In the **1** area, you can specify the **Time selector**, **Max count**, and **Category** for the collected metrics. **Time selector** and **Category** must have been specified.

- **Time selector:** You can query by specifying the time when metrics were collected. The default value is 1 hour. In addition to the lookup time provided as the default option, you can specify the date and time after selecting the **Time selector** tab.

! For more information on how to use the **Time selector** option.

- **Max count:** You can specify the maximum number of metrics to search in the **3** list. You can set the count up to 10, 50, 100, 200, 300, 1000, 2000, and 3000.
- **Category:** It is a classification unit for the related metrics. You can set the desired **Category** after selecting the **Category** tab.
- **Refresh categories, tags, and field options:** **Category**, **Tags**, and **Field** options can be fetched.

2 Tag and field selection

In the 2 area, select **Tags** and **Field**. The default setting is "Select All," unless specified separately by user.

- **Tags:** Unique data that can identify the collected targets.
- **Field:** Metrics collected from the monitored targets.
- **Filter:** Select  **Filter by tag value** and then set the tag value for filtering.

For example, you can search filtered data by setting the value of `oname` to `demo-8101` .

-  **Search:** If you set the conditions and then select **Search**, you can search the source data of the metric in the 3 area.
-  : You can download the source data of the metric as a CSV file.

3 Metrics table

Because the metrics to be collected cannot be specified in advance, it is important to check the source data for all metrics being collected. After setting the desired conditions in the condition area above, you can search the original data of the metric in a table format in the 3 area. Columns in the table are changed when you specify the conditions for **Tags** and **Field**.

- To search metrics, **Time selector** and **Category** must have been specified.
- To search metrics, **Tags** and **Field** are optional.

Log

Home > Select Project > [Log](#)

You can check various logs related to Kubernetes. The monitoring categories provided by WhaTap Kubernetes are as follows:

| Category | Description |
|-----------------|--|
| #K8sEvent | <ul style="list-style-type: none">• Logs that store the events that occur in Kubernetes• Generation by default regardless of user settings |
| #WhatapEvent | <ul style="list-style-type: none">• Logs that store the events that occur by WhaTap event settings• Generation by default regardless of user settings |
| containerStdout | <ul style="list-style-type: none">• Container Standard Out logs• Generation by user settings• When <code>logsink_enabled=true</code> setting is added to the node agent |
| AppLog | <ul style="list-style-type: none">• Application logs in the container• Generation by user settings• When <code>logsink_enabled=true</code> setting is added to the application agent |

ⓘ Depending on the settings, duplicate log contents may be stored. Be sure to check for duplicates.

Setting the containerStdOut log filter

Home > Select Project > **Management** > **Agent CONF.**

The following guides you to filtering options to select information needed in the Kubernetes logs. You can set

which logs to allow or exclude through filtering. In **Agent CONF.**, select **Direct Input** in the **Write options** manner.

The screenshot shows the 'Agent CONF.' configuration page for the 'Node' agent type. The 'Write options' section is highlighted with a yellow dashed border. It contains three input fields: 'log_filter_enabled' set to 'true', 'log_filter_allow_list' set to '*_*_*', and 'log_filter_deny_list' set to '*_infra_*_*_*_monitoring_*'. Below these is a 'Select option' dropdown menu with 'Direct Input' selected. The 'Node' agent type is also visible in the left sidebar under 'Agent type list'.

Log naming conventions

The Kubernetes containerStdOut logs are generated under the `/var/log/containers/` path on the node. They keep the following naming conventions.

```
<podName>_<namespace>_<containerName>-<containerId>.log
```

Log filtering options

- **log_filter_enabled** `bool`

Default `false`

It sets whether to use log filters.

- **log_filter_allow_list** `list`

It sets the list of logs to allow. It indicates the items to be included for filtering.

- **log_filter_deny_list** `list`

It sets the list of logs to exclude. It indicates the items to be excluded for filtering.

Log filtering operation

Log filters work only when the `log_filter_enabled` option is active. You can selectively collect logs via `log_filter_allow_list` and `log_filter_deny_list` when the value of the option is set to `true`.

Log filters work on a blacklist basis. For example, if you enable `log_filter_enabled` and do not set the items to be added to the `log_filter_allow_list` option, the log collection for all containers is blocked by default.

Additionally, in the log filters, deny rules take precedence over allow rules. If the same items are set in both allow and deny rules, the deny rule takes precedence and logs are not collected.

Log filter example

Collecting logs for a single namespace

- Allow `log_filter_allow_list` `*_infra_*`

All containers that exist in the `infra` namespace are targeted for collection.

Collecting multiple namespace logs

- Allow `log_filter_allow_list` `*_infra_*, *_monitoring_*`

All containers that exist in the `infra` and `monitoring` namespaces are specified as collection targets.

Pods containing specific terms are excluded.

- Allow `log_filter_allow_list` `*_*_*`
- Deny `log_filter_deny_list` `*prod*_**`

All containers are targeted for collection, but the logs that contain `prod` in `podName` are excluded.

Pods containing specific terms are collected.

- Allow `log_filter_allow_list` `*prod*_**`

Only the logs that contain `prod` in `podName` are collected.

Multi-namespace logs are collected and Pods containing specific terms are excluded.

- Allow `log_filter_allow_list` `*_infra_*, *_monitoring_*`
- Deny `log_filter_deny_list` `*prod*_**_*`

The logs for the `infra` and `monitoring` namespaces are collected, and the logs that contain `prod` in `podName` are excluded.

The container logs that start with a specific term in the container name are excluded.

- Allow `log_filter_allow_list` `*_*_db-*`

The container logs whose `containerName` starts with `db` in all namespaces are excluded.

Live Tail

Home > Select Project > [Log](#) > [Live Tail](#)

You can check the content in the WhaTap Kubernetes monitoring category in real time. For more information about [Live Tail](#), see [the following](#).

Log Trend

Home > Select Project > [Log](#) > [Log Trend](#)

You can see the stored log trend by setting a time range. For more information about [Log Trend](#), see [the following](#).

Log Search

Home > Select Project > [Log](#) > [Log Search](#)

You can search the stored logs by setting a time range. For example, you can see the standard out logs of a specific container or all containers in the deployment at once. You can also highlight a specific keyword for

display.

For more information about [Log Trend](#), see [the following](#).

Setting the log

Home > Select Project > [Log](#) > [Setting](#)

You can configure various log-related settings including log activation. For more information about the log settings, see [the following](#).

Report is a document that reports the service usage status and failure records for individual projects. The persons in charge of service monitoring share the status report with the people of related departments. The monitoring data analysis is important because it serves as a metric for determining the direction of improvement. However, it is cumbersome to aggregate and document data from multiple dashboards.

WhaTap's **Report** menu supports **automated report creation**. Through the **Report** menu, you can create weekly reports, share reports at the set times, and manage various templates.

- ✓ • Select [Download](#) to download the report. You can download it in HTML format.
- Select [Print](#) to print the report.
- To schedule a report transmission, select [Reservation of Report Email](#).

! Difference between the report and integrated report

Report menu generates a report for a single project. Accordingly, you can select a project and then access it through the report menu on the left. Integrated Report menu generates a report for multiple projects. It can be accessed through the main menu on the Home screen without selecting a project. For more information, see [the following](#).

Select report type to create

You can check various reports related to Kubernetes. The report types provided by WhaTap Kubernetes are as follows:

- [Daily Application Report](#)
- [List of Transactions exceeding the Threshold by Application Type](#)
- [Daily Metric Trend Status](#)
- [Daily Application Error Statistics Report](#)
- [Weekly Application Report](#)
- [Monthly Application Report](#)
- [Daily CPU/Memory Report by POD](#)
- [Weekly CPU/Memory Report by POD](#)

- [Daily Server Report](#)
- [Weekly Server Report](#)
- [Monthly Server Report](#)

Email delivery schedule

To receive regular reports by email, select [Reservation of Report Email](#). You can replace various service checking tasks performed right after work with email confirmation.

Reservation of Report Email

Report will be sent on 04~06 o'clock from tomorrow.

Maximum 5 report mail settings can be registered.

Report type

Reservation Daily Report

Report time

00:00 ~ 23:59

Weekdays

☒ Mon

☒ Tue

☒ Wed

☒ Thu

☒ Fri

☐ Sat

☐ Sun

Emails

Input email and enter

+ Save

Integrated Report

Home > Integrated Report

You can create and share reports that have integrated multiple projects. You can also create a report by selecting a report type and the projects to include, and check the task progress in the creation list. If an error occurs in a specific project, you can see the error details. The reports that have been created can be viewed immediately or shared with the download and print features.

Integrated Report

Report type

Create Report

Application Daily Error Statistics Report

Application Daily Http Call Report

Application Weekly Okind Error Statistics Report

Application Kinds Weekly Avg Resp Time Statistics Report

Daily DB Report

Weekly DB Report

Application Daily Error Statistics Report

Application Weekly Okind Error Statistics Report

Kube Daily Cpu/Memory Report

Kube Weekly Cpu/Memory Report

Daily Server Report

Monthly Server Report

Report generation list ⓘ

All My report

Report title, Project

No reports were generated.

Download

Print



Integrated report

Integrated reports make it easy to create reports for multiple projects.

GuideShortcut

ⓘ The **Integrated Report** menu can generate only the reports for projects of **Application**, **Server**, **Kubernetes**, and **Database** products.

Guide to major features

The **Integrated Report** menu provide the following features:

- **Supporting multiple projects**

It is convenient to be able to generate reports at once by selecting multiple projects. The **Report** menu available in any project can generate reports only for a desired project.

- **Create Report by Kind**

In the **Create Report** feature, you can select projects by the work unit to save as a template. This feature is useful in the Micro Service Architecture (MSA) environment where projects are usually divided by the work unit.

- **Big Data Optimization**

With an aid of this, the report creation time has been drastically reduced. Even if an error occurs while creating a project report, you can view the result of the report.

- **Simultaneous operation during report generation**

You can do other tasks while generating a report. In the **Report** menu available in any project, you must wait until the report generation is completed.

In **Integrated Report**, you can go to another page while creating a report. You can also see its progress and view the created report from the report result list.

- **Sharing the report work**

Any user with project permission can view the created reports. You can check whether the related report has already been created, so you can avoid creating duplicates.

Difference between the report and integrated report

The **Report** menu available in any project generates reports for the project. You can access it through **Report** or **Stat/Report** in the side menu after selecting a project on the initial screen.

The **Integrated Report** menu can generate reports for multiple projects. It can be accessed from Home screen without selecting a project.

Create Report

Selecting the report type

If you select a desired report type in the **Report type** list, the **Create Report** button is activated. The [icon](#) before the report name indicates the project platform that supports report creation.

-  indicates the **Application** project platform.
-  indicates the **Database** project platform.
-  indicates the **Kubernetes** project platform.
-  indicates the **Server** project platform.

Report type

Create Report

</> Application Daily Error Statistics Report

</> Application Daily Http Call Report

</> Application Weekly Okind Error Statistics Report

</> Application Kinds Weekly Avg Resp Time Statistics Report

⌵ Daily DB Report

⌵ Weekly DB Report

⊗ Application Daily Error Statistics Report

⊗ Application Weekly Okind Error Statistics Report

⊗ Kube Daily Cpu/Memory Report

⊗ Kube Weekly Cpu/Memory Report

☰ Daily Server Report

☰ Monthly Server Report

Configuring in detail

When the **Create Report** window appears, set each item.

- **Title:** If there are multiple reports of the same type, the report names make it easier to distinguish the report results.
- **Time:** Select a period for the data to be used in the report. The

period differs depending on the report type (daily, weekly, monthly).

- **Project:** Select a project to create a report. You can select a group or project. The desired project can be searched via the **Search** field.
- **Save report template:** You can save the settings of the current report as a template. You can load it from the **Type** list when creating a new report.

After configuration, select **Create Report**.

Create Report

Type

Application Daily Error Statistics Report

Title

Application Daily Error Statistics Report

Time

< 2025/01/15 00:00 ~ 2025/01/16 00:00 1 day

Project

Search

▼ Project

[34786] .NET APM Demo

[31130] Go APM Demo

[5490] Java APM Demo

[6969] Node.js APM Demo

[35917] OpenTelemetry Monitoring Demo

[39251] petclinic-Mysql-apm

[40824] petclinic-Postgresql-apm

[31324] PHP APM Demo

[29744] Python APM Demo

Select all

Save report template

Create Report

✓ The methods how to set the query period depending on the format of the selected report are as follows:

- **Daily report:** You can select the start/end date and time.

573



- **Weekly report:** Simply select a start date. Then the end date is selected automatically after calculation.
- **Monthly report:** Simply select a start date. Then the end date is selected automatically after calculation.
 - If the start date is the 1st date, up to the end of the month is automatically selected.
 - If the start date is not the 1st date, up to the date before the next month is automatically selected.

Example 1: If the start date is **January 1**, from **January 1 00:00:00** to **January 31 23:59:59** is automatically selected.

Example 2: If the start date is **January 10**, from **January 10 00:00:00** to **February 9 23:59:59** is automatically selected.



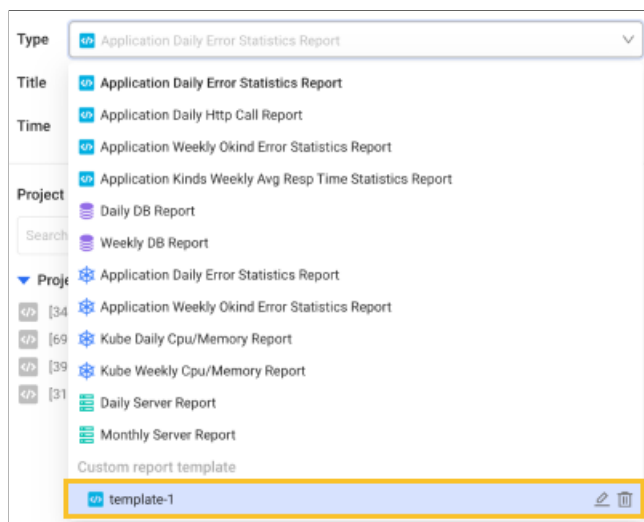
- For more information on how to use the **Time** option, see [the following](#).
- For more information on how to load the saved report forms, see [the following](#).

Loading the report forms

When creating a new report, the form saved with the **Save report template** feature can be checked in **Type** when creating a subsequent report.

Users who frequently use projects as work units can find this highly useful.

- : You can modify any saved report form.
- : You can delete any saved report form.



Checking the report list

You can see the reports that were created and are being created in **Report generation list**.

- In the **All** tab, you can see the reports created by you and the reports created by users with the same roles.
- In the **My report** tab, you can see only the reports created by you.
- If you select  on the upper right of a report item, you can use a feature to copy or delete the report.

Report generation list

All
My report




Application Daily Http Call Report 


2023-08-22 00:00~2023-08-22 23:59
Java APM Demo, .NET APM Demo, Go APM Demo

- 
 • A report created with no error has the  icon. In case of a report with any error, the  warning icon is displayed together. If you select the  icon, you can see in which project the error occurred.
- The generated reports are stored for 7 days, up to a maximum of 100 records.

Viewing and sharing

- **Viewing on the screen**

In the **Report generation list** list, if you select a created report, you can see its details.

- **Download**

Select **Download** to download the viewed report as an HTML file. You can share the report with users who do not have the monitoring management role.

• **Printing and saving PDF**

Select **Print** to print the report via a printer or save it as a PDF file.

DownloadPrint

Application Daily Http Call Report

Search Condition :2023-08-22(Tue) 00:00:00 ~ 2023-08-22(Tue) 23:59:59 (GMT+9)

[5490] Java APM Demo

Application Daily Http Call Report

Search Condition 2023-08-22(Tue) 00:00:00 ~ 2023-08-22(Tue) 23:59:59 (GMT+9)

Http Call Delay Top 5

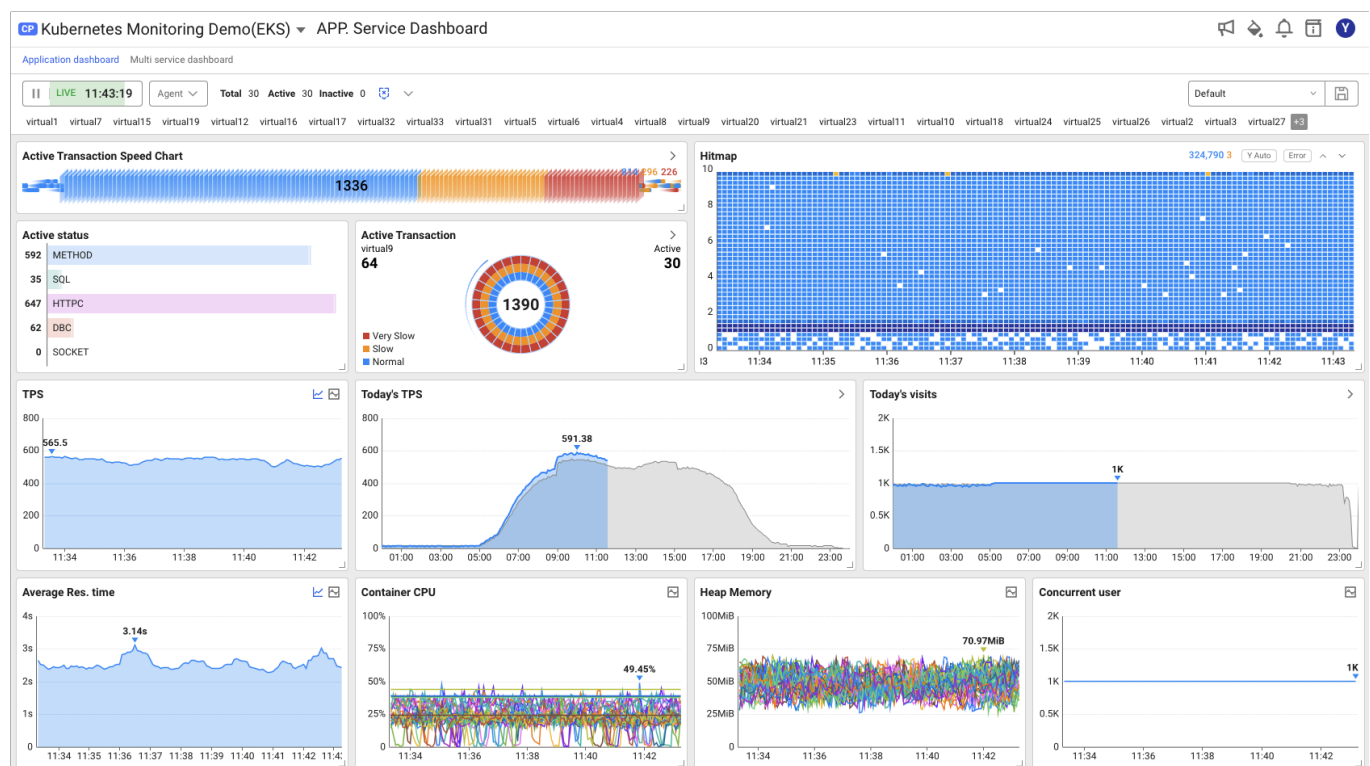
| No. | URL | Call URL | Host | Port | Total | Error | Average Time(m s) | Total Time(m s) | Maximum s) |
|-----|-----------------------------------|---------------------------|-----------|------|-------|-------|-------------------|-----------------|------------|
| 1 | /remote/account/save/dept/kwangju | /account/load/dept/daejun | 127.0.0.1 | 8105 | 2264 | 0 | 1247 | 2823259 | |

Application Service Dashboard

Home > Select Project > [Application](#) > **Dashboard** > **Application service dashboard**

It is a dashboard that allows you to see the status of applications running inside the container in real time. For normal data output of each widget, the node agent must have been installed in the Kubernetes environment, and the WhaTap agent must also have been installed in the application inside the container.

- WhaTap agent installation is supported for the applications developed in [Java](#), [Node.js](#), [Python](#), and [PHP](#).
- You can access it through the [Dashboard](#) menu under [Dashboard](#) or [Application](#).



Application dashboard allows you to collect key metrics of applications through agents, configure charts to monitor the metrics in real time, and view the past data. You can quickly identify and analyze application problems by using the widgets on the dashboard.

In case of general charts excluding CPU and memory, the stable data appears in blue and the elements identified as problems appear in red, making it easy to recognize the current status.

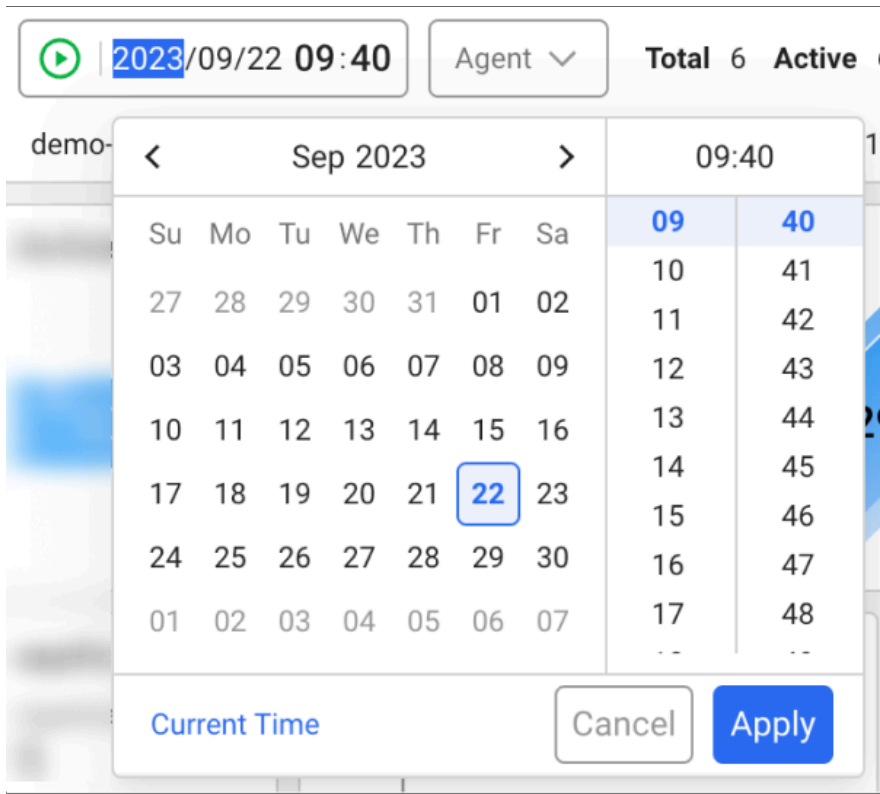
Application Dashboard

Viewing the past data

Application dashboard basically provides real-time monitoring features, in which you can view the past data.

| Real-time mode | Non-real-time mode |
|--|--|
| <div><div> </div><div>LIVE</div><div>15:50:29</div></div> | <div><div>▶</div><div>2023/09/20 15:51</div></div> |

- 1. Select || in the time selector at the upper left of the screen.
- 2. Click the date and time to set the desired time zone.

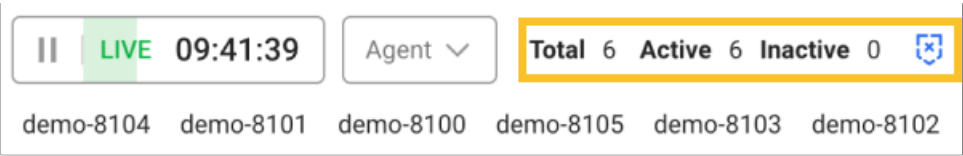


3. Select **Apply**.

It updates the data in widgets placed on the dashboard based on the set time. To switch to the Realtime mode, select .

Checking the agent

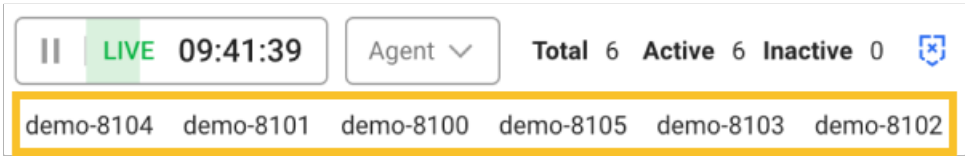
Checking the agent connection status



On the upper left of the screen, the right area of the time selector provides information that allows you to check the status of agents connected to the project. This allows you to immediately check whether the application server is running normally.

- **Total**: Number of all agents connected to the project
- **Active**: Number of active agents
- **Inactive**: Number of inactive agents
- : Can display or hide inactive agents.

Agent-based monitoring

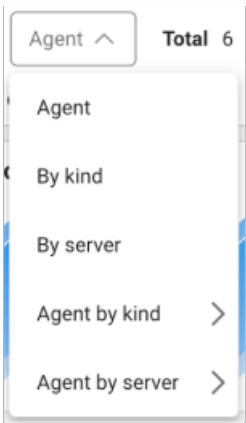


By default, the dashboard displays the metrics collected from all agents in charts and you can also view the data by agent. Select one or more agents under the time selector. The data of the widgets on the dashboard is updated with the metrics of the selected agent(s).

 To select all agents again while one or more agents is selected, deselect them or select [Total](#).

 If there are numerous agents connected to the project, it is efficient to set the agent names to be short. For more information about setting the agent name, see [the following](#).

Monitoring each classified agents



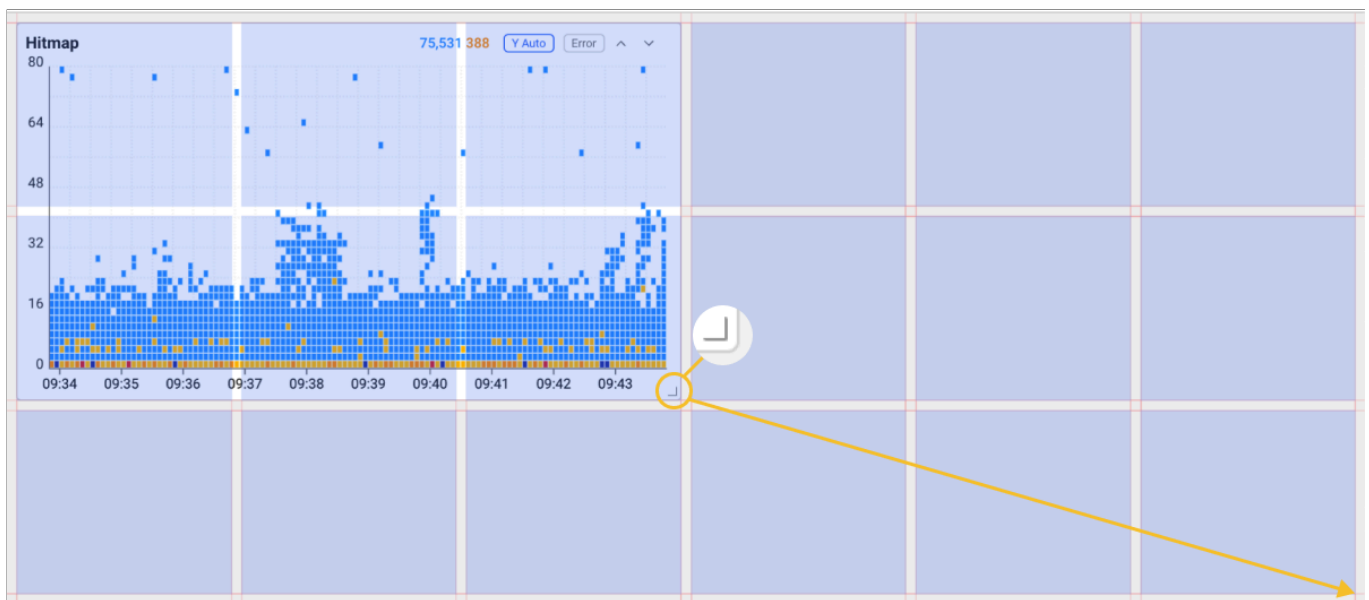
You can select and monitor agents by each group classified through agent settings.

- **Agent:** You can select an individual agent or all agents.
- **By kind:** Agents can be monitored in each group classified by `whatap.okind` in the agent configuration.
- **By server:** Agents can be monitored in each group classified by `whatap.onode` in the agent configuration.
- **Agent by kind:** Only the agents classified as `whatap.okind` in the agent configuration can be monitored.
- **Agent by server:** Only the agents classified as `whatap.onode` in the agent configuration can be monitored.

Editing a dashboard widget

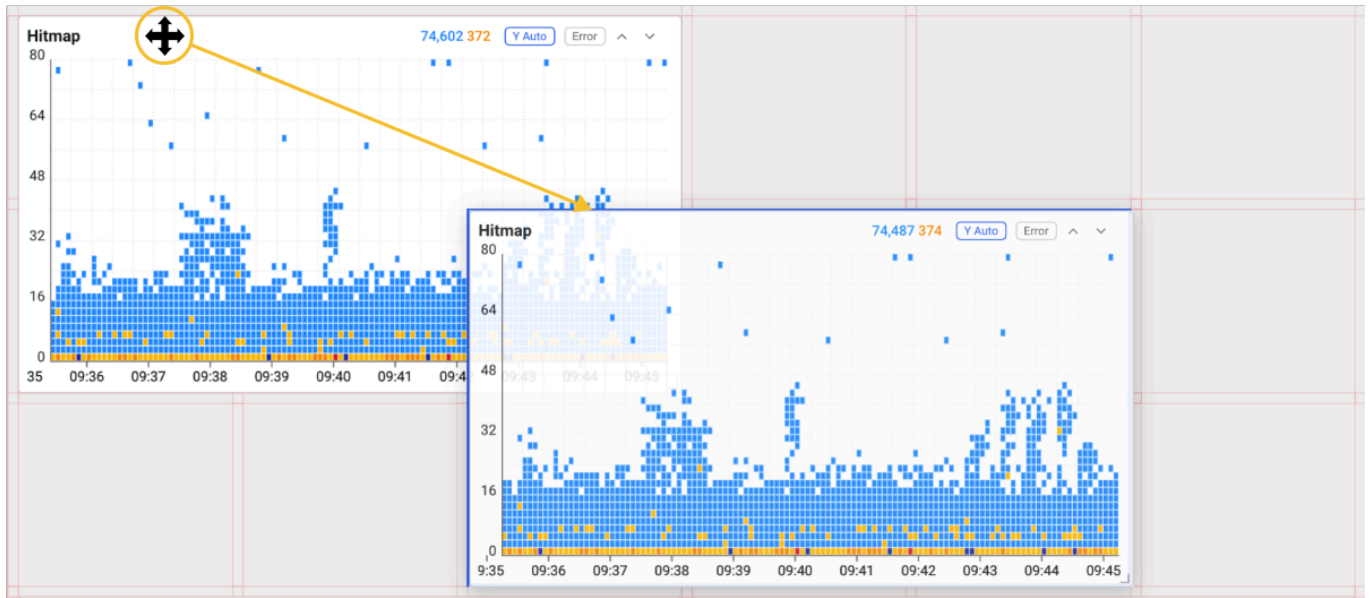
The widgets on the dashboard can be adjusted to the desired size and placed in the desired positions. You can also delete unnecessary widgets or add them again.

Resizing a widget



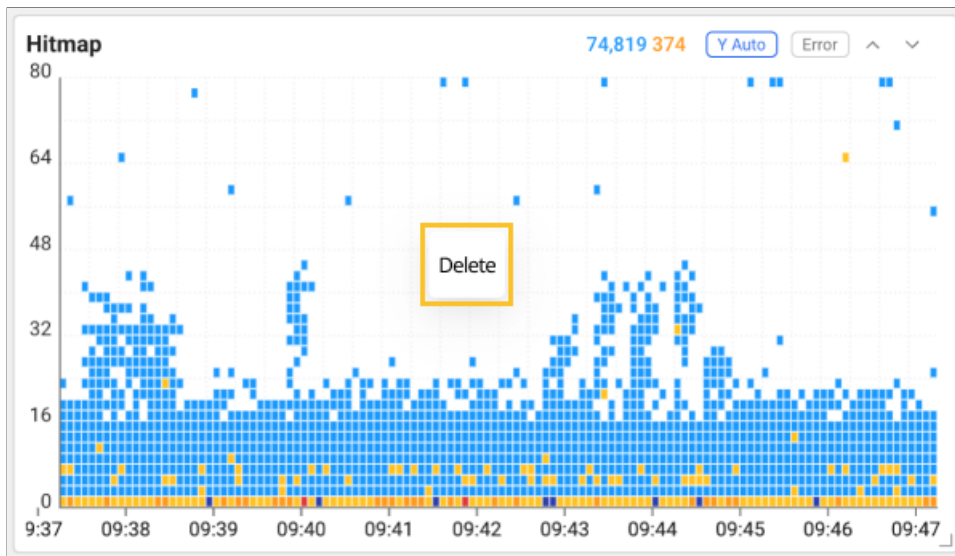
Click and hold the mouse on the  element at the lower right of the widget, and then drag it to the desired size. Grids with a uniform horizontal-to-vertical ratio appear and the size of the widget can be adjusted in each grid.

Moving a widget



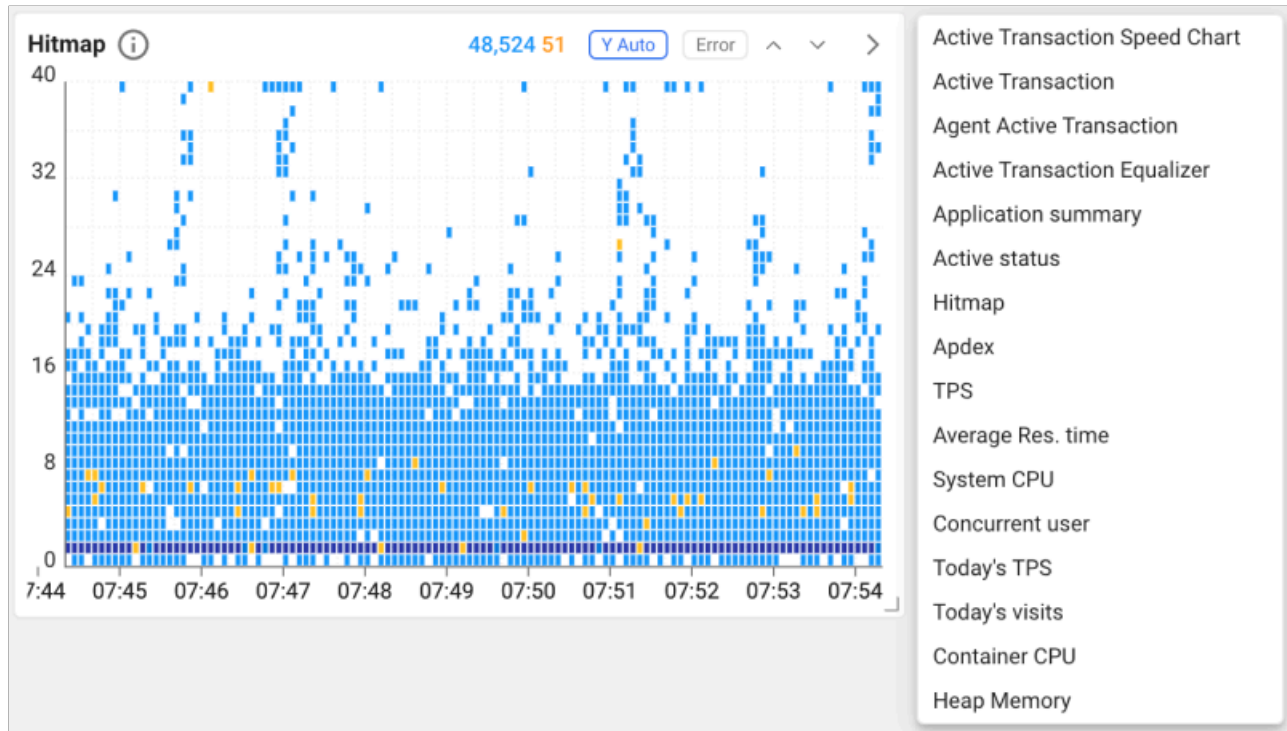
When you move the mouse cursor to the top of the widget, the cursor shape changes to $\oplus$. At this time, you can move the widget by dragging with your left mouse button and move to the desired position.

Deleting a widget



Right-click on a widget to delete. If you select **Delete**, the widget is deleted from the dashboard.

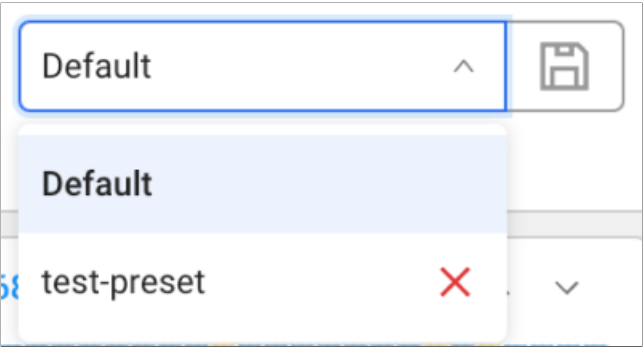
Adding a widget



Move the mouse cursor to an empty space on the dashboard and then right-click on it. Select a widget to add from the pop-up menu. Place the widget to the desired position and then resize it.

! The widgets that can be added are fixed, but we plan to increase supports through future updates.

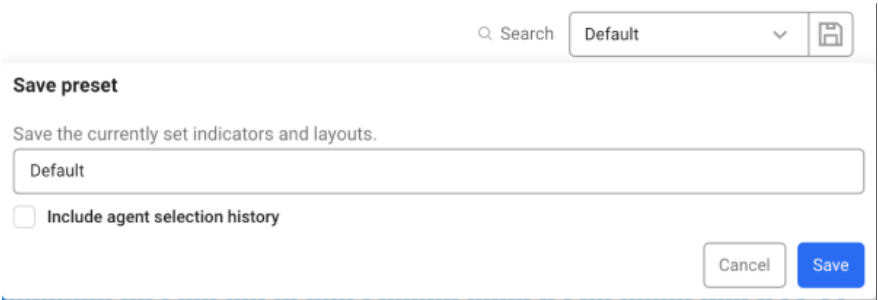
Preset



You can save and load the settings and layouts for widgets on the dashboard. The default preset (**Default**) cannot be changed, but you can create a new preset by resizing and placing widgets in the desired format.

Creating a new preset

- 1. Place the widgets in the desired format on the dashboard. You can also resize and place only the widgets to be frequently checked.
- 2. On the upper right of the screen, select .
- 3. Enter a new preset name.



To save the selected agent data, select **Include agent selection history**.

- 4. Select **Save**.

You can see the newly saved preset from the preset list.

- ⓘ • If any changes are made to the newly created preset, save the preset again. Select  and then save the preset with the same name. Any changes are overwritten in the existing preset.
- If you go to another menu without saving changes on the dashboard, the changes are not saved.
- Presets are saved in your user account and cannot be shared with other users. We plan to update it for better features.

Deleting presets

If you have unused presets, you can delete them from the preset list. Select  on the right of the item to delete from the preset list.

Using the widget options

The features of the icon displayed on the top of the widget are as follows:

- ⓘ: You can see the features and information for major widgets.
- ~ **Merge/view each:** The agent data for the widget is displayed individually or merged in a graph.
-  **Detail:** A modal window appears where you can search the data of the widgets separately by agent.

- ⓘ The provided options may differ depending on the widget.

Learning about widgets

- ⓘ This document mainly describes the widgets placed on the **Application dashboard**. For more information about how to analyze transactions and users, services, and resources through **Application dashboard**, see the following documents:
 - [Exploring the Application Dashboard](#)
 - [HITMAP transaction](#)
 - [DB connection delay and connection pool](#)

Agent-related widgets

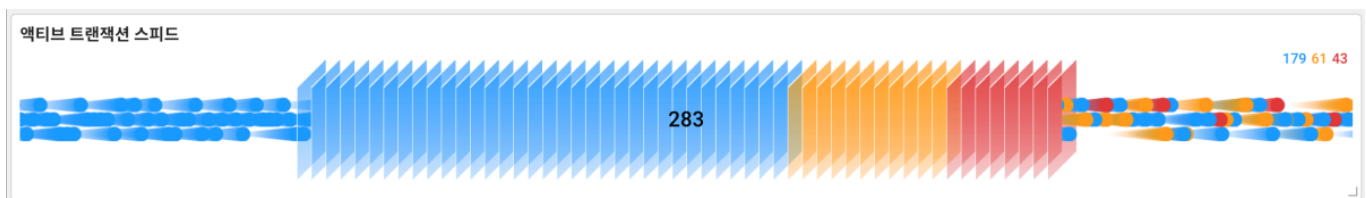
You can see the application information through the agent installed on the application.

Transaction-related widget

The failure status that can be checked through the transaction, can be recognized through the priority response time. If a transaction in progress does not end, this should be considered a failure. WhaTap divides sections for display by the time of the progress status. **Blue (Normal)** means transactions within the specified response time, **Orange (Slow)** means slow transactions with the response time of around 8 seconds, and **Red (Very slow)** means transactions with the response time that is twice as slow as normal. This allows you to intuitively recognize failures as fast as possible.

- ✓ If you add a database project through the **Correlated Project Management** menu for the Java platform, you can see the performance metrics of the project linked in the following widget.
- **Active Transaction Speed Chart, Active Transaction:** You can view the SQL performance of the database associated with the active transaction.
 - **Hitmap:** You can view performance metrics of the database associated with transactions in the past.
- For more information about adding and analyzing the linked project, see [the following](#).

Active Transaction Speed Chart



You can see the status of active transactions (middle area) and completed transactions (right area). The water drop icon passing from left to right means a transaction. If you click a chart area, a pop-up window appears where you can see a list of active transactions. If you select one of the transaction lists, you can view the transaction details, SQL query, and call stack information.

- Area on the left

This indicates the number of requested transactions and displays data in proportion to the number of transactions.

- Area in the middle

The status of active transactions is color displayed by the speed along with the count at the upper right. The number in the chart is the total number of active transactions. The colors in the chart indicate as follows:

- Blue (**Normal**): 0 to 3 seconds
- Orange (**Slow**): 3 to 8 seconds
- Red (**Very Slow**): 8 seconds or more

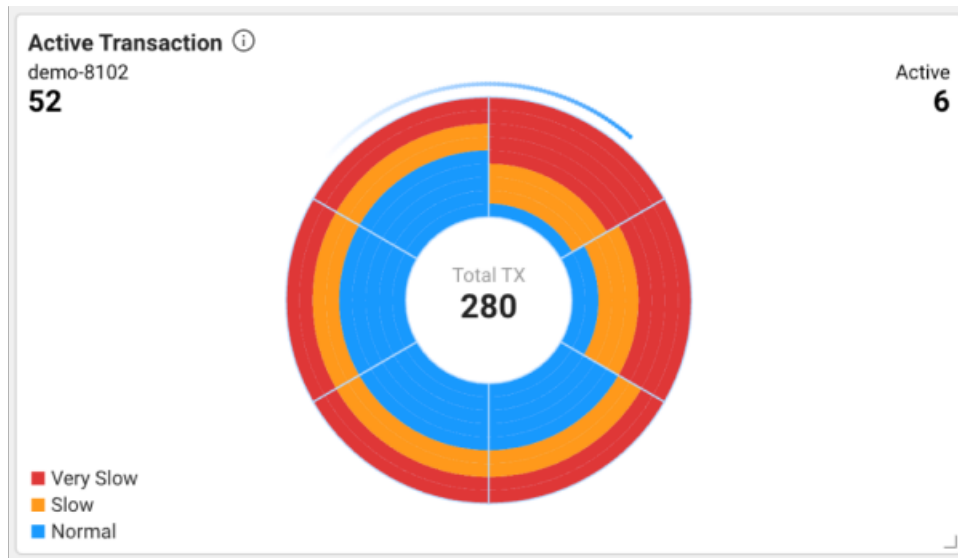
- Area on the right

This indicates the Application Performance Index (Apdex). This metric is measured for customer satisfaction of a web application and has a value between 0 and 1. The droplet flying speed is proportional to the total number of Apdex transactions, and the droplet colors mean as follows:

- Blue: **Apdex** satisfied count
- Orange: **Apdex** tolerating count
- Red: **Apdex** frustrated count = $\text{Total number of APDEX transactions} - (\text{Apdex satisfied count} + \text{Apdex tolerating count})$

❗ For more information about the Application Performance Index (Apdex), see [the following link](#).

Active Transaction



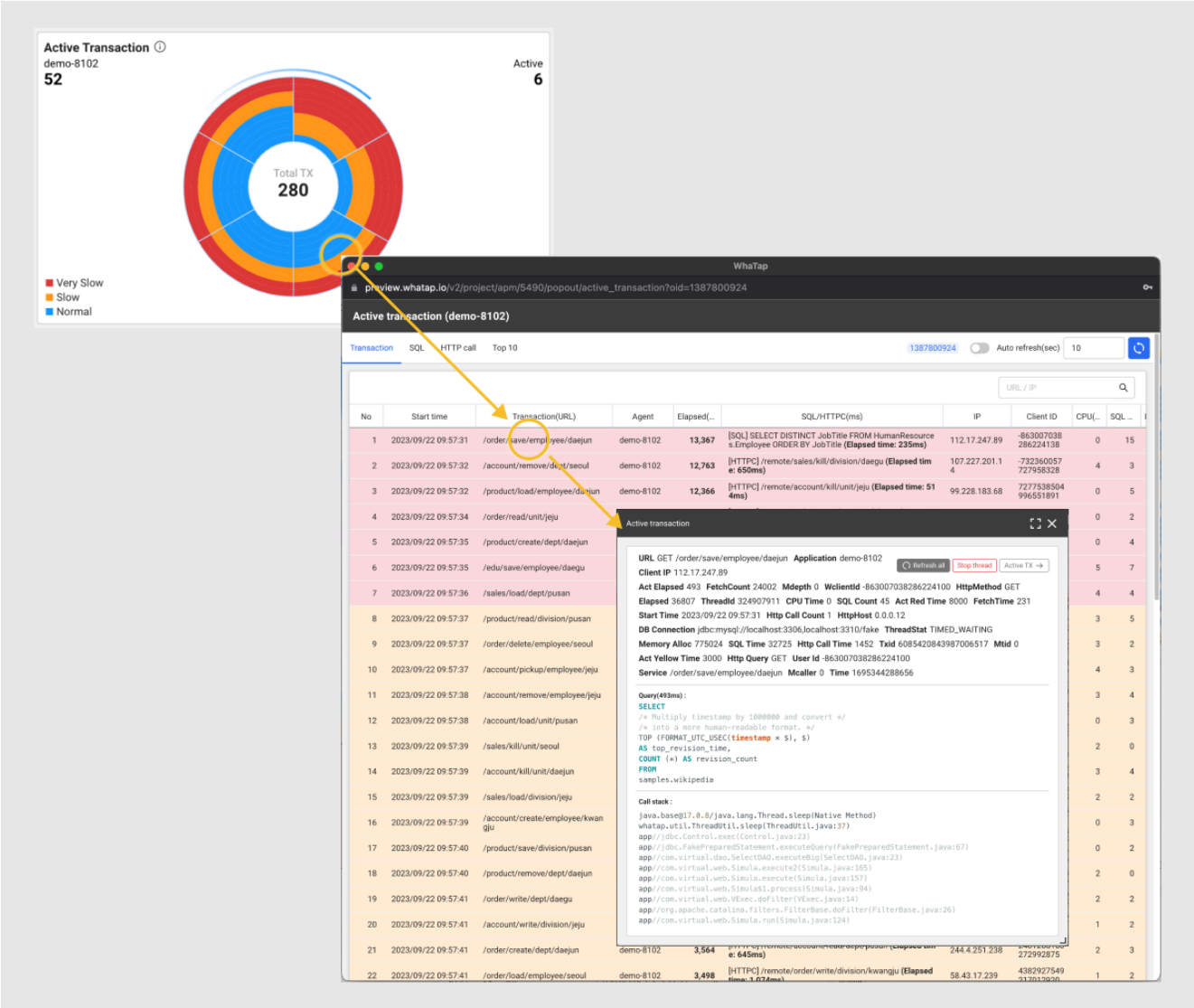
In the **Active Transaction** widget, the transactions in progress are displayed in the **arc equalizer** chart. You can see the status of delays by agent. The arc is divided by the number of agents. By representing the number of requests being processed by the server every 5 seconds, you can tell how long the request has been processed. Requests detected at an interval of 5 seconds are classified by color as follows to determine whether they are dangerous or not.

- Red (**Very Slow**): 8 seconds or more
- Orange (**Slow**): 3 to 8 seconds
- Blue (**Normal**): 0 to 3 seconds

The situation where lots of blue colors are represented is not a problem. You must watch the trend to see whether some of them turn to orange or red. In case of a failure, the number of active transactions increases and the percentage of red color increases. On the other hand, a very responsive system with a high number of process transactions (TPS) might have a low number of active transactions.

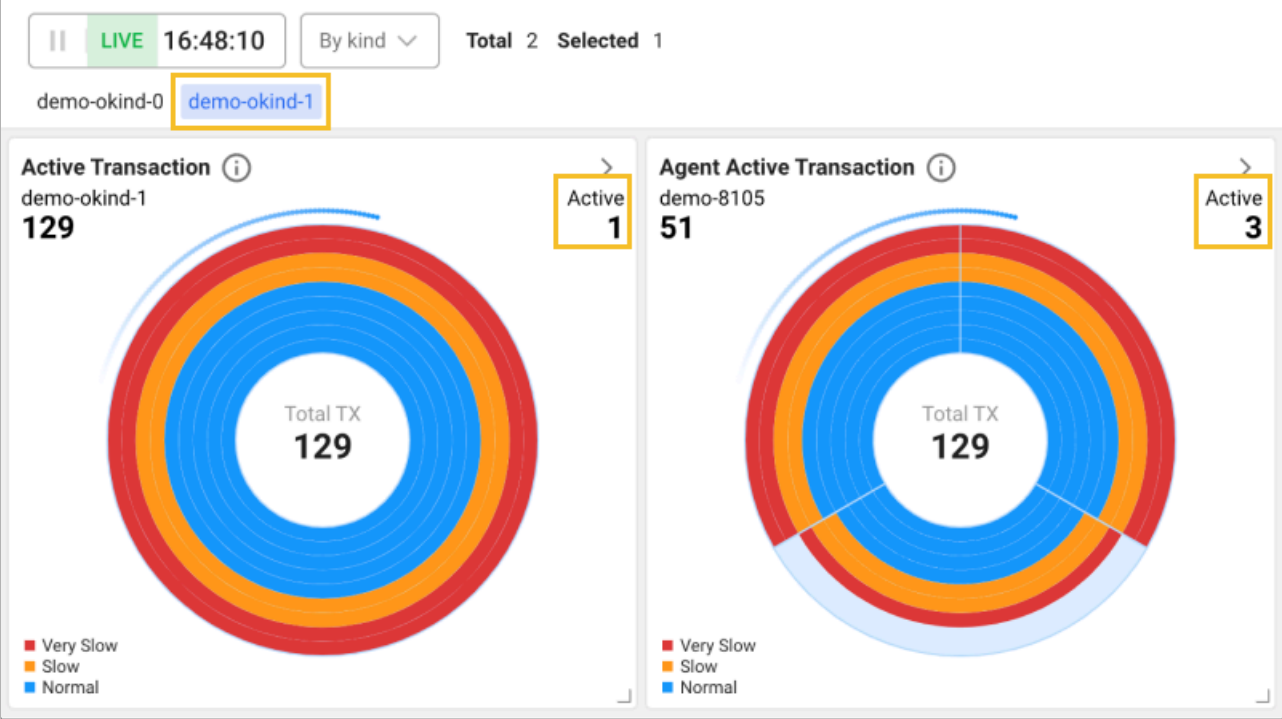
- On the upper left, the agent name with the most active transactions appears.
- The middle number is a sum of active transactions of all agents.
- Two bars around active transactions rotate around the chart at a three-step rate based on the throughput (TPS).
- When you select a divided arc area, a pop-up window appears listing the transactions in progress on the

agent. To check the transaction details, select an item in the transaction list.



- If the item selected in the transaction list is a transaction that has already ended, select **Show trace**. The **Transaction information** window appears. For more information about the **Transaction information** window, see [the following](#).
- For more information about transactions, see [the following](#).

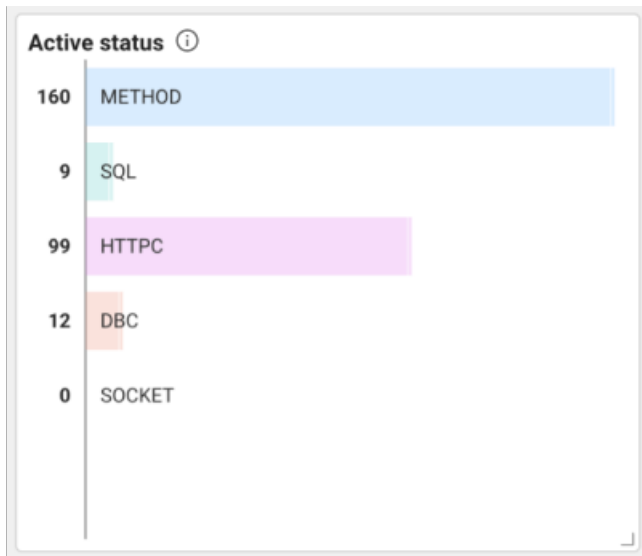
Agent Active Transaction



The **Active Transaction** widget displays on a chart the data merged based on the criteria selected in the agent selection options. On the other hand, the **Agent Active Transaction** widget allows you to individually check the agents included in the selected agent classification criteria.

Right-click the dashboard and then select the **Agent Active Transaction** widget from the widget list.

Active status



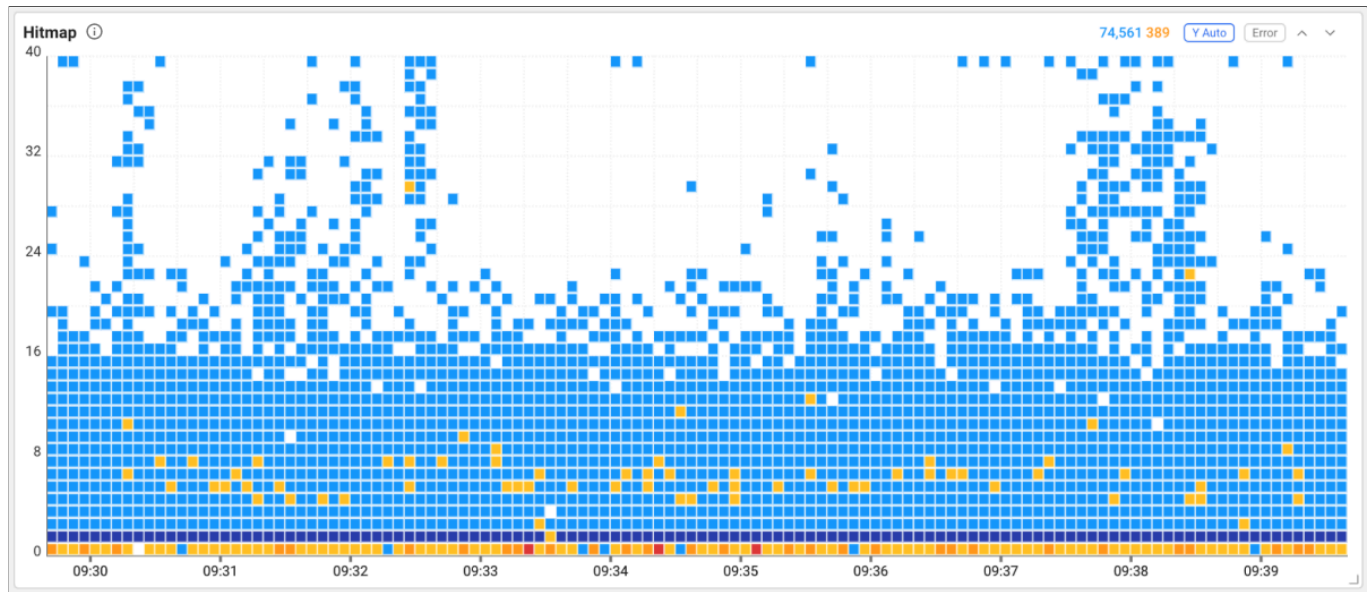
All agents in the project collect the active transaction count every 5 seconds, along with the progress status. The progress status is classified into **METHOD**, **SQL**, **HTTPC**, **DBC**, and **SOCKET**.

- **METHOD**: Status in which the method is executing. Normal situations are displayed in blue.
- **SQL**: SQL is running. If the **SQL** number increases, check for problems connecting to the database server.
- **HTTPC**: It is calling any external API. If the **HTTPC** number increases, it means that the server connected to outside is not responding normally.
- **DBC**: The transaction is attempting to get a new connection from the connection pool. If DB connection pools are insufficient, a delay occurs whenever a new connection request occurs, causing the performance failure. In this case, the **DBC** number increases.

! For more information about the DB connection delay, see [the following](#).

- **SOCKET**: Indicates the TCP connection attempt to an external system. If the **SOCKET** number continues to increase, there is a high possibility of failures due to disconnection of the external system.

Hitmap



The **Hitmap** chart represents the distribution of response times of completed transactions. The horizontal axis is the transaction end time. The vertical axis is the execution time. Transactions with longer execution times are at the top of the distribution. If you drag the chart area in **Hitmap**, the **Trace analysis** window appears that you can see the transaction information details. For more information about **Trace analysis**, see [the following](#).

- Normal transactions are displayed in light blue → blue → Indigo in order.
- Erroneous transactions are displayed in yellow → orange → red in order.
- The number on the upper right indicates the [transaction count](#)/[error count](#) in the chart.
- If **Y Auto** is selected, the Y-axis (execution time) is automatically adjusted.
- If **Error** is selected, only the error transactions appear.
- Select ^ or v to zoom in/out the chart from 5 seconds to 80 seconds.

- ! • The detailed information for the transactions with no error whose execution time is less than 500 ms is collected once every 5 minutes for each URL.
- It does not affect the statistics such as TPS and average response time.
- You can set a policy through the agent configuration. For more information, see [the following](#).
- See the following document for the procedure to analyze transactions through the hitmap widget.

Service-related widget

The service-related widget can be used as the metric for application performance tuning.

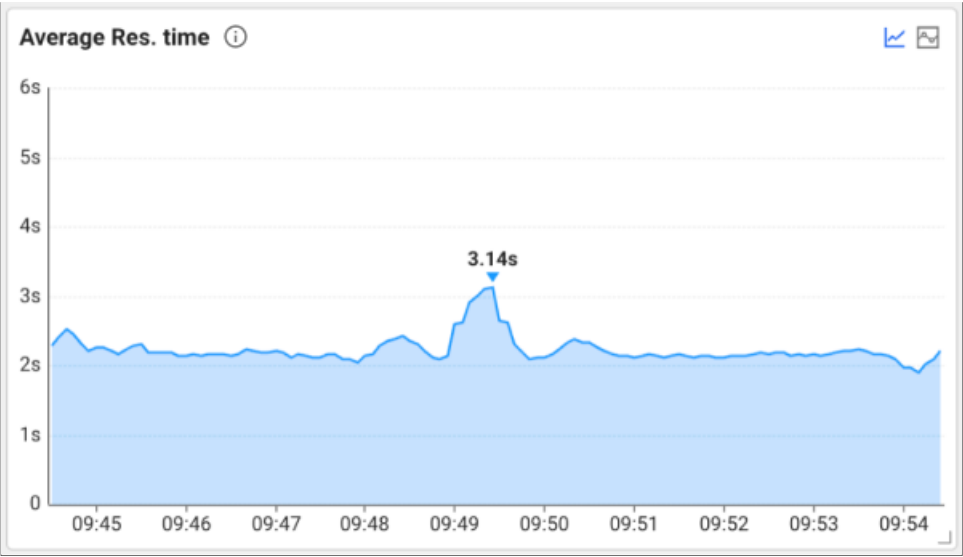
TPS



Transaction Per Second (TPS) indicates the number of transactions processed per second and is the reference for service performance metric. The number of transactions processed every 5 seconds is converted to the number of transactions per second and is expressed with a chart. It displays TPS for the last 10 minutes.

! For more information about analyses for TPS, response time, and CPU utilization, see [the following](#).

Average Res. time

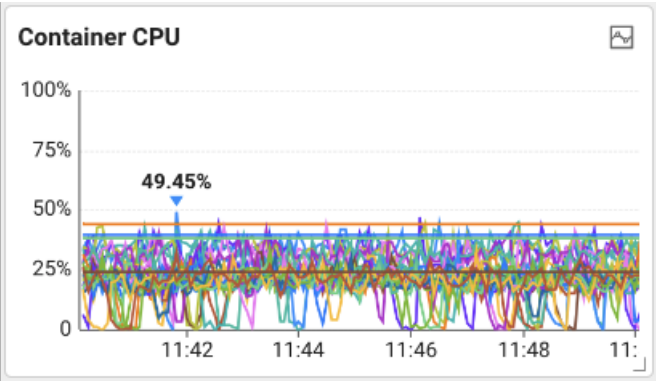


It monitors the times needed for the application server to return the request result. **Average Res. time** is meaningful as a tuning metric. The WhaTap's monitoring service calculates the average response time for transactions every 5 seconds.

ⓘ For more information about analyses for TPS, response time, and CPU utilization, see [the following](#).

Resource-related widget

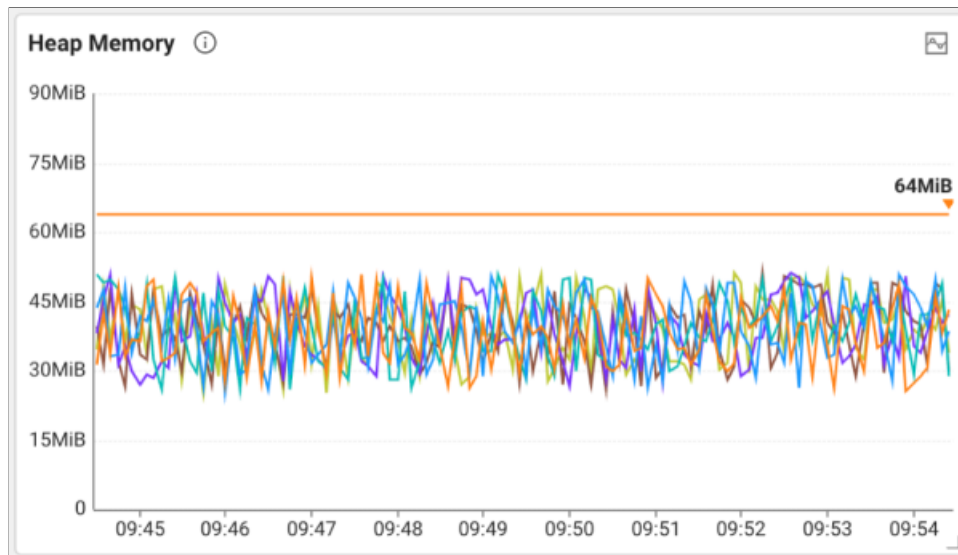
Container CPU



It is the container's CPU usage. You can see the change trend in CPU usage in real time.

! For more information about analyses for TPS, response time, and CPU utilization, see [the following](#).

Heap Memory



You can see which servers are at risk by displaying the maximum available memory and the current memory for each server. The changes in memory usage over time can be seen in real time.

The memory line chart usually ripples continuously. It increases when using memory to process the requests from the application server. It decreases when memory is cleaned up through GC (Garbage Collection). **Heap Memory** metric is displayed with average values.

- ! • Java and Node.js display the total heap memory and usage.
- PHP, Python, and .Net display the process memory usage.

✓ For more information about the heap memory and its chart analysis, see the following links.

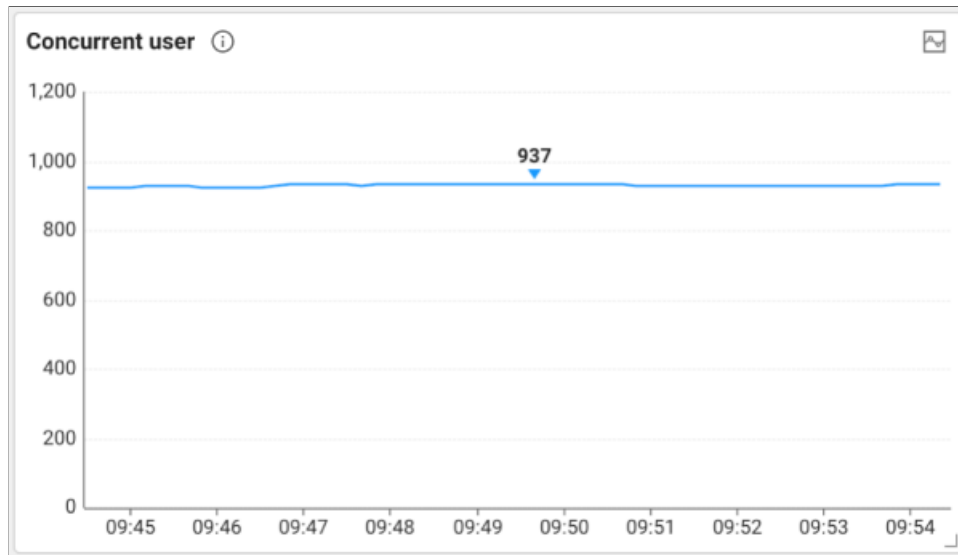
- [Learning: Heap memory metrics and metrics alert configuration](#)
- [Analysis of Java heap memory chart: Observing the Ch.1 heap chart](#)



- [Analysis of JAVA heap memory chart: Analysis of Ch.2 memory leak and heap dump](#)

User-related widget

Concurrent user



It displays the number of real-time browser users. Every 10 seconds, users who made transactions within the last 5 minutes are counted and displayed without duplicates. Users are counted based on their browser's IP.

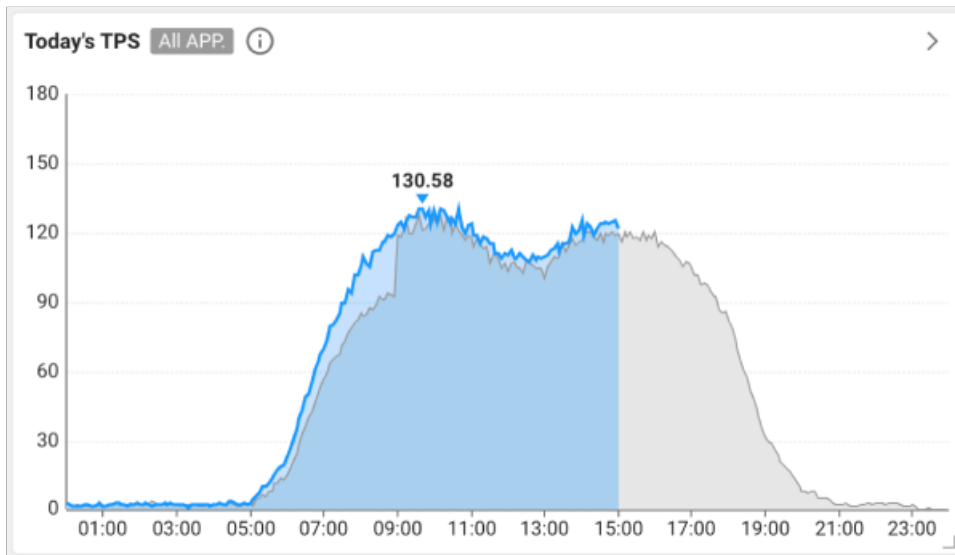


- In general, the connected user means a connected user who has access to the current network. However, web systems are different because they use connectionless networks. It does not mean that you are connected, but rather the user who sent the recent request can be measured.
- Real-time (concurrent access) users indicate the users who are using the system in the same time zone. There are various ways to count or measure real-time (concurrent access) users.
- The number of users is calculated by **HyperLogLog** rather than simply adding up. **HyperLogLog** is a probabilistic data structure that estimates the number of elements in a set with very little memory.
- For more information about the configuration of agent options for aggregating the user count, see [the following](#).

Comparison widget on a daily basis

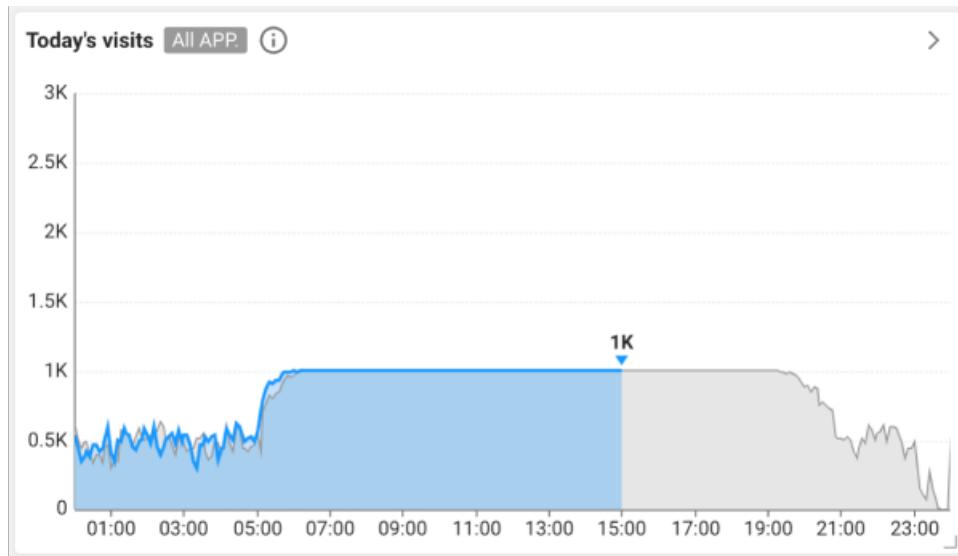
! The widgets marked with the **All APPS.** tag display the data for all agents.

Today's TPS



It displays the TPS for today by hour. You can check it by comparing with the TPS (gray) in the same time zone of yesterday. If you click **>**, **Stat/Report > Daily APP. Status** appears.

Today's visits



This metric shows the number of **Concurrent user** for the day by time zone. You can check it by comparing with the user count (gray) in the same time zone of yesterday. If you click **>**, **Stat/Report > Daily APP. Status** appears.

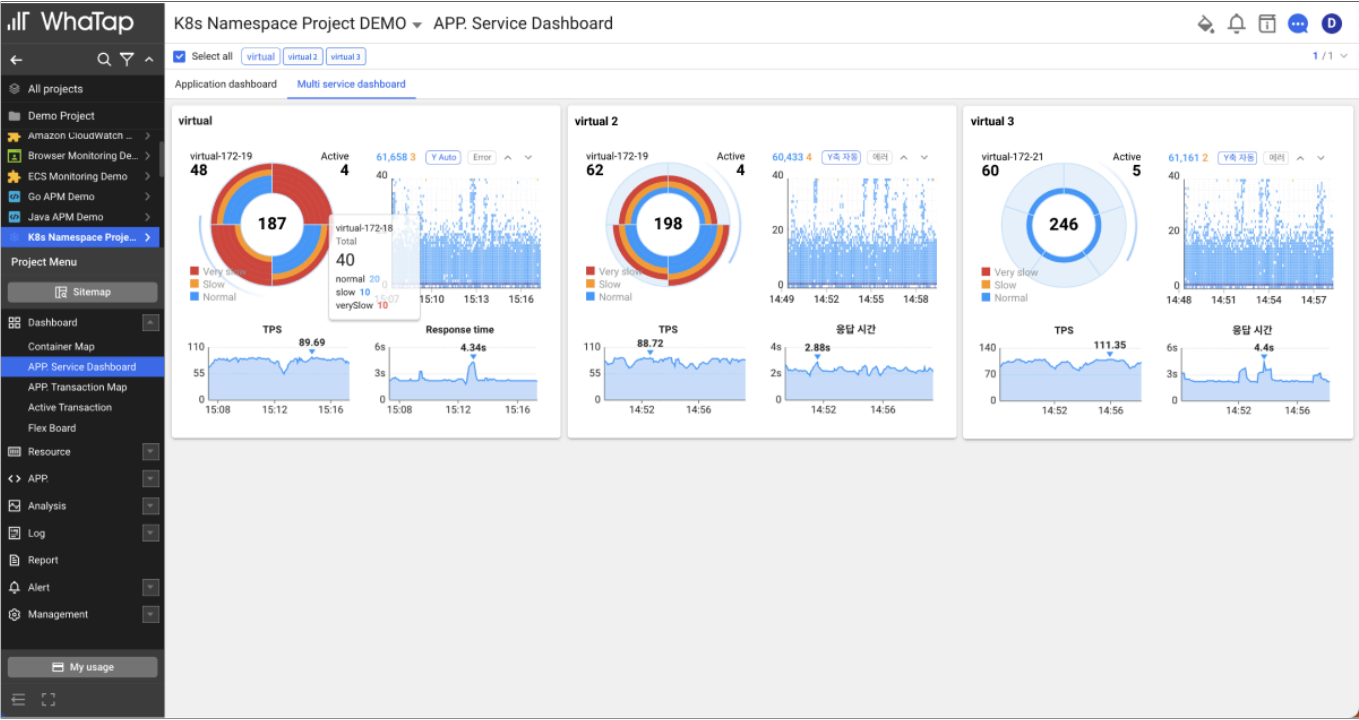
Checking the real-time alert

If you select **Alert Status** on the upper right of the screen, the recent events appear. You can turn on or off the browser notifications by clicking the toggle menu.

! For more information about basic elements in the fixed menu area at the top of the screen, see [the following](#).

Multi Service Dashboard

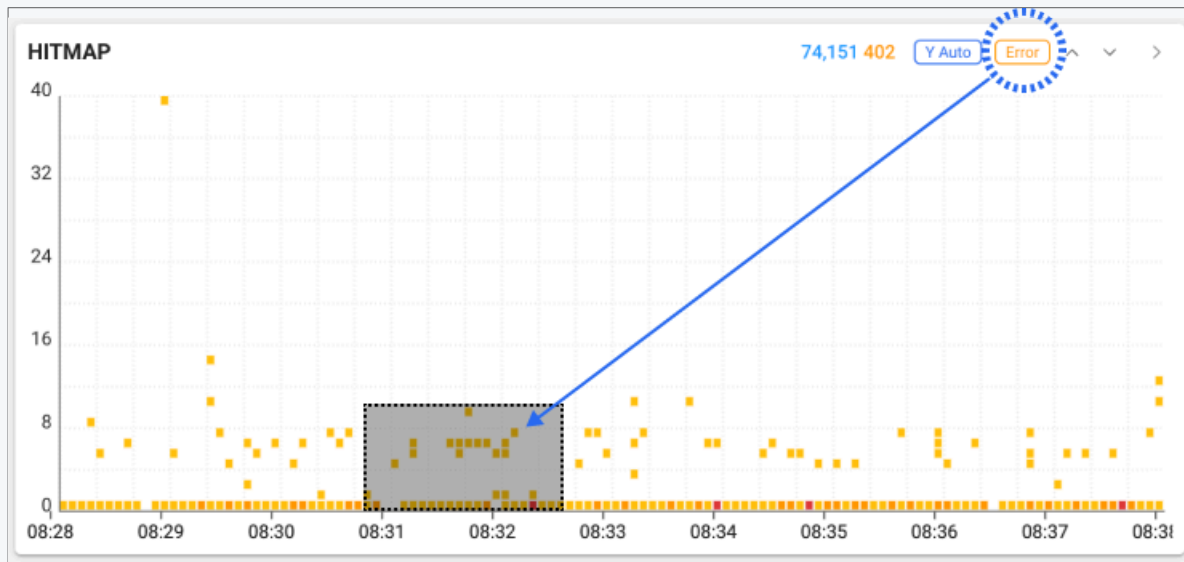
You can view and compare the selected applications in the individual card format.



HITMAP - Trace Analysis

Through the **Hitmap** widget in **Application Dashboard**, you can open the **Trace analysis** window to check the transaction list and step information of each transaction at once. Drag the desired area on the hitmap widget. The **Trace analysis** window appears that contains the transaction list and sub-steps of the selected area.

- ✓ To classify and see only the transactions with errors, select **Error** on the upper right of the **Hitmap** widget in **Application Dashboard** and then drag the chart area.



Basic screen guide

| Trace analysis 15 cases have been inquired. 1 | | | | | | | | | | | | | | Copy URL | |
|--|----|--------------|---------------------|--------------|------------|-----------------------|-----------------------|----------------------|-----------|----------|-----------------|--------------------------|--------------|----------|--|
| All A Active Stack M Multi-transaction Error 2 | | | | Topology | | Column Settings | | TXT | | CSV | | Please enter your sea... | | 3 | |
| 4 | No | Agent nam... | Transaction | Elapsed time | Start time | End time | HTTP call ... | HTTP call time (...) | SQL count | SQL time | SQL fetch co... | SQL patch tim... | DB CONN. tin | | |
| | 1 | demo-8100 | /account/load/... | A | 61,290 | 26/04/06 14:16:32.419 | 26/04/06 14:17:33.709 | 22 | 34,254 | 12 | 71 | 1,541 | 16 | 4 | |
| | 2 | demo-8100 | /sales/write/div... | A | 60,619 | 26/04/06 14:16:23.781 | 26/04/06 14:17:24.400 | 22 | 31,405 | 12 | 301 | 2,456 | 39 | 3 | |
| | 3 | demo-8100 | /account/save/... | A | 59,912 | 26/04/06 14:16:28.263 | 26/04/06 14:17:28.175 | 24 | 31,313 | 12 | 753 | 1,877 | 24 | 8 | |
| | 4 | demo-8100 | /account/kill/d... | M A | 58,395 | 26/04/06 14:16:45.099 | 26/04/06 14:17:43.494 | 21 | 30,937 | 12 | 570 | 2,093 | 16 | 27 | |
| | 5 | demo-8100 | /order/create/d... | A | 57,584 | 26/04/06 14:16:33.561 | 26/04/06 14:17:31.145 | 20 | 28,965 | 18 | 30 | 1,808 | 29 | 13 | |
| | 6 | demo-8100 | /product/create... | A | 57,473 | 26/04/06 14:16:45.165 | 26/04/06 14:17:42.638 | 23 | 30,910 | 15 | 17 | 2,429 | 25 | 22 | |
| | 7 | demo-8103 | /sales/save/uni... | A | 57,184 | 26/04/06 14:17:15.132 | 26/04/06 14:18:12.316 | 13 | 15,398 | 515 | 387 | 248,494 | 3,219 | 5,67 | |
| | 8 | demo-8100 | /edu/load/unit/... | A | 56,959 | 26/04/06 14:16:27.801 | 26/04/06 14:17:24.760 | 22 | 27,989 | 16 | 161 | 1,283 | 17 | 22 | |
| | 9 | demo-8100 | /edu/create/uni... | A | 56,045 | 26/04/06 14:16:27.219 | 26/04/06 14:17:23.264 | 17 | 27,168 | 12 | 141 | 2,187 | 29 | 34 | |
| | 10 | demo-8100 | /account/creat... | A | 54,499 | 26/04/06 14:16:48.751 | 26/04/06 14:17:43.250 | 19 | 26,299 | 12 | 10 | 2,356 | 23 | 2 | |

- **1 Transaction lookup count:** You can see the number of transaction lists in the dragged area on the hitmap widget. You can query up to 1000 cases.
- **2 Transaction classification:** By separating **A Active Stack**, **M Multi-transaction**, and **Error**, you can view the list.
- **3 Transaction search:** You can search the transaction list by transaction name, agent name, and error message.
- **4 Sorting the list:** You can sort the lists of transactions by the selected item by clicking a header column in the table.
- The transactions with errors appear in red in the list.

- ✓ You can place, add, or hide desired columns to the front of the list. For more information, see [the following](#).
- If you drag another area in the **Hitmap** widget, a new trace analysis window appears. You can check it by comparing with the previous trace analysis window.
- The transactions that are viewed can be shared with other users. With the transaction item selected (a), click  **Copy URL** on the trace analysis window. Pass the copied URL address to another user. Any user who has received the URL address can quickly see the same items by entering the URL address in the browser address bar.

Trace analysis 1000 cases have been inquired. ⓘ

Record summary Table view Tree view Methods SQLs HTTP calls Transaction log

/account/kill/employee/daegu

Start : 08/02 08:25:44.017 End : 08/02 08:25:46.409 Elapsed : 2,392ms

Agent name (oname) : star_star Client IP : 2.184.24.254

| No | Agent nam... | Transaction | Elapsed time | Start time | End time |
|----|--------------|--------------------|--------------|-----------------------|-----------------------|
| 5 | demo-8103 | /product/write/... | 2,416 | 24/08/02 08:25:41.128 | 24/08/02 08:25:43.544 |
| 6 | star_star | /account/kill/e... | 2,392 | 24/08/02 08:25:44.017 | 24/08/02 08:25:46.409 |
| 7 | demo-8105 | /account/delet... | 2,371 | 24/08/02 08:25:40.383 | 24/08/02 08:25:42.754 |

Analyzing a trace

Select an item to view details from the transaction list. On the right of the screen, the detailed analysis of the selected transaction appears. In this state, if another transaction item is selected, the detailed analysis of the selected transaction item is changed.

To analyze the transaction performance, WhaTap collects and displays the properties such as name and client data, transaction performance, and detailed execution records for each section.

Trace analysis 378 cases have been inquired. ⓘ

Record summary Table view Tree view Multi transaction Active stack Methods SQLs HTTP calls Transaction log

/account/(aa)/(bb)/(cc)

Start : 08/02 09:32:20.276 End : 08/02 09:32:51.540 Elapsed : 31,264ms Agent name (oname) : demo1

Client IP : 193.209.234.170

| No | Agent nam... | Transaction | Elapsed time | Start time | End time | HTTP call ... | HTTP call time |
|----|--------------|---------------------|--------------|-----------------------|-----------------------|---------------|----------------|
| 15 | demo3 | /account/(aa)/(... | 31,406 | 24/08/02 09:33:26.029 | 24/08/02 09:33:57.435 | 22 | 25 |
| 16 | demo2 | /account/(aa)/(... | 31,276 | 24/08/02 09:32:31.982 | 24/08/02 09:33:03.258 | 14 | 16 |
| 17 | demo1 | /account/(aa)/(... | 31,264 | 24/08/02 09:32:20.276 | 24/08/02 09:32:51.540 | 25 | 3 |
| 18 | demo6 | /account/(aa)/(... | 31,176 | 24/08/02 09:32:08.278 | 24/08/02 09:32:39.454 | 3 | 2 |
| 19 | demo5 | /account/(aa)/(... | 31,132 | 24/08/02 09:32:36.202 | 24/08/02 09:33:07.334 | 6 | 8 |
| 20 | demo4 | /account/(aa)/(... | 31,055 | 24/08/02 09:32:06.308 | 24/08/02 09:32:37.363 | 22 | 30 |
| 21 | demo4 | /account/(aa)/(... | 31,041 | 24/08/02 09:31:53.398 | 24/08/02 09:32:24.439 | 21 | 25 |
| 22 | demo2 | /account/(aa)/(... | 30,853 | 24/08/02 09:32:38.313 | 24/08/02 09:33:09.166 | 15 | 20 |
| 23 | demo3 | /sales/save/de... | 30,743 | 24/08/02 09:32:59.481 | 24/08/02 09:33:30.224 | 25 | 25 |
| 24 | demo6 | /product/delete... | 30,393 | 24/08/02 09:33:32.787 | 24/08/02 09:34:03.180 | 19 | 28 |
| 25 | demo2 | /product/write/... | 30,370 | 24/08/02 09:32:50.389 | 24/08/02 09:33:20.759 | 15 | 20 |
| 26 | demo2 | /order/kill/divi... | 30,312 | 24/08/02 09:31:48.304 | 24/08/02 09:32:18.616 | 22 | 25 |
| 27 | demo3 | /product/create... | 30,285 | 24/08/02 09:32:49.944 | 24/08/02 09:33:20.229 | 22 | 28 |
| 28 | demo1 | /product/kill/un... | 30,262 | 24/08/02 09:33:03.011 | 24/08/02 09:33:33.273 | 6 | 8 |
| 29 | demo6 | /edu/read/divis... | 30,194 | 24/08/02 09:32:51.435 | 24/08/02 09:33:21.629 | 24 | 25 |
| 30 | demo2 | /sales/pickup/... | 30,165 | 24/08/02 09:32:10.276 | 24/08/02 09:32:40.441 | 22 | 25 |
| 31 | demo3 | /sales/read/de... | 30,093 | 24/08/02 09:33:25.545 | 24/08/02 09:33:55.638 | 23 | 28 |
| 32 | demo2 | /sales/read/em... | 30,090 | 24/08/02 09:32:48.950 | 24/08/02 09:33:19.040 | 14 | 18 |
| 33 | demo3 | /product/load/... | 29,992 | 24/08/02 09:32:08.434 | 24/08/02 09:32:38.426 | 17 | 18 |
| 34 | demo3 | /order/kill/unit... | 29,920 | 24/08/02 09:32:02.357 | 24/08/02 09:32:32.277 | 12 | 12 |
| 35 | demo2 | /account/(aa)/(... | 29,882 | 24/08/02 09:33:35.839 | 24/08/02 09:34:05.721 | 21 | 28 |

Method DB Connection SQL HTTP Call Socket Active Stack

Select column TXT SQL %a %s

| No | Time | Gap | Elapsed | Contents |
|----|--------------|-------|---------|--|
| 1 | 09:32:20.276 | | | Start 2024-08-02 09:32:20.276 |
| 2 | 09:32:20.276 | | | HTTP-HEADERS
host=255.255.255.221
gid=gid.23dab7cb-6ad4-
referer=http://www.test.com/account/pickup/de
pt/daejun |
| 3 | 09:32:20.276 | 0 | 31,264 | com.virtual.web.VExec#doFilter |
| 4 | 09:32:20.276 | 0 | 31,264 | com.virtual.web.Simula\$1#process |
| 5 | 09:32:20.276 | 0 | 2,331 | com.virtual.web.Simula#exec |
| 6 | 09:32:20.276 | 0 | 2,267 | com.virtual.dao.RemoteDAO#execute |
| 7 | 09:32:20.276 | 0 | 2,267 | [ApacheClient] 127.0.0.1:8094 /remote/ |
| 8 | 09:32:20.276 | | | [Socket] 127.0.0.1:8094 |
| 9 | 09:32:22.543 | 2,267 | 0 | com.virtual.TestLogger#printin |
| 10 | 09:32:22.543 | 0 | 5 | com.virtual.dao.SelectDAO#execute2 |
| 11 | 09:32:22.543 | 0 | 1 | MYSQL jdbc:mysql://localhost:3306/localho... |
| 12 | 09:32:22.544 | 1 | 0 | com.virtual.TestLogger#printin |

- In the 5 area, the step data can be seen in various formats. Select a desired type of tab.
- In the 6 area, you can see the basic information for transactions. The information matches the columns in

the list on the left.

- : The statistical data window for the corresponding URL or each execution section appears. Click a desired time on the graph chart in the statistical data window to go to **Statistics** or **Transaction Search**. You can view the statistical data based on the selected time.
- By dragging  to the left or right, you can adjust the screen size. This is useful when you want to see more information.

-  • For more information about **Statistics**, see [the following](#).
- For more information about **Transaction Search**, see [the following](#).
 - If you click  or , you can change the transaction list or analysis details to full screen.

The collected types of **Step** are as follows. The diagrams and texts are color coded.

- **DB Connection** step **START-END**

It includes the performance for connection to RDB. Step information includes the name, response time, and error.

- **SQL** step **START-END**

It includes the performance for JDBC SQL. Step information includes connection data, SQL statements, and errors.

- **HTTP Call** step **START-END**

It includes the performance for the outbound HTTP service call. Step information includes URL, host, port, response time, and errors.

- **Message** step **ADD**

While collecting traces, the message steps are used to collect the records of all irregular sections.

- **Socket** step **ADD**

Step to represent the socket open.

- **Method** step **START-END**

The method response time is traced.

- **Active Stack** step **START-END**

It includes the active stack information. It is created and collected by a separate thread and it is added to the transaction trace.

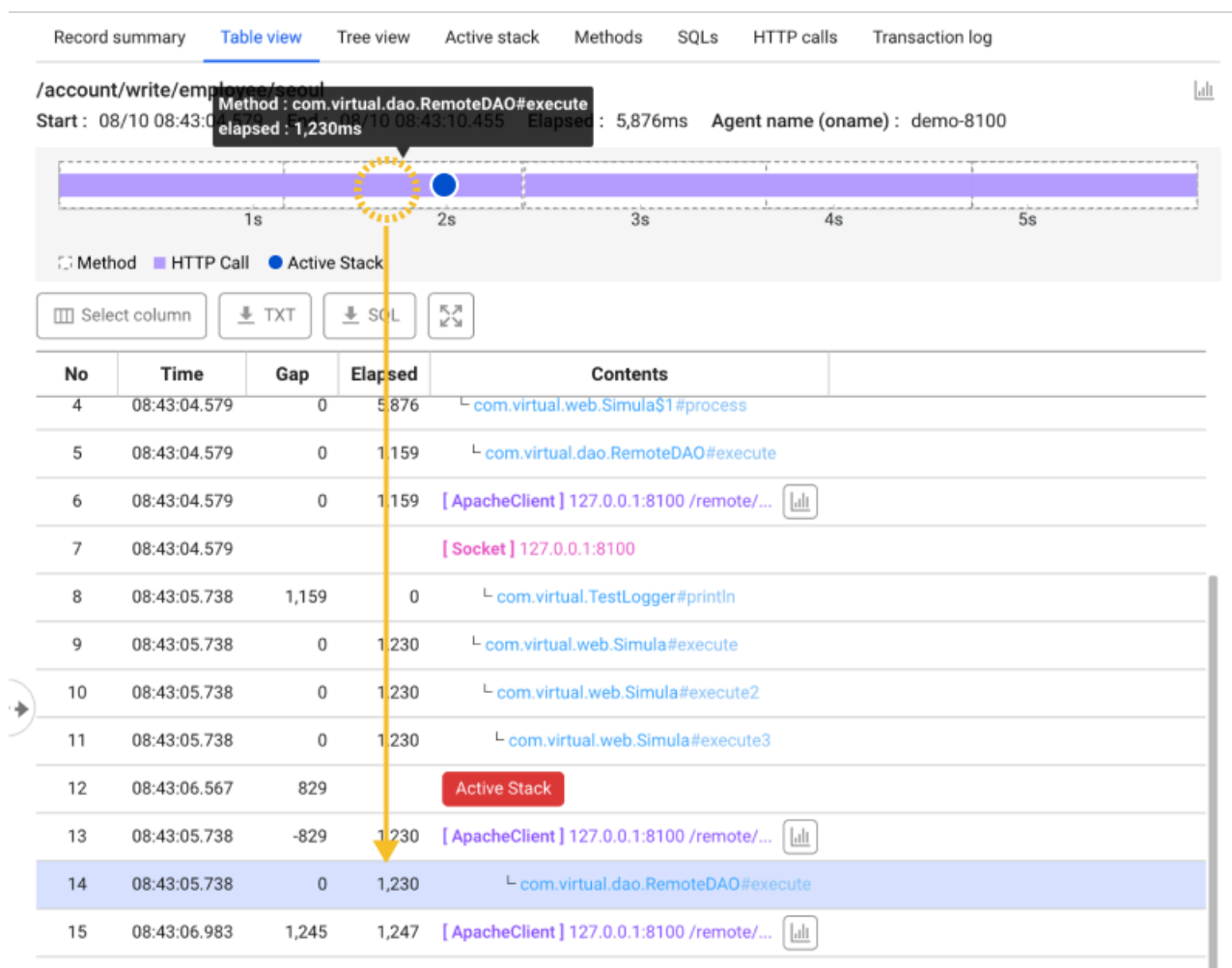
✔ If you add a database project through the **Correlated Project Management** menu for the Java platform, you can see the performance metrics of the linked project. For more information about adding and analyzing the linked project, see [the following](#).

ⓘ • In the WhaTap monitoring service, **Step** is used as the same meaning as **Span**.
• To see the updates in the **Trace analysis** function released on July 25, 2023, see [the following](#).

Table view

In the **Table view** tab, you can see the transaction execution steps in chronological order.

- You can classify transactions by each execution section and see them in a diagram. Out of all elapsed times, you can quickly find the times spent for each section and the section that took the longest.
- If you select each section in the diagram, the corresponding table list appears where the step is located.



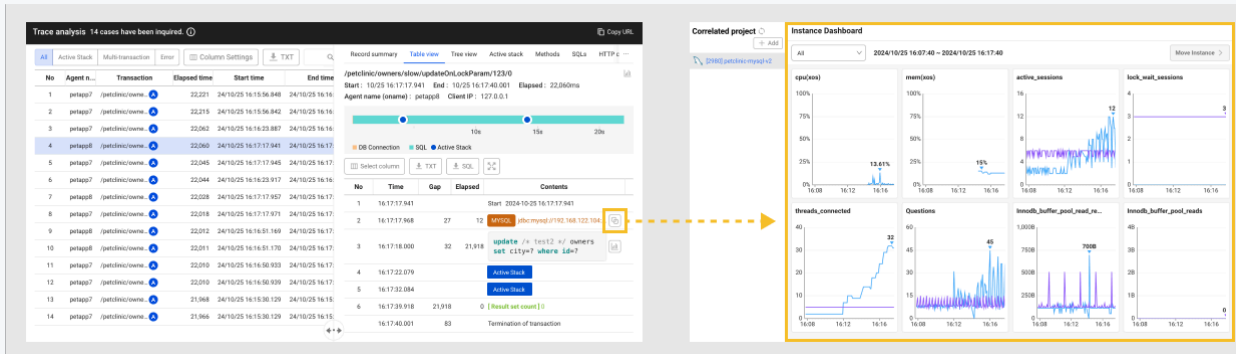
- The area where the red circle (●) is located, is the moment when the active stack has been collected. If you select the red circle, the table list appears where the **Active Stack** button is located. If you click **Active Stack**, you can check the step performed during the section.

- In the transaction list, you can see them from the items marked with the **A** icon.
 - For more information about active stacks, see [the following](#).

- Select column:** You can add or hide the memory accumulation and CPU accumulation data as columns in the table list. The columns are as follows:
 - No:** Step occurrence sequence

- **Time:** Start time of each step
- **Gap:** The waited time between the start time of the previous step and the current step may have a difference, while there may be a gap from the elapsed time if a delay happens due to external factors.
- **Elapsed:** Total elapsed time from the start to the end of each method
- **Contents:** Execution details of the step
-  **TXT:** You can download basic transaction data and execution data for each section in a text file.
-  **SQL:** The basic transaction information and SQL execution data can be downloaded as a text file.
 - **Download with substitute value:** It downloads the SQL statistics data after replacing the bind variable value with its original one. If a security key (`paramkey`) is set, the security key input window appears. You must enter the security key for downloading. For more information about the agent configuration related to the security key, see [the following](#).
 - **Download without substitute value:** It downloads SQL statistics data without replacing the bind variable value.
- : You can see only the table list, with hiding the transaction summary and diagrams. If you select , the hidden information reappears. This feature is useful when you have a long list of tables.
- If you select the SQL step, the **SQL** window appears to display parameters. For more information, see [the following](#).

- ✓ If you add a database project through the **Correlated Project Management** menu for the Java platform, you can see the performance metrics of the project linked in the **DB connection step**.



For more information about adding and analyzing the linked project, see [the following](#).

Record summary

To see the basic information for transactions, select the **Record summary** tab. You can see the basic information of the transaction, agent on which the transaction was performed, memory allocation usage, and client information.

| Record summary | | | |
|--|--|-------------------|---------------------------|
| Table view Tree view Methods SQLs HTTP calls Transaction log | | | |
| Transaction | /account/save/division/daejun | | |
| Original URL | | | |
| Project code | 5490 | Status | 200 |
| Error level | Warning | Error class | Internal RuntimeException |
| Error message | Internal RuntimeException | | |
| Elapsed time | 6,638ms | | |
| Start time | 24/09/24 15:49:52.317 | End time | 24/09/24 15:49:58.955 |
| User agent | Mozilla/5.0 (X11; U; Linux i686; en-US; rv:1.3) Gecko/20030320 | | |
| Transaction ID | -8541160086278366865 | | |
| Agent name (oname) | demo-8101 | Agent ID (oid) | -877561626 |
| Agent group name | demo-okind-1 | Agent group ID | -628198688 |
| Agent server name | node-1 | Agent server ID | 334634079 |
| HTTP method | GET | SQL time | 918ms |
| HTTP call time | 4,305ms | SQL count | 8 |
| HTTP call count | 4 | SQL patch time | 16ms |
| DB CONN. time | 249ms | SQL fetch count | 1,065 |
| CPU usage time | 0ms | Memory allocation | 623,136byte |
| Client IP | 152.4.69.63 | Operating system | Linux |
| Domain | 255.255.255.196 | Country/Region | US |
| City | Chapel Hill | Client type | Other |
| Client name | Other | WClientID | -1744550593 |
| methodCount | 0 | methodTime | 0 |

| Classification | Properties | Description |
|----------------|-------------|---|
| Transaction | Transaction | API or URL call executed by the application, and the path of the transaction. It converts to a specific format using regular expressions. |

| Classification | Properties | Description |
|----------------|----------------------|--|
| | Original URL | It is the URL path that was actually called. A concrete request path that does not include variables. If there is an original URL, the original URL appears. If there is no original URL, it is not displayed. |
| | Project code | Identification code of the project registered in the WhaTap monitoring service. |
| | Status | HTTP response status code for the transaction. |
| | Error level | Level of the error that occurred in the transaction. It is displayed as Warning or Critical. |
| | Error class | Class of the error that occurred in the transaction. |
| | Error message | Message of the error that occurred in the transaction. |
| | Elapsed time | Transaction execution time. |
| | Start time | Transaction start time. |
| | End time | Transaction end time. |
| | User agent | Browser used to extract client-related data. |
| | Referer | Page address or incoming path used by the browser of the client. |
| | Transaction ID | It is a unique ID of the transaction. |
| | Multi transaction ID | It is a unique ID of the multi-transaction. It appears when executed in parallel with transactions of other applications. |
| Agent | Agent name (oname) | Agent name |

| Classification | Properties | Description |
|----------------|-------------------|--|
| | Agent ID (oid) | It is a unique ID of the agent. |
| | Agent group name | Agent group name set with the <code>okind</code> option. |
| | Agent group ID | It is a unique ID of the agent group. |
| | Agent server name | Agent server name set with the <code>ondoe</code> option. |
| | Agent server ID | It is a unique ID of the agent server. |
| Step | HTTP method | HTTP method. (GET, POST, PUT, HEAD, etc.) |
| | HTTP call time | Outbound HTTP call time. |
| | HTTP call count | Number of outbound HTTP calls. |
| | DB CONN. time | Time taken to connect to the database. |
| | SQL time | SQL execution time. |
| | SQL count | SQL execution count. |
| | SQL patch time | Time taken to search for the record. Nesting may occur or other logic may be included. |
| | SQL fetch count | Number of the records retrieved from SQL. |
| Resource | CPU usage | CPU time used for processing transactions. |

| Classification | Properties | Description |
|----------------|--------------------|---|
| | time | |
| | Memory allocation | Amount of memory used by the transaction. |
| Containers | Pod name | It is the name of the Kubernetes Pod that processed the transaction. |
| | Container Id | It is a unique identifier for a specific container running within a Pod. |
| Client | Client IP | Client IP address. |
| | OS | OS environment in which the browser runs. |
| | Domain | Internet address assigned to the IP address that the client connected to. |
| | Country/
Region | Client's country. |
| | City | City in the client's country. |
| | Client type | Browser type used by the client. |
| | Client name | Device name used by the client. |
| | WClientID | This is a unique ID of the client. |
| Method | methodCount | Number of methods called by transactions. |
| | methodTime | Time taken for method execution. |

- The collected data may differ depending on the application type, settings, and step type. **Container** entries appear only when the application is running in the container environment.
- Error related items are displayed when an error occurs from the transaction.
- Some transaction properties provide statistical features. If you select the  button, a window

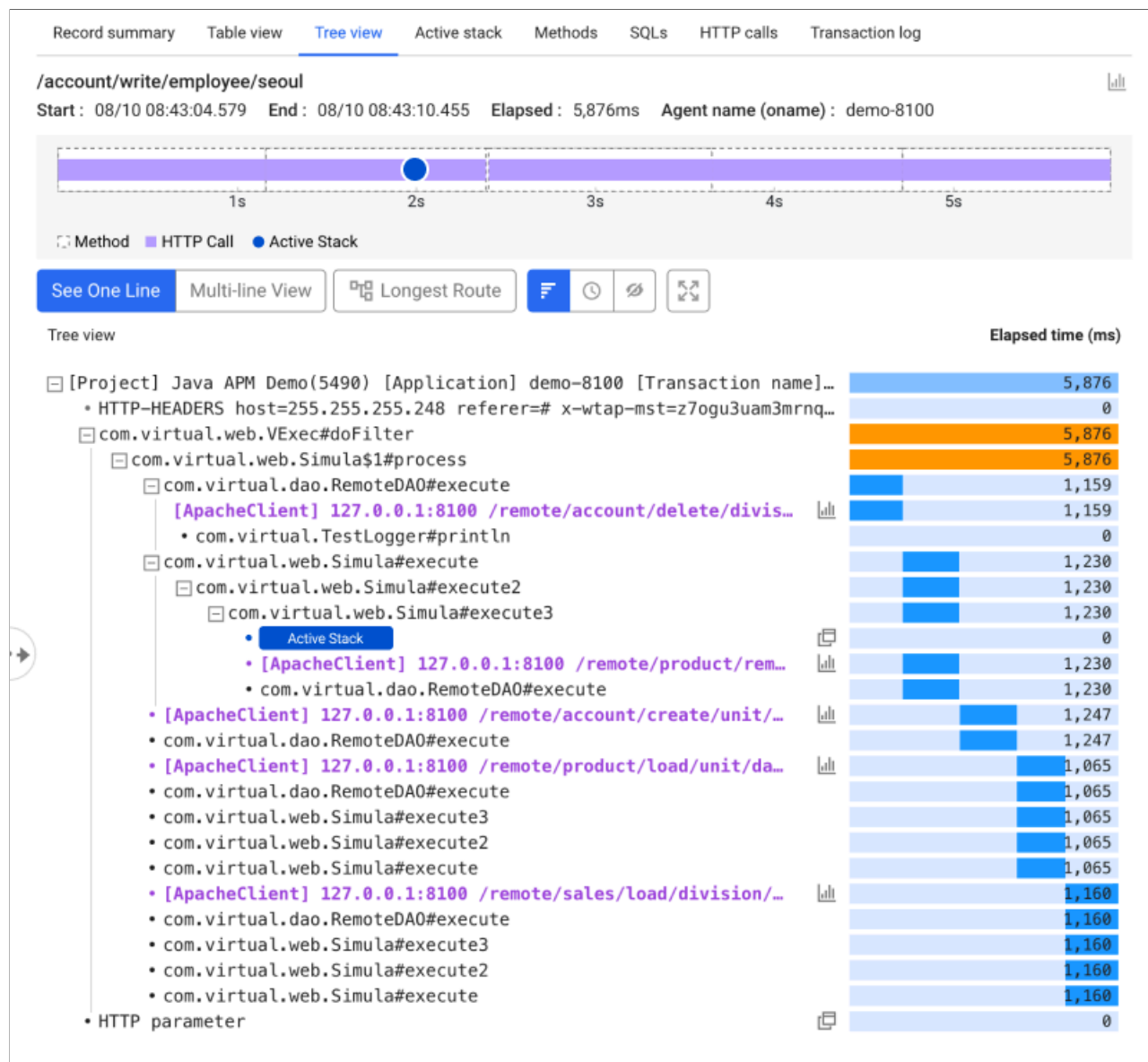


appears where you can see the status of transactions occurred during the query time through a time series chart. If you select a bar graph in the chart, you can go to the **APP. > Analysis > Transaction Search** menu to view details.

- WhaTap basically stores the client-related information. For more information about the agent settings for user data collection, see [the following](#).

Tree view

To see the transaction execution steps in a tree format, select the **Tree view** tab. You can see the details of each transaction and its trace details, trace start time, step time, and call relationship. If you select each section of the diagram, the tree view of the step appears.



- **Single-line View:** You can arrange the tree format at intervals by displaying the texts displayed in the execution data for each section in a single line.
- **Multi-line View:** All the texts displayed in the execution data for each section, appear by wrapping words.
- **Longest Path:** The longest route can appear.
- **Show Time Bar:** The elapsed times are displayed in a bar chart.
- **Show Time:** The timestamps, gaps, and elapsed times for each segment are displayed in text.

- 8 seconds or more: **Excessive delay** state is distinguished in **red**.
- Between 3 and 8 seconds: **Delay** state is distinguished in **orange**.
- Less than 3 seconds: **Normal** state is distinguished in **blue**.
-  **Hide Time**: The time is hidden.
- : You can see only the tree view that hides transaction summary and diagrams. If you select , the hidden information reappears. This feature is useful when the tree view list is long.
- : A window appears where you can view SQL variables and HTTP queries. For more information, see [the following](#).

ⓘ In case of start and step times, the time differences caused by the transaction call environment are corrected on the screen through traces and mappings in the parent transactions. Accordingly, there may be differences from the actual time data.

Active Stack

Record summary
Table view
Tree view
Active stack
Methods
SQLs
HTTP calls
Transaction log

/account/load/division/daegu


Stack


Active Stack analysis
Beta

Start : 04/06 14:16:32.419
End : 04/06 14:17:33.709
Elapsed : 61,290ms
Agent name (oname) : demo-8100
Client IP : 167.109.194.221

Active Stack (6)

| No | Time | |
|-----|--------------|--|
| 30 | 14:16:37.807 | java.base@17.0.12/sun.nio.ch.SocketDispatcher.read0(Native Method) |
| 70 | 14:16:48.223 | java.base@17.0.12/sun.nio.ch.SocketDispatcher.read(SocketDispatcher.java:47) |
| 80 | 14:16:58.504 | java.base@17.0.12/sun.nio.ch.NioSocketImpl.tryRead(NioSocketImpl.java:266) |
| 81 | 14:17:08.542 | java.base@17.0.12/sun.nio.ch.NioSocketImpl.implRead(NioSocketImpl.java:317) |
| 82 | 14:17:18.547 | java.base@17.0.12/sun.nio.ch.NioSocketImpl.read(NioSocketImpl.java:355) |
| 108 | 14:17:28.575 | java.base@17.0.12/sun.nio.ch.NioSocketImpl\$1.read(NioSocketImpl.java:808) |

```

java.base@17.0.12/java.net.Socket$SocketInputStream.read(Socket.java:966)
app/org.apache.http.impl.io.SessionInputBufferImpl.streamRead(SessionInputBu
fferImpl.java:139)
app/org.apache.http.impl.io.SessionInputBufferImpl.fillBuffer(SessionInputBu
fferImpl.java:155)
app/org.apache.http.impl.io.SessionInputBufferImpl.readLine(SessionInputBuff
erImpl.java:284)

```

Select the **Active stack** tab to view the list of active stacks collected from the transaction.

- Select a collection point (No, Time) from the **Active Stack** list on the left to display the stack information for that point on the right.
-  **Stack**: You can download the active stack details as a TXT file.

- You can check the active stack from items marked with the  icon in the transaction list.
- For more information about active stacks, see [the following](#).

AI Active Stack Analysis

Beta

Click the **Active Stack Analysis** button to automatically analyze collected active stacks with AI. It identifies performance issues based on the stack dumps of in-progress transactions and suggests improvements.

The analysis results consist of the following 4 sections:

- **Summary**: Summarizes the number of analyzed stacks and the overall execution flow.
- **Common Patterns**: Displays repeatedly observed execution patterns, including pattern name, occurrence count, and description.
- **Issues Found**: Displays identified performance issues with severity levels (**High**, **Medium**, **Low**) along with the stack information used for the assessment.

| Severity | Criteria |
|----------|--|
| High | Deadlock, complete blocking, same blocking repeated 4+ times |
| Medium | I/O wait, 2-3 lock contentions, slow external calls |
| Low | Single-occurrence wait, minor optimization opportunities |

- **Recommendations**: Suggests specific improvement actions by priority.

Each analysis item displays the collected stack number (**No.N**). Click the stack number to view the original stack.

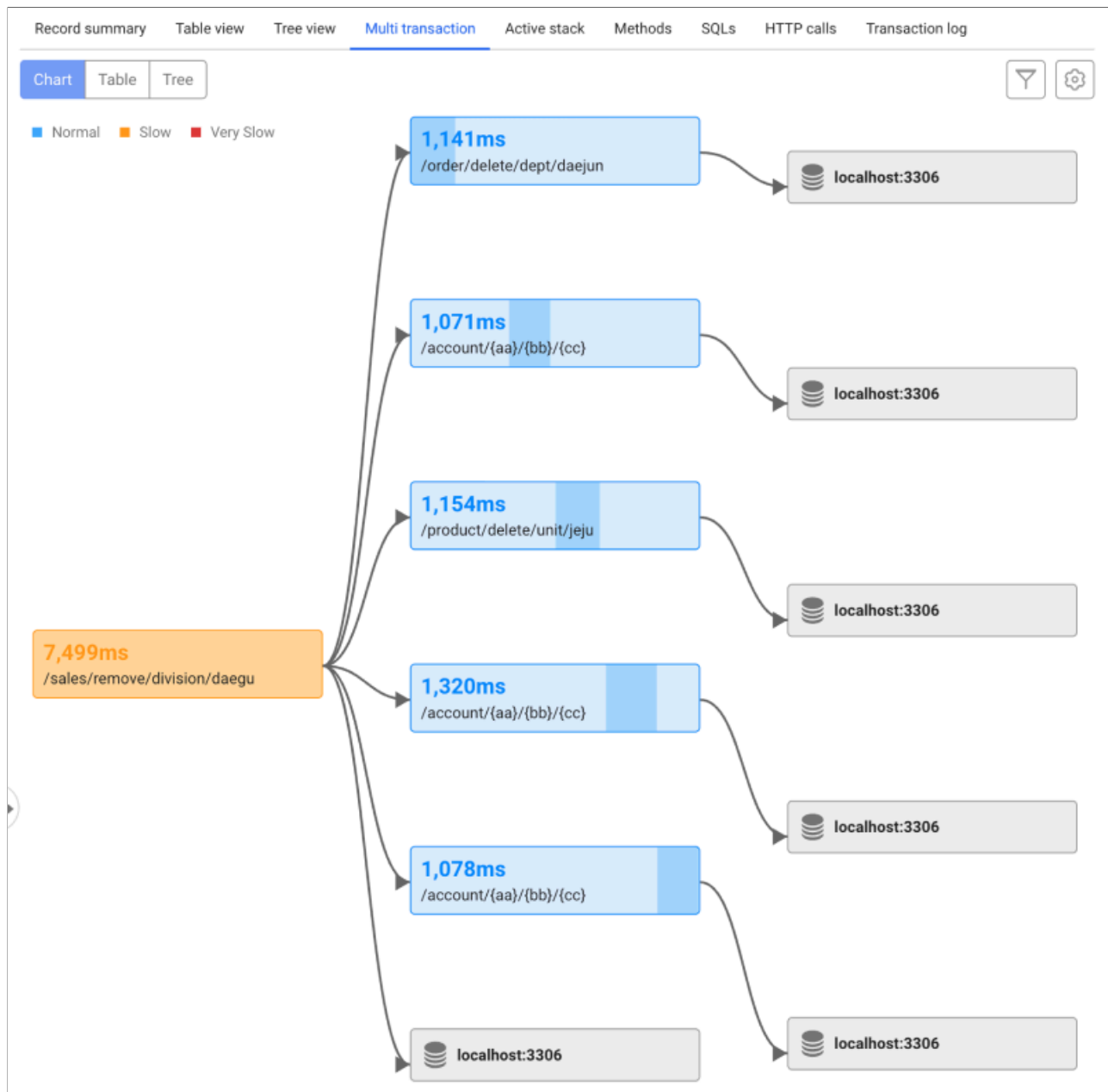
- The analysis results are generated by AI and may differ from actual conditions. Use them as

- ! reference material.
- **Member** role or above is required.

Multi-transaction

Multi-transaction indicates the transaction involving other agents or projects. In the **Multi transaction** tab, you can see the call relationships between applications registered in the WhaTap monitoring service.

- ✓ To trace multi-transactions, set the `mtrace_enabled` option to `true` in **Management > Agent CONF..** For more information about the agent configuration, see [the following](#).



- **Chart:** Provides the call relationships for each transaction in a flowchart. Select a transaction node to see the trace analysis for the transaction node. In **Chart** mode, you can use the mouse to go to a desired position or zoom in or out by scrolling.
 - **Chart view settings:** You can display or hide the elements on the chart.
- **Table:** In the form of table, you can check the data for each transaction included in the multi-transaction. You

can select the  column selection icon to edit the table header columns. You can see the trace analysis data by selecting each transaction item.

- **Tree:** You can see the call relationships between transactions in a tree format. The related additional functions are the same as those of the **Tree view** tab.
-  **Project selection:** You can select or deselect projects to be displayed on the chart.

- You can see them in the transaction list marked with the  icon.
- For more information about the multi-transaction, see [the following](#).

Methods

| Record summary Table view Tree view Active stack Methods SQLs HTTP calls Transaction log | | | | | | | |
|---|----------------------------|----------|---------------------------|-------|----------|----------|--|
| /account/read/unit/jeju | | | | | | |   Method |
| No | Class | Method | Parameter/Return | Count | Sum (ms) | Max (ms) | |
| 1 | com.virtual.dao.Remot... | execute | ()V | 27 | 39,063 | 2,776 | |
| 2 | com.virtual.dao.PITest | exec | (Ljava/sql/PreparedSta... | 3 | 0 | 0 | |
| 3 | com.virtual.dao.InsertD... | execute2 | (Ljava/lang/String;)V | 2 | 4 | 3 | |
| 4 | com.virtual.dao.Update... | execute2 | (Ljava/lang/String;)V | 2 | 18 | 14 | |
| 5 | com.virtual.dao.Delete... | execute | ()V | 2 | 3 | 2 | |
| 6 | com.virtual.dao.Select... | execute2 | (Ljava/lang/String;)V | 1 | 5 | 5 | |
| 7 | com.virtual.dao.Update... | execute | ()V | 1 | 4 | 4 | |
| 8 | com.virtual.dao.Select... | execute2 | ()V | 1 | 7 | 7 | |
| 9 | com.virtual.dao.Select... | execute | ()V | 1 | 9 | 9 | |
| 10 | com.virtual.dao.InsertD... | execute2 | ()V | 1 | 3 | 3 | |

To see only the method data, select the **Methods** tab. It displays the method name and step time for which tracing has been set on the agent. When unnecessary logics are repeatedly executed, it can be used as analysis data to improve the method logic by finding the case where the execution takes longer than the low count.

- : You can see the **Transaction statistics** information. If a graph on the chart is selected, the **Analysis > Transaction Search** menu appears. You can see the list of transactions for the selected time zone.

-  **Method:** You can download the method-related details as a TXT file.

 For more information about the agent configuration options on methods, see [the following](#). Method tracing support may differ depending on the product in use.

SQLs

| Record summary | Table view | Tree view | Active stack | Methods | SQLs | HTTP calls | Transaction log |
|-------------------------|----------------------------|-----------------------------|--------------|----------|----------|---|---|
| /account/read/unit/jeju | | | | | |  |  SQL |
| No | Database | SQL | Count | Sum (ms) | Max (ms) | | |
| 1 | jdbc:mysql://localhost:... | DELETE FROM Custom... | 2 | 0 | 0 | | |
| 2 | jdbc:mysql://localhost:... | SELECT distance, lat, lo... | 1 | 0 | 0 | | |
| 3 | jdbc:mysql://localhost:... | SELECT distance, lat, lo... | 1 | 0 | 0 | | |
| 4 | jdbc:mysql://localhost:... | UPDATE emp set e.ena... | 1 | 0 | 0 | | |
| 5 | jdbc:mysql://localhost:... | update table set x=# w... | 1 | 0 | 0 | | |
| 6 | jdbc:mysql://localhost:... | SELECT p.Name AS Pr... | 1 | 0 | 0 | | |
| 7 | jdbc:mysql://localhost:... | SELECT customer_info... | 1 | 0 | 0 | | |
| 8 | jdbc:mysql://localhost:... | INSERT INTO EMP VAL... | 1 | 1 | 1 | | |

To check the SQL statement information, select the **SQLs** tab. In case of unnecessarily repetitive execution, it can be used as analytical data to improve query performance by finding the case where the execution time takes longer compared to the low count.

- : You can see the **Transaction statistics** information. If a graph on the chart is selected, the **Analysis > Transaction Search** menu appears. You can see the list of transactions for the selected time zone.
-  **SQL:** You can download the SQL-related details as a TXT file.

HTTP calls

| | | | | | | | |
|----------------|------------|-----------|--------------|---------|------|------------|-----------------|
| Record summary | Table view | Tree view | Active stack | Methods | SQLs | HTTP calls | Transaction log |
|----------------|------------|-----------|--------------|---------|------|------------|-----------------|

| /account/read/unit/jeju | | | | | | | HTTP |
|-------------------------|----------------|----------------------------|-------|----------|----------|--|------|
| No | Remote server | URL | Count | Sum (ms) | Max (ms) | | |
| 1 | 127.0.0.1:8105 | /remote/account/save... | 2 | 4,116 | 2,776 | | |
| 2 | 127.0.0.1:8105 | /remote/order/save/di... | 1 | 238 | 238 | | |
| 3 | 127.0.0.1:8105 | /remote/sales/pickup/... | 1 | 922 | 922 | | |
| 4 | 127.0.0.1:8105 | /remote/sales/remove... | 1 | 1,980 | 1,980 | | |
| 5 | 127.0.0.1:8105 | /remote/order/save/de... | 1 | 2,639 | 2,639 | | |
| 6 | 127.0.0.1:8105 | /remote/edu/kill/unit/p... | 1 | 1,519 | 1,519 | | |
| 7 | 127.0.0.1:8105 | /remote/order/read/de... | 1 | 1,467 | 1,467 | | |
| 8 | 127.0.0.1:8105 | /remote/order/delete/... | 1 | 485 | 485 | | |
| 9 | 127.0.0.1:8105 | /remote/sales/kill/emp... | 1 | 2,210 | 2,210 | | |

To see the count, time sum, average time, and such for HTTP calls, select the **HTTP calls** tab. If unnecessary external calls are repeatedly executed, it can be used as analytical information to determine whether the transaction delays are external or internal by finding the cases where the execution time takes longer compared to the low count.

- : You can see the **Transaction statistics** information. If a graph on the chart is selected, the **Analysis > Transaction Search** menu appears. You can see the list of transactions for the selected time zone.
- **HTTP**: You can download the HTTP call-related details as a TXT file.

Transaction log

To see the logs of the transaction, select the **Transaction log** tab.

| Transaction information | | | |
|-------------------------|------------|------------|-------------------|
| Record summary | Table view | Tree view | Multi transaction |
| Methods | SQLs | HTTP calls | Transaction log |

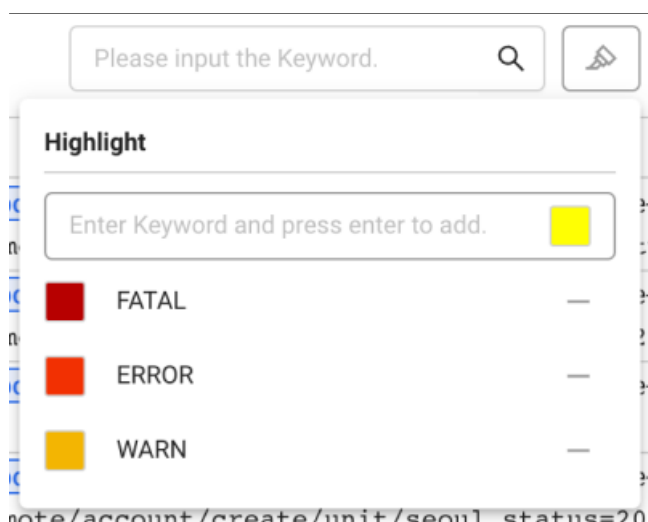
| /sales/save/division/daejun | | | |
|---|------------|-------------------------|--|
| Log search > <input type="text" value="Please input the Keyword."/>     | | | |
| ▶ | oname | Timestamp | Log |
| ▶ | java-demo2 | 2025-04-16 09:17:12.571 | @txid 2737043795003671073 pcode 8 onodeName node-1 oid -1737750367 okind 867318026 oname java-demo2 agenttime 1744762632114
select distinct pp.lastname, pp.firstname |
| ▶ | java-demo2 | 2025-04-16 09:17:12.773 | @txid 2737043795003671073 pcode 8 onodeName node-1 oid -1737750367 okind 867318026 oname java-demo2 agenttime 1744762632155
select |
| ▶ | java-demo2 | 2025-04-16 09:17:12.773 | @txid 2737043795003671073 pcode 8 onodeName node-1 oid -1737750367 okind 867318026 oname java-demo2 agenttime 1744762632193
select productid, avg(orderqty) as averagequantity, sum(linetotal) as total |

- : You can see the **Transaction statistics** information. If a graph on the chart is selected, the **Analysis > Transaction Search** menu appears. You can see the list of transactions for the selected time zone.
- **Log Search**: You can go to Log Search to easily find the desired log data.
- **Keyword search**: Enter the desired text to search in the keyword search field, and then press Enter or select the  button. It highlights the keywords that match the entered text.
-  **Keyword highlight**: If you set the keyword and color, the texts that match the keyword are automatically highlighted in the log list. For more information, see [the following](#).
-  **Column settings**: You can add columns or change the order of columns displayed in the log list table. For more information, see [the following](#).
-  **Log view detail setting**: You can configure the items to be displayed in the log list. For more information, see [the following](#).

-  To enable the **Transaction log** tab, apply the agent setting options related to the logs. For more information about the log settings, see [the following](#).
- The **Transaction log** tab can be accessed by only the members with the **View log** role. For more information about the member roles, see [the following](#).

Setting the keyword highlight

1. To add keywords to highlight, select .



2. Enter a desired keyword.
3. Select the color icon and then select a desired color.
4. Press Enter. The added keywords are added in the list.

Deleting the keyword highlight

To delete the items unused from the highlight list, select —.

Displaying or hiding the table columns

1. Select  on the upper right of the table.
2. If the **Column settings** window appears, select or unselect the desired item in the list on the left.
You can also hide columns by selecting the  button in the list on the right.
3. After all settings are finished, select **Confirm**. The set content is applied to the table.

Reordering the table columns

1. Select  on the upper right of the table.
2. When the **Column settings** window appears, change the order by dragging the desired items from the list on the right.

3. After all settings are finished, select **Confirm**. The set content is applied to the table.

Setting to display the table content

1. Select  on the upper right of the table.
2. When the **Log view detail setting** window appears, set the desired options.
 - Select **content** or **Tag** to determine whether or not to display the log content.
 - **Tag management**: Tags are displayed in the log in the order of addition. The order of tags can be changed by dragging. Disabled tags do not appear in the logs.
3. After all settings are finished, click . The set content is applied to the table.

 For more information about **Log view detail setting**, see [the following](#).

Additional options

Viewing HTTP parameters

You can view the HTTP parameters of the transaction in the **Table view** tab.

1. Scroll down the page to the final step of the transaction.
2. Select the  **HTTP parameters** item.
3. If the **HTTP-PARAMETERS** window appears, select **Password**.
4. Enter the specified Param Key. You can see the hidden parameters.

-  • For more information about the agent configuration with HTTP parameters, see [the following](#).
- **Password**: You can see the decrypted parameter value. The password is a 6-digit string in the `WHATAP_HOME /security.conf (paramkey.txt)` file. You can change it to another string. It is required to view the SQL variables and HTTP queries and to stop the threads.
 - Depending on the agent version, the password configuration file may be different. Change the

! string in the [security.conf](#) file or [paramkey.txt](#) file.

Viewing SQL parameters

Select the SQL step in the **Table view** tab or select  in the **Tree view** tab. The **SQL** window appears to display parameters.

SQL

Connection URL

jdbc:mysql://localhost:3306,localhost:3310/fake@BasicDataSource

CPU accumulation

3ms

Memory accumulation

329,752byte

Query (Hash: -2057375474)

🔒 Password

SQL format

📄

Statistics >

SID

107

```

insert into
dept
values(
    #, ' #', ' #' )

```

📄

Substitution variables

The parameter does not exist or decryption has failed.

Please check your password at [paramkey.txt] in the agent installation path to decrypt parameters.

- **Password:** You can see the decrypted parameter value. The password is a 6-digit string in the `WHATAP_HOME /paramkey.txt` file. You can change it to another string.

! The key in [paramkey.txt](#) is required to view the SQL variables and HTTP queries, and stop the threads.

- **SQL format:** You can indent and format your SQL statements for better readability.
- : A SQL statement can be copied to the clipboard.
- **Statistics:** Go to **Statistics > SQL**.

✓ To search SQL variables and HTTP queries, add the following options to the agent settings. For more information about the agent settings for DB and SQL, see [the following](#).

624



whatap.conf

SQL parameter lookup option: If the option is applied, SQL parameters are collected after being encrypted.`profile_sql_param_enabled=true`*// HTTP parameter lookup option: If the option is applied, HTTP query parameters are collected after being encrypted.*`profile_http_parameter_enabled=true`

Configuring the table columns

In the transaction list, you can hide the table header columns or add any of them. You can also change the column order. Select  **Column Settings**.

Column Settings

Please select a column to display

3 Search by column name

1

2

Display column.

No

|| Agent name (oname) X

|| Transaction X

|| Elapsed time X

|| Start time YY/MM/DD 00:00:00 X

|| End time YY/MM/DD 00:00:00 X

|| HTTP call count X

|| HTTP call time X

|| SQL time X

|| SQL count X

|| DB CONN. time X

|| SQL patches count X

|| Error message X

|| Client IP X

Cancel Confirm

- After configuration, select **Confirm** to apply the settings in the table.
- In the 3 search bar, enter text to search the desired columns. Only the columns that meet the entered text are displayed.

Adding columns

From the 1 list, select the items to add as table header columns. To select all items, select **Select All**.

Deleting columns

From the **1** list, unselect the columns to delete. Alternatively, select **×** on the right of the item to delete from the **2** list.

Changing the column order

Drag an item to reposition from the **2** list, and then move it to the desired position.

Initializing the configuration

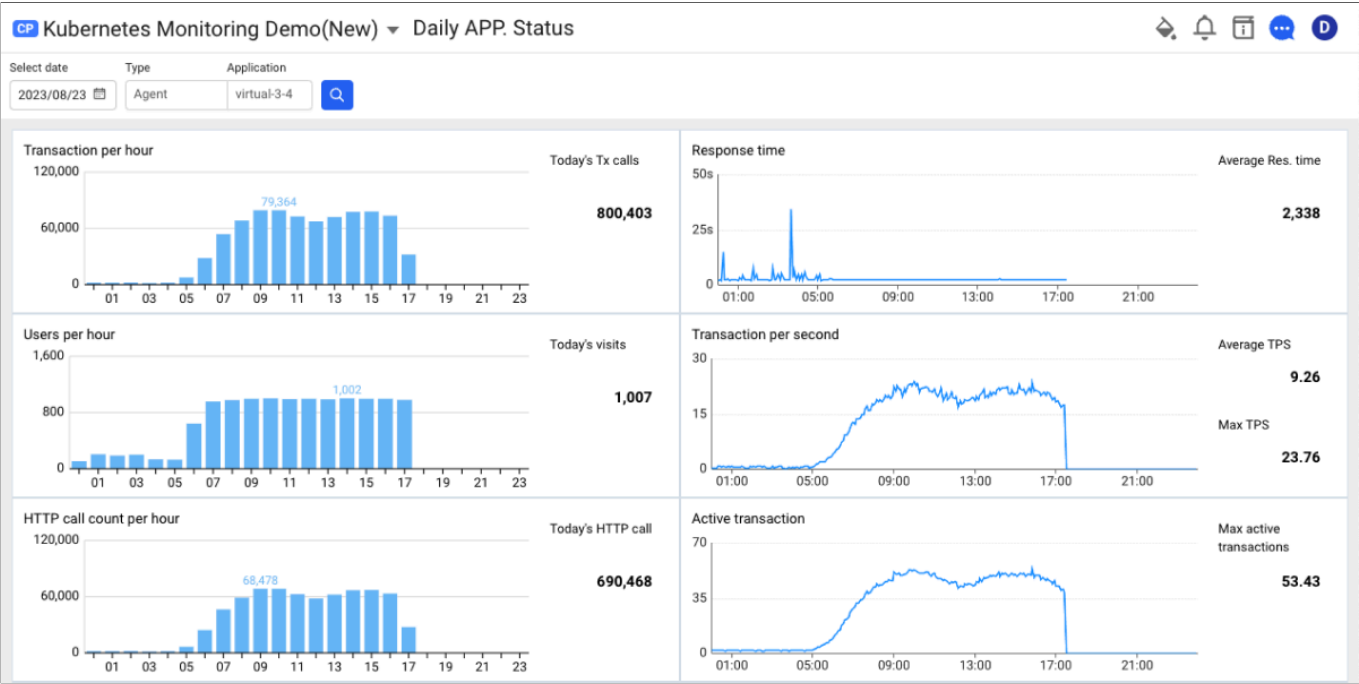
To cancel all changes and reset them, select **↺ Initialized**.

Downloading the viewed data

To download the data output in the table in .txt file format, select **↓ TXT**.

Daily Application Status

Home > Select Project > [Application](#) > [Analysis](#) > [Daily APP. Status](#)



- Users per hour
- HTTP Call Count per hour
- SQL Count per hour
- SQL Fetch Count per hour
- Response Time
- Transaction Per Second
- Active Transaction
- Errors in 5mins
- Error Rate

Transaction Search

Home > Select Project > [Application](#) > [Analysis](#) > [Transaction Search](#)

In [Transaction Search](#), you can view the statistics on the performance and execution status for each transaction for a specific period of time. You can view all transactions and their errors. The provided information allows you to see which transactions are busy and resource-consuming. The transactions are filtered in various ways depending on the transaction properties, making it useful for finding transactions with specific conditions.

CP Kubernetes Monitoring Demo(EKS) Transaction Search

Time selector: < 2024/03/07 00:00 ~ 2024/03/07 23:59 1Day > Container: All selected

Filter: + Elapsed time + Error message + HTTP call count + SQL count + SQL fetch count + Transaction + Client IP + Domain + User agent + Referer + Transaction ID + Multi transaction ID + WClientID + Login ID ☐ Errors only

Search icon CSV

| Agent name (oname) | Start time | End time | Elapsed time | Transaction | CPU usage... | HTTP call ... | SQL count | SQL fetch c... |
|---|------------|----------|--------------|-------------|--------------|---------------|-----------|----------------|
| <p>Please use filter for detailed search</p> <p>It supports data search by multiple search conditions.
If the search range is too large, there is a limit to data extraction.</p> | | | | | | | | |

Set the search time in the 1 area, and then select search targets from the [Container](#) options. If you select , the search results appear in the table. If you select desired items from the result list in the 2 area, the [Transaction Info](#) window appears. For the detailed information on the transaction trace analysis using the [Transaction Info](#) window, see [the following](#).

- To search for transactions in error, click **Errors only** and then select .
- It is recommended to set the search time as short as possible because a response delay may occur depending on the search conditions.

Select Column

If you click  [Select column](#) in the  area, the column selection window appears as follows. Select desired columns.

Select all
Deselect all
Initialize(default)
×

| | | | | | | |
|--|---|--|---|--|---|--|
| ☐ Agent | ☐ Client | ☐ Error | ☐ HTTP call | ☐ Resource | ☐ SQL | ☐ Transaction |
| ☐ Container Id | ☐ City | ☐ Error class | ☒ HTTP call count | ☒ CPU usage time | ☐ DB CONN. time | ☐ Active stack status |
| ☐ Agent ID (oid) | ☐ Operating system | ☐ Error level | ☐ HTTP call time | ☐ Memory allocation | ☒ SQL fetch count | ☐ Custom TID |
| ☐ Agent group ID | ☐ Country/Region | ☐ Error message | | | ☐ SQL patch time | ☐ Domain |
| ☐ Agent group name | ☐ Client IP | | | | ☐ Transaction hash | ☒ Elapsed time |
| ☒ Agent name (oname) | ☐ Login ID | | | | ☒ SQL count | ☒ End time |
| ☐ Agent server ID | ☐ User agent | | | | ☐ SQL time | ☐ CallerTxId |
| ☐ Agent server name | ☐ UserAgentHash | | | | | ☐ Call depth |
| ☐ Project code | ☐ User type | | | | | ☐ HTTP method |
| ☐ Pod name | ☐ WClientID | | | | | ☐ Multi transaction ID |
| ☐ Sub type | | | | | | ☐ Original URL |
| | | | | | | ☐ Referer |
| | | | | | | ☒ Transaction |
| | | | | | | ☒ Start time |
| | | | | | | ☐ Transaction ID |

Apply Filter

You can perform detailed search using filters in the [Leader line 1](#) area. It supports data exploration using various search conditions. Multiple filters can also be applied. Specify desired filters and then select .

- ⓘ • Some filters may not be supported depending on the product.
- When applying multiple filters, to search by the **Multi transaction ID** filter, only the **Transaction ID** filter can be added.

Elapsed Time

CP

Kubernetes Monitoring Demo(EKS) ▾ Transaction Search

🔊

🔔

📅

⋮

Y

Time selector

< 2024/03/07 00:00 ~ 2024/03/07 23:59 1Day >

Container

All selected ▾

Filter

— Elapsed time

1000

5000

+ Error message

+ HTTP call count

+ SQL count

+ SQL fetch count

+ Transaction

+ Client IP

+ Domain

+ User agent

+ Referer

+ Transaction ID

+ Multi transaction ID

+ WClientID

+ Login ID

☐ Errors only

🔍

📄 CSV

Select Column

| | Agent name (oname) | Start time | End time | ↓ Elapsed time | Transaction | CPU usage... | HTTP call ... | SQL count | SQL fetch c... |
|----|--------------------|-------------------------|-------------------------|----------------|----------------------------------|--------------|---------------|-----------|----------------|
| 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 |

Previous

Next

Search is performed based on the transaction response time. Search by entering two values above and below the reference.

Error Message

CP

Kubernetes Monitoring Demo(EKS) ▾

Transaction Search

Time selector

<

2024/03/07 00:00 ~ 2024/03/07 23:59

1Day

>

Container

All selected ▾

Filter

+ Elapsed time

— Error message

SLOW

+ HTTP call count

+ SQL count

+ SQL fetch count

+ Transaction

+ Client IP

+ Domain

+ User agent

+ Referer

+ Transaction ID

+ Multi transaction ID

+ WClientID

+ Login ID

☐ Errors only

Q

CSV

Select Column

| name (oname) | Start time | End time | Elapsed time | Transaction | CPU usage... | HTTP call ... | SQL count | SQL fetch ... | Error message |
|--------------|-------------------------|-------------------------|--------------|--------------------------------|--------------|---------------|-----------|---------------|---------------|
| | 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 16:03:53.980 | 148,965 | /sales/delete/division/pusan | 4 | 6 | 7 | 1,800 | SLOW_SQL |

Previous

Next

Search is performed based on the transaction error message. In the default state, error-related columns may not be seen. Select a column to check.

ⓘ Transactions in error are displayed in red.

HTTP Call Count

CP Kubernetes Monitoring Demo(EKS) Transaction Search

Time selector: < 2024/03/07 00:00 ~ 2024/03/07 23:59 1Day > Container: All selected

Filter: + Elapsed time + Error message - HTTP call count 5 + SQL count + SQL fetch count + Transaction + Client IP + Domain + User agent + Referer + Transaction ID

+ Multi transaction ID + WClientID + Login ID ☐ Errors only

Q CSV

Select Column

| | Agent name (oname) | Start time | End time | Elapsed time | Transaction | CPU usage time | HTTP call count | SQL count | SQL fetch c... |
|----|--------------------|-------------------------|-------------------------|--------------|--------------------------------------|----------------|-----------------|-----------|----------------|
| 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/kwa
ngju | 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/kwa
ngju | 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 |

Previous Next

It searches the transactions in which the number of calls for external HTTP services is greater than or equal to the input value. For example images, 5 or more cases are viewed.

SQL Count

CP Kubernetes Monitoring Demo(EKS) Transaction Search

Time selector: < 2024/03/07 00:00 ~ 2024/03/07 23:59 1Day > Container: All selected

Filter: + Elapsed time + Error message + HTTP call count - SQL count 1200 + SQL fetch count + Transaction + Client IP + Domain + User agent + Referer + Transaction ID

+ Multi transaction ID + WClientID + Login ID ☐ Errors only

Q CSV

Select Column

| | Agent name (oname) | Start time | End time | Elapsed time | Transaction | CPU usage... | HTTP call ... | SQL count | SQL fetch c... |
|---|--------------------|-------------------------|-------------------------|--------------|-----------------------------|--------------|---------------|-----------|----------------|
| 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 |

It searches the transactions in which the number of SQL executions per transaction is greater than or equal to the set input value. It is used to search for transactions that exceptionally call an excessive number of SQLs.

SQL Fetch Count

CP Kubernetes Monitoring Demo(EKS) Transaction Search

Time selector: < 2024/03/11 00:00 ~ 2024/03/11 23:59 1Day > Container: All selected

Filter: + Elapsed time + Error message + HTTP call count + SQL count + SQL fetch count 500000 + Transaction + Client IP + Domain + User agent + Referer + Transaction ID

+ Multi transaction ID + WClientID + Login ID ☐ Errors only

Q CSV

| | Agent name (oname) | Start time | End time | Elapsed time | Transaction | CPU usage... | HTTP call ... | SQL count | SQL fetch c... |
|---|--------------------|-------------------------|-------------------------|--------------|-------------------------------|--------------|---------------|-----------|----------------|
| 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 |

After execution of SQL, the transactions whose count is the input value or more based on the number of DB patches. It is used to search transactions that perform excessive data patching.

Transactions

CP Kubernetes Monitoring Demo(EKS) Transaction Search

Time selector: < 2024/03/11 00:00 ~ 2024/03/11 23:59 1Day > Container: All selected

Filter: + Elapsed time + Error message + HTTP call count + SQL count + SQL fetch count + Transaction product + Client IP + Domain + User agent + Referer + Transaction ID

+ Multi transaction ID + WClientID + Login ID ☐ Errors only

Q CSV

| | Agent name (oname) | Start time | End time | Elapsed time | Transaction | CPU usage... | HTTP call ... | SQL count | SQL fetch c... |
|---|--------------------|-------------------------|-------------------------|--------------|---------------------------------|--------------|---------------|-----------|----------------|
| 1 | virtual6 | 2024/03/11 00:00:00.261 | 2024/03/11 00:00:01.465 | 1,204 | /product/create/division/daegu | 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 | 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 | 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 | 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 | 2 | 0 | 11 | 5,343 |

Search is performed based on the transaction. The entered search string is compared for a search based on whether it is included or not (partial match).

Client IP

CP Kubernetes Monitoring Demo(EKS) Transaction Search

Time selector: < 2024/03/11 00:00 ~ 2024/03/11 23:59 1Day > Container: All selected

Filter: + Elapsed time + Error message + HTTP call count + SQL count + SQL fetch count + Transaction + Client IP 87.132 + Domain + User agent + Referer + Transaction ID + Multi transaction ID + WClientID + Login ID ☐ Errors only

Q CSV

Select Column

| | Agent name (oname) | Start time | End time | Elapsed time | Transaction | Client IP | CPU usage... | HTTP call ... | SQL c |
|---|--------------------|-------------------------|-------------------------|--------------|-------------------------------|----------------|--------------|---------------|-------|
| 1 | virtual5 | 2024/03/11 00:00:00.849 | 2024/03/11 00:00:01.959 | 1,110 | /account/save/employee/seoul | 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 | 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/daejun | 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 | 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 | 87.132.232.135 | 1 | 0 | |

The transactions are searched based on the client IP. However, IP strings are searched using PREFIX comparison (forward match).

Domain

CP Kubernetes Monitoring Demo(EKS) Transaction Search

Time selector: < 2024/03/11 00:00 ~ 2024/03/11 23:59 1Day > Container: All selected

Filter: + Elapsed time + Error message + HTTP call count + SQL count + SQL fetch count + Transaction + Client IP + Domain 127.0.0.1:8181 + User agent + Referer + Transaction ID + Multi transaction ID + WClientID + Login ID ☐ Errors only

Q CSV

Select Column

| | Agent name (oname) | Start time | End time | Elapsed time | Transaction | Domain | CPU usage... | HTTP call ... | SQL count | 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 | |

Transactions are retrieved based on the HTTP service domain (host:port). The search conditions must match the domain completely (full match).

UserAgent

CP Kubernetes Monitoring Demo(EKS) Transaction Search

Time selector

2024/03/11 00:00 ~ 2024/03/11 23:59 1Day

Container

All selected

Filter

+ Elapsed time

+ Error message

+ HTTP call count

+ SQL count

+ SQL fetch count

+ Transaction

+ Client IP

+ Domain

- User agent

Chrome

+ Referer

+ Transaction ID

+ Multi transaction ID

+ WClientID

+ Login ID

☐ Errors only

Q

CSV

Select Column

| name (oname) | Start time | End time | Elapsed time | Transaction | CPU usage... | HTTP call ... | SQL count | SQL fetch ... | User agent | |
|--------------|-------------------------|-------------------------|--------------|--------------------------------|--------------|---------------|-----------|---------------|------------|---|
| | 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 6.1; en-US) AppleWebKit/532.0 (KHTML, like Gecko) Chrome/3.0.198.0 Safari/532.0 |

Search transactions based on the user agent (partial match).

Referer

Search transactions based on the referer string.

!

Agents in their default settings do not collect referer data. If the referer configuration is required, see the following to add the related settings.

• Java

• Node.js

• Python

Transaction ID

CP Kubernetes Monitoring Demo(EKS) Transaction Search

Time selector: < 2024/03/14 00:00 ~ 2024/03/14 23:59 1Day > Container: All selected

Filter: + Elapsed time + Error message + HTTP call count + SQL count + SQL fetch count + Transaction + Client IP + Domain + User agent + Referer **- Transaction ID: -4884510994358729563**

+ Multi transaction ID + WClientID + Login ID ☐ Errors only

Q CSV

Select Column

| Agent name (oname) | Start time | End time | Elapsed time | Transaction | CPU usage... | HTTP call ... | SQL count | SQL fetch ... | Transaction 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 | -4884510994358729563 |

Search transactions based on their transaction ID (full match). For transaction IDs, the search indexes are created for searching at a high speed.

- The transaction ID can be found in the **Record summary** tab of the **Transaction information** window. For more information, see [the following](#).
- If you are not using the latest version of the agent, a full scan may occur rather than using indexes.

Multi Transaction ID

CP Kubernetes Monitoring Demo(EKS) Transaction Search

Time selector: < 2024/03/14 00:00 ~ 2024/03/14 23:59 1Day > Container: All selected

Filter: + Elapsed time + Error message + HTTP call count + SQL count + SQL fetch count + Transaction + Client IP + Domain + User agent + Referer + Transaction ID

- Multi transaction ID: -165248013223001848 + WClientID + Login ID ☐ Errors only

Q CSV

Select Column

| Agent name (oname) | Start time | End time | Elapsed time | Transaction | CPU usage... | HTTP call ... | SQL count | SQL fetch ... | Multi transacti... |
|--------------------|-------------------------|-------------------------|--------------|--------------------------------|--------------|---------------|-----------|---------------|---------------------|
| 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 | -165248013223001848 |
| 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 | -165248013223001848 |
| 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 | -165248013223001848 |
| 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 | -165248013223001848 |
| 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 | -165248013223001848 |
| 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 | -165248013223001848 |
| 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 | -165248013223001848 |

Search transactions based on the multi-transaction ID (full match).

- The multi-transaction ID can be found in the **Record summary** tab of the **Transaction information** window. For more information, see [the following](#).
- If you are not using the latest version of the agent, a full scan may occur rather than using indexes.
- When searching with the **Multi transaction ID** filter applied, only the **Transaction ID** filter can be additionally applied.

WClientID

CP Kubernetes Monitoring Demo(EKS) Transaction Search

Time selector: < 2024/03/14 00:00 ~ 2024/03/14 23:59 1Day > Container: All selected

Filter: + Elapsed time + Error message + HTTP call count + SQL count + SQL fetch count + Transaction + Client IP + Domain + User agent + Referer + Transaction ID + Multi transaction ID

- WClientID + Login ID ☐ Errors only ☐

Q CSV

Select Column

| Agent name (oname) | Start time | End time | Elapsed time | Transaction | CPU usage... | HTTP call ... | SQL count | SQL fetch ... | WClientID |
|--------------------|-------------------------|-------------------------|--------------|--------------------------------|--------------|---------------|-----------|---------------|-----------|
| 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 | |
| irtual1 | 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 | |
| irtual1 | 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 | |
| irtual9 | 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 | |
| irtual12 | 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 | |
| irtual12 | 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 | |
| irtual12 | 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 | |

The client ID (wClientID) is an ID given by the WhaTap agent to identify users. Transactions are searched based on this ID (full match). There is no separate index for searching.

! WhaTap basically stores the client-related information.

Login ID

Search transactions based on the login ID.

! Agents in their default settings do not collect login IDs. For more information about the login ID setting, see [the following](#) to add settings.

Downloading the searched results

The searched results can be downloaded as a CSV file according to the filter conditions set by the user.

1. Click  to view the searched results.
2. Select .
3. Enter the maximum number of CSV lines. You can save the searched results as many lines as you enter.
4. Select **Download** to download the CSV file.

 The CSV file name's format is `profile_ {pcode} _ YYYYMMDD .csv`.

Alert Notifications

This feature delivers notifications to users based on predefined conditions when issues occur in monitored systems or applications. Notifications can be sent through various channels such as email, SMS, messengers, and mobile push notifications.

For example, you can configure alert thresholds for key performance indicators such as CPU usage, memory usage, disk I/O, network traffic, transaction count, error occurrences, and response time latency. When these thresholds are exceeded or abnormal behavior is detected, notifications are automatically sent.

With WhaTap's AI-based alerting, you do not need to define complex conditions manually. The AI learns normal application behavior patterns and automatically detects and notifies anomalies. Additionally, by using tag-count alerts, you can define conditions on values aggregated by specific tags, enabling precise detection even in complex scenarios.

Key Features

- **Event Configuration:** Configure threshold-based alert conditions and notification channels
- **Event Notification Settings:** Configure notification channels and time schedules per user
- **Event History:** View the history of alert occurrences

Permissions

To add or configure events, the **Alert Configuration** permission is required.

- **View:** To view metrics event configurations, the **View & Analyze** permission is required.
- **Add/Edit/Delete:** To modify configurations, the **Alert Configuration** permission is required.
- **Receive Notifications:** To receive notifications, the **Alert Receive** permission is required.

Reference

For more details, see [Member Permissions](#).

Event Configuration

Home screen > Select project  **Alert** > **Event Configuration**

In **Event setting**, you can configure and manage events based on anomaly conditions that may occur in monitored systems or applications. When the configured conditions are met, notifications can be delivered through your preferred channels such as email, SMS, messengers, or app push notifications. You can tailor event configurations according to your monitoring objectives.

Metrics

Metrics events allow you to monitor system and application performance indicators based on real-time data.

Composite Metrics

The composite metrics feature allows you to create events using complex rules and send alert notifications.

Hitmap Patterns

Hitmap patterns automatically detect anomalies based on changes in distributions such as response time and call density, and send alert notifications.

Anomaly Detection

The anomaly detection feature allows you to configure alert notifications by detecting unexpected patterns.

 Real-time logs

Sends alert notifications when event conditions configured in real-time collected logs are detected.

 Composite log

Configure composite log events that trigger alert notifications based on log content.

 Transaction

Transaction events allow you to closely monitor transactions based on specific URLs, errors, status codes, and other conditions.

Home Screen > Select Project >  **Alert** > **Event Configuration**

Event setting provides features to configure and manage events based on anomaly signals that may occur in monitored systems or applications. When configured conditions are met, you can receive notifications through your preferred channels such as email, SMS, messengers, or app push notifications. You can design event configurations according to your monitoring objectives.

 Metrics

Metrics events allow you to monitor system and application performance indicators based on real-time data.

 Composite Metrics

The composite metrics feature allows you to create events using complex rules and send alert notifications.

 Hitmap Patterns

Hitmap patterns automatically detect anomalies based on changes in distributions such as response time and call density, and send alert notifications.

 Anomaly Detection

The anomaly detection feature allows you to configure alert notifications by detecting unexpected patterns.

 Real-time logs

Sends alert notifications when event conditions configured in real-time collected logs are detected.

 Composite log

Configure composite log events that trigger alert notifications based on log content.

 Transaction

Transaction events allow you to closely monitor transactions based on specific URLs, errors, status codes, and other conditions.

 Reference

- When configuring events, use appropriate threshold values to prevent unnecessary event generation.
- When setting up notifications, consider enabling recovery notifications (alerts sent when issues are resolved) to support effective operational response.

Common Features

All event configurations can be downloaded in **JSON or Excel format**. By sharing the same type of product configuration, you can reduce repetitive event setup work.

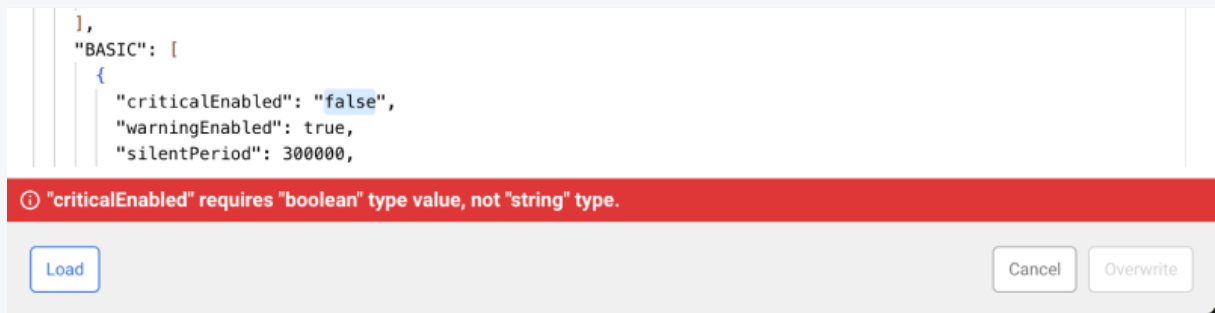
- **Batch JSON Edit:** Edit all event conditions in bulk using JSON format.
- **Batch JSON Download:** Save all currently registered event conditions as a JSON file.
- **Batch Excel Download:** Download the event list as an Excel file for external reporting or backup purposes.

Batch JSON Edit

1. Click the [Batch JSON Edit] button at the top of the screen.
2. Import a JSON file downloaded via the **Batch JSON Download** feature, or edit the content directly.
3. Click the [Overwrite] button to save.
4. When a confirmation message appears, click the [Overwrite] button.

 **After overwriting, you cannot restore the previous event configuration.**

- Arbitrarily modifying the values of the `productType` or `platform` properties may cause events to malfunction in other projects.
- The JSON editor includes built-in validation. If the format or values are invalid, an error message appears below the editor and the [Overwrite] button is disabled.



The screenshot shows a JSON editor with the following content:

```

    },
    "BASIC": [
      {
        "criticalEnabled": "false",
        "warningEnabled": true,
        "silentPeriod": 300000,

```

A red error bar at the bottom of the editor displays the message: **"criticalEnabled" requires "boolean" type value, not "string" type.**

Below the error bar, there are three buttons: **Load** (on the left), **Cancel** (in the middle), and **Overwrite** (on the right).

 **JSON Data Structure**



```
{
  "productType": "{productType}",
  "platform": "{platform}",
  "events": {
    "HITMAP": {
      ...
    },
    "LOG_REALTIME": [
      {...}
    ],
    "METRICS": [
      {...}
    ],
    "COMPOSITE_LOG": [
      {...}
    ],
    "BASIC": [
      {...}
    ],
    "COMPOSITE_METRICS": [
      {...}
    ],
    "ANOMALY": [
      {...}
    ]
  }
}
```

| JSON Field | Description |
|-------------|--|
| productType | Product type (e.g. APM, Infra) |
| platform | Platform information (e.g. java, node) |
| events | Event data |
| ◆ HITMAP | Hitmap pattern |

| JSON Field | Description |
|---------------------|----------------------|
| ❖ LOG_REALTIME | Real-time log events |
| ❖ METRICS | Metrics |
| ❖ COMPOSITE_LOG | Composite log events |
| ❖ BASIC | Basic events |
| ❖ COMPOSITE_METRICS | Composite metrics |
| ❖ ANOMALY | Anomaly detection |

Batch JSON Download

Click the [Batch JSON Download] button at the top of the screen.

- The JSON file name follows the format `integrated-event-YYYYMMDD.json`.
- The structure of the JSON data and the sub-properties under the `events` attribute may vary depending on the product (`platform`) or product type (`productType`).

⚠ If you arbitrarily modify the values of the `productType` or `platform` properties in the JSON data, events may not function correctly in other projects.

Batch Excel Download

Click the [Batch Excel Download] button at the top of the screen.

Metrics

Home > Select Project >  **Alert** > **Event Configuration** > **Metrics** tab

You can view the complete list of all metrics-based events configured for the project. You can review, edit, or delete existing metrics events, or add new events as needed.

Default Options

- **Event Level:** Indicates the severity of an event and is classified into three levels: Normal (**Info**), Warning (**Warning**), and Critical (**Critical**).
- **Duplicate Agent Alert Suppression:** If two or more agents with the same name run simultaneously, events are not triggered for the selected duration (30 seconds, 1 minute, 3 minutes, 5 minutes, 10 minutes).
- **Search:** Search events by event name.
- **JSON Download/Upload:** Edit event rules directly in the JSON editor, or download them as a file, modify them, and upload them again.
- **+ Add Event:** Add a new metrics event.
- **Resize Columns:** Adjust column widths by dragging column borders.

Table | Metrics Event List Structure

| Item | Description |
|---|--|
| No. | Event sequence number |
|  | Event activation status
– Enabled: An event is triggered when conditions are met
– Disabled: The event does not run and conditions are not evaluated |
|  | Edit or delete event settings |
| Event Name | User-defined event name
– Up to 255 characters |

| Item | Description |
|-----------------------|---|
| Rule | Event trigger conditions |
| Target | Monitoring targets to which the event applies |
| Trigger Count | The time window and number of occurrences required to trigger an event
– An event is triggered when conditions are met the configured number of times within the selected duration |
| Pause | Temporarily suppresses the same event after an alert is triggered
– Available durations: Disabled, 5 min, 10 min, 15 min, 20 min, 30 min, 1 hr, 2 hrs, 3 hrs, 6 hrs, 12 hrs, 1 day
– If recovery notifications are enabled, alerts are not triggered for the selected duration after a RECOVERED notification is sent |
| Recovery Notification | Whether to receive a RECOVERED (normal) notification when Critical or Warning events are resolved
– Enabled: Sends a RECOVERED notification when the status changes from in progress to normal and allows viewing active and resolved states in event history
– Disabled: Triggers a one-time event each time the threshold is exceeded |
| Event Recipients | Users or groups that receive notifications for the event (recipient tags) |

Event Configuration

Metrics events allow you to define events based on conditions that occur in monitored targets and configure notifications so you can respond efficiently.

Add Event

To add a metrics event, go to **Event Configuration > Metrics** tab, then click the [+ Add Event] button in the upper-right corner.

| How to Add a Metrics Event |
|--|
| Step 1. Select Template: Choose the basic structure of the event. |
| Step 2. Define Event Conditions: Configure the conditions that determine when the event is triggered. |
| Step 3. Select Event Target: Select the monitoring targets where the event applies. |
| Step 4. Basic Information and Notification Settings: Configure the event name, message, and recipients. |

Select Template

- Depending on how you want to configure the event, select one of the three templates on the left side of the screen.
 - Create New:** Freely configure categories and metrics from scratch. This option is suitable for advanced users who need complex conditions or fine-grained monitoring rules.
 - Quick Setup:** Provides predefined categories, metrics, and default thresholds. You can apply it quickly by adjusting only the threshold values, which makes it suitable for beginners.
 - Advanced Setup:** Add conditions or freely modify settings based on predefined categories and metrics.

ⓘ Metrics events support multiple configuration approaches. The available templates may vary depending on the monitoring platform.

Define Event Conditions

In the **Define Event Conditions** step, configure the metrics that determine when the event is triggered, along with the trigger count, pause settings, recovery notifications, and event active time.

Metric Settings

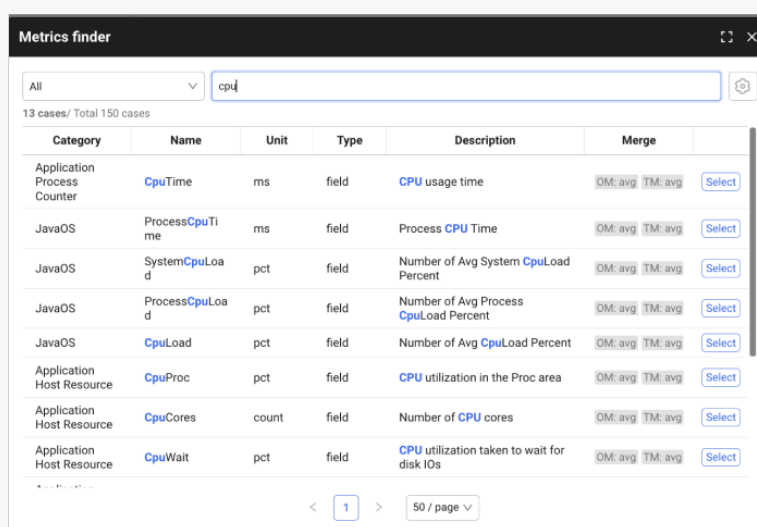
- Enter a category in the search field, or select a category (metric) from the category list. You must select a category to proceed with metric configuration.

- The category list shows the category name, data collection interval, and key. It displays metrics data collected in the project within the last 3 hours.
2. If the metric you need is not listed, select **Direct Input** and enter the category manually.

❗ **If you are not sure about categories or metrics, use the Metrics Finder.**

When configuring metrics for the first time, the Metrics Finder helps you locate metrics more quickly.

- Search for a metric in the Metrics Finder to view related categories and detailed information.
- Click the [Select] button for a metric in the search results to automatically apply the selected value to the event configuration screen.



3. Select the event level (**Critical**, **Warning**, Info). You can configure multiple levels for a single metric.

❗ **Event Level Behavior**

Even when multiple conditions are satisfied, only the highest-priority level (**Critical** → **Warning** → **Info**) triggers an event. For details, see the [Level Action Guide](#).

- Event levels are classified as Critical (**Critical**), Warning (**Warning**), and Info (Info).



- It is recommended to configure multiple levels for the same metric with stepwise thresholds.

4. Enter the field, operator, and value for the selected event level.

a. Click + **Add** to add more conditions.

- **&&** : An event is triggered only when **all conditions are satisfied simultaneously**
- **||** : An event is triggered when **at least one condition is satisfied**

b. Click — to remove the condition.



For condition syntax and supported operators, refer to the [Condition Settings Guide](#).

Trigger Count

Configure how many times a condition must be met before an event is triggered. This is useful for detecting persistent issues only.

1. Select the trigger frequency.

- **Consecutive:** An event is triggered when the condition occurs consecutively for the configured number of times.
- **Recent:** An event is triggered when the condition occurs the configured number of times within the selected time window.
 - Available durations: 5 seconds , 10 seconds , 15 seconds , 30 seconds , 1 minute , 2 minutes , 3 minutes , 4 minutes , 5 minutes , 10 minutes , 30 minutes , 1 hour

2. Enter the trigger count.

- The **Interval** value corresponds to the data collection interval of the selected category. (Interval: 10s)

Pause

To prevent excessive event generation, select a pause duration for event execution. After an event occurs, the same event will not be triggered again during the configured time period.

- Available durations: Disabled , 5 minutes , 10 minutes , 15 minutes , 20 minutes , 30 minutes , 1 hour , 2 hours , 3 hours , 6 hours , 12 hours , 1 day

! When recovery notifications are enabled

Even if the same event condition is met again, no notification is sent until the configured time has passed since the **RECOVERED** notification was received.

Resolved Notification

When the resolved notification feature is enabled, you can track state changes after an event occurs (in progress → resolved). The event is shown as an ongoing event in the **Event History** menu.

- Resolved notifications apply only to **Critical** and **Warning** levels. Info-level events always operate as one-time events.
- When a **Critical** or **Warning** event is resolved, a notification with the RECOVERED (normal) state is sent.
- Click the toggle button to **enable** or **disable** this feature.
 -  **Enabled:** Events operate based on state. They enter an in-progress state when the threshold is exceeded and transition to a resolved state when the value recovers below the threshold.
 -  **Disabled:** A one-time event is generated each time the threshold is exceeded.

Event Operation Time

Configure tags so that events operate only during specific time periods (such as business hours, non-business hours, or maintenance windows). If no tag is configured, events operate continuously for 24 hours while enabled.

1. Click **+ Add**.
2. In Event Operation Time, click **+ Create New Tag**.

3. In **Create Event Operation Time Tag**, select the **tag name, days, time, and color**, then click the [Apply] button.
4. You can view created tags in the tag list, and apply them by selecting the checkbox.
5. To modify or delete a tag, click the  icon and update or delete it in **Edit Event Operation Time Tag**.

 When a tag is deleted, it is removed from all users to whom it was applied. However, tags that are currently in use by event rules cannot be deleted.

Simulation

The simulation feature lets you pretest configured event conditions using historical data. If the expected number of events is too high or too low, you can adjust the conditions to find optimal threshold values.

- **Rule Validation:** Verify how newly created event rules would behave when applied to historical data.
- **Threshold Tuning:** Adjust thresholds to an appropriate level when events occur too frequently or not at all.

Start Simulation

1. After completing event condition settings, click the [Simulation] button.
2. Review the expected number of events and the visualized results.

Understanding Simulation Results

1. Time-Series Data

A graph displays metric changes over time along with threshold lines for each level, allowing you to intuitively identify when events are triggered.

2. Event Counts by Level

The number of events for the Critical, Warning, and Info levels is shown on the right side of the screen. This helps you understand overall event distribution and frequency.

3. Target-Level Details

Hover over a specific point on the chart to see the exact metric value and monitoring target at that moment. When multiple targets exist, they are distinguished by color. If patterns differ by target, you can use filtering to analyze them individually.

4. Peak Interval Identification

Identify periods where metrics spike sharply to understand when issues occur most frequently or to recognize recurring patterns.

✓ Simulation Tips

- **Simulate Different Time Ranges**

Run simulations for weekdays vs. weekends and peak vs. off-peak hours to set optimal thresholds for each time period.

- **Include Actual Incident Periods**

Include past outage periods in simulations to verify whether current rules accurately detect real incidents.

- **Add Target Filtering**

When many targets are monitored, simulation results can become complex. Use filters to focus on specific targets for clearer insights.

- **Adjust Thresholds Gradually**

Instead of making large changes at once, adjust thresholds incrementally, such as in 10 percent steps, to find optimal values.

Select Event Target

If no event target is specified, events are triggered for all targets in the project, which can result in excessive events. It is therefore recommended to clearly define the targets to which the event applies.

ⓘ If you change the event targets, the number of events may change. Click the **Simulation** button and run it again to check the updated expected event count.

Target Selection

You can specify the targets to which the event applies using either **Select Input** or **Direct Input**.

- **Select Input** **Direct Input**

1. Select or enter a tag, operator, and value.
 - a. If it is difficult to find a tag, you can search for it using the **Metrics Finder**.
2. Click + **Add** to add event targets using conditions (`&&` , `||`).

Enter the tag, operator, and value of the target to monitor directly.

- When you choose direct input, a list of available tags (**Tag Select Input**) is provided below the input field.

```
ex. endsWith(okindName, 'example_name') && container == 'prod.billing'
ex. ${4xxErrorType} == '401'
```

 If you change the input method, any content you have entered will be reset.

Basic Information and Notification Settings

Configure the **event name**, **message**, and **recipients**. Because this information is displayed as is in event history and actual notification messages, it should be clear and easy to identify.

Event Activation

Configure whether the event is triggered when the defined conditions are met.

-  **Enabled:** The event is triggered when the configured conditions are satisfied.
-  **Disabled:** Conditions are not evaluated.

Event Name

Enter the event name to be used as the event title. The configured event name is displayed in the **Event List** and **Event History** menus and can be used as a search keyword.

Message

Enter the event message. The message you enter is used in event history and notifications. Click the  icon to view previously written message history. You can include variables in the message to insert actual values at the time the event occurs.

- **Variable Usage Rules**

- Variable format: `${metric_name}`
- Only metrics within the **same category as the configured category** can be used as variables.
- You can combine multiple variables to create a more detailed message.
- The list of available variables can be found in the **Metrics Query** menu.

- **Variable Usage Example**

| | Event Configuration | When Event Occurs |
|---------|-------------------------------|---------------------|
| Title | Disk Usage Increase | Disk Usage Increase |
| Message | Disk = <code>\${disk}%</code> | Disk = 89.9% |

Event Recipients

Specify the members who will receive event notifications.

- **Receive All Receive by Tag Selection**

Notifications are sent to all targets (members and channels) in the project that have event reception enabled.

Notifications are sent to project members who have the selected tag.

1. Select **Receive by Tag Selection**.
2. In the **Reception Tag** section, click the [+ Add Tag] or [+] button.
3. Under the **Event Reception Tag** window, click + **Create New Tag**.
4. In the **Create Tag** window, enter a tag name, select a color, and click the [Create Tag] button.
5. You can edit or delete created tags by clicking the  icon in the tag list.

6. In the **Event Reception Tag** window, select the desired tag from the **Tag List** to apply it.

ⓘ Reception Tags

Reception tags are a feature for managing users or channels that receive notifications **at a group level**. When a reception tag is assigned to an event, notifications are delivered only to users associated with that tag. If no reception tag is set, notifications are sent to all users in the project.

Edit/Delete Event

1. Go to **Alert Notifications > Event Settings** and open the **Metrics** tab.
2. In the event list, click the  icon for the event you want to edit or delete.
3. In the **Edit Event Rule** window, update the options and click the [Save] button.
 - a. To delete the selected event, click the [Delete] button in the upper-right corner of the **Edit Event Rule** window.

Event Sharing

You can save metric event settings as a JSON file to share configurations with other users or import settings created by others.

- JSON file name: `event-rules-YYYYMMDD.json`

Export

1. Click the [**JSON** ] button at the top right of the screen.
2. When the JSON editor opens, click the [ Export] button.
 - If you use the export function after searching for events, only the searched list will be downloaded as a JSON file.
3. After the JSON file is downloaded, share it with other users.

Import

1. Click the [] button at the top right of the screen.
2. Select the JSON file downloaded using the **Export** function.
3. When the JSON editor opens, choose either [Add to List] or [Overwrite].

 It is recommended to use this feature between projects of the same product type. You can import event settings from projects of different products, but they may not function correctly.

Edit in JSON Format

1. Click the [**JSON** ] button at the top right of the screen.
2. When the editor opens, modify the content according to the JSON format.
3. After completing the edits, click the [Save] button at the bottom of the screen.

 If the modified content does not conform to the JSON format, an error message will be displayed at the bottom of the screen and the content cannot be saved. The error message may vary depending on the type of formatting issue.

JSON error

JSON Data Structure

```
{
  "version": 2,
  "eventId": "zcef7s7f27rum1",
  "enabled": true,
  "stateful": false,
  "title": "Active Transaction",
  "message": "Active Transaction = ${active_tx_count}",
  "category": "app_counter",
  "alertLabel": [
    "oid"
  ],
  "repeatCount": 1,
  "repeatDuration": 0,
```



```

"silent": 300000,
"receiver": [],
"timeTag": [],
"selectString": "",
"conditions": [
  {
    "level": 20,
    "enabled": true,
    "rule": "active_tx_count > 100"
  }
],
"createTime": 1763632674345,
"lastModifiedTime": 1763632674347,
"lastModifiedUser": "support@whatap.io",
"basic": true,
"metaId": "java004",
"selectCondition": {
  "oid": [
    "-1010758404",
    "-250906941"
  ]
}
}

```

Table | JSON Data Structure

| JSON Field | Type | Description | Notes |
|------------|---------|-------------------------------------|------------|
| version | Integer | Version of the event rule | Restricted |
| eventId | String | Unique identifier of the event rule | |
| enabled | boolean | Whether the event is enabled | |
| stateful | boolean | Whether the event is state-based | |
| title | String | Event rule name defined by the user | |
| message | String | Event rule message defined by the | |

| JSON Field | Type | Description | Notes |
|------------------|--------------|---|--|
| | | user | |
| category | String | Data category | |
| alertLabel | List<String> | Default identifier (primary key) value assigned per category for event state management | Restricted |
| repeatCount | Integer | Number of repetitions | |
| repeatDuration | Long | Duration for repetition | |
| silent | Integer | Pause duration | |
| receiver | List<String> | List of recipient tag keys | |
| timeTag | List<String> | List of operation-time tag keys | |
| selectString | String | Target selection | |
| conditions | List<Object> | List of trigger conditions by level | |
| ◇ level | byte | Level value | Critical : 30, Warning : 20, Info : 10 |
| ◇ enabled | boolean | Whether the level is enabled | |
| ◇ rule | String | Trigger condition for the level | |
| createTime | Long | Initial creation time of the event rule | |
| lastModifiedTime | Long | Last modification time of the event rule | |
| lastModifiedUser | String | Account that last modified the event rule | |

| JSON Field | Type | Description | Notes |
|-----------------|---------|---|---|
| basic | boolean | Whether the event was created using Quick Setup | Restricted |
| metaId | String | Unique identifier of the Quick Setup template | Restricted: This value exists only for events created via Quick Setup (basic=true). |
| selectCondition | Object | Target selection value for Quick Setup | This value exists only for events created via Quick Setup (basic=true). |

 Fields marked as “Restricted” may affect event behavior, so it is recommended not to modify their values.

Appendix

Trigger Conditions and Target Selection Guide

Metric alert trigger conditions and event target selection use the same syntax. However, trigger conditions use the field key as the variable, while target selection uses the tag key as the variable.

Level Behavior Guide

Event levels are categorized into Critical, Warning, and Info.

1. Priority-based Event Triggering

When multiple level conditions are met at the same time, only the event with the highest priority is triggered.

Example

Configuration:

- Warning: CPU > 70%
- Critical: CPU > 90%

Current state: CPU 95%

→ Result: Only the Critical event is triggered (Warning is suppressed)

2. Level Escalation

For events that maintain an in-progress state, when a higher-level condition is met while a lower-level event is already in progress, both levels enter the in-progress state.

Example

Configuration:

- Warning: CPU > 70%
- Critical: CPU > 90%

Scenario:

- 1) CPU 80% → Warning triggered (in progress)
- 2) CPU 95% → Warning (in progress), Critical (in progress)

3. Level De-escalation

For events with an in-progress state, when the level transitions from a higher level to a lower level, the higher-level event is resolved while the lower-level event remains active.

Example

Configuration:

- Warning: CPU > 70%
- Critical: CPU > 90%

Scenario:

- 1) CPU 97% → Critical triggered
- 2) CPU 85% → Critical resolved, Warning remains
- 3) CPU 65% → Warning resolved

4. Execution Flow

- Warning: CPU > 70%

- Critical: CPU > 90%

Metric value: 60% → 75% → 92% → 85% → 60%

State: Info Warning Critical Warning Info
 (triggered) (triggered) (maintained) (resolved)
 Warning
 (maintained)

- ⓘ Events that have an in-progress state remain active as long as their conditions are met and continue until those conditions are cleared.

Condition Configuration Guide

- ⓘ If a metric name starts with a number or contains special characters, you must wrap it in the `${metric_name}` format.

- Brackets `()`, `[]`
- Operators `+`, `-`, `*`, `/`, `%`
- Delimiters `:`, `@`, `#`, `,`, `,`, (space)
- Other special characters `!`, `^`, `&`, `|`, `~`, ```, `"`, `=`

Examples using `${}` :

```
${cpu(xos)} > 50
${mem[0]} >= 100
${cpu-usage} > 80
${namespace:cpu} > 70
${metric name} > 60
${4xx_error} > 10
```

Usable without `${}` : Metric names that contain only alphabetic characters, underscores (`_`), or dots (`.`)



```
cpu > 80
cpu_usage > 50
CPUUtilization.Average > 0.8
```

1. Comparison Operators

Numeric Comparison

```
Greater than:      cpu > 80
Greater than or equal:  cpu >= 80
Less than:         memory < 1000
Less than or equal:  memory <= 1000
Equal to:          status == 200
Not equal to:       error != 0
```

String Comparison

```
status == 'OK'
region == "us-east-1"
```

2. Arithmetic Operators

- Basic arithmetic operations

```
Addition:    cpu + 10 >= 90
Subtraction:  memory - 100 >= 500
Multiplication:  cpu * 2 >= 100
Division:       disk / 1024 >= 100
Modulo:         value % 10 == 0
```

- Controlling precedence with parentheses

```
(cpu + memory) * 2 >= 200
(disk - used) / total >= 0.2
((cpu + memory) / 2) >= 50
```

- Negative values

```
cpu > -100
```

3. Logical Operators

- AND operator (&&)

```
cpu > 80 && memory > 1000  
${cpu(xos)} > 50 && ${mem(xos)} > 60
```

- OR operator (| |)

```
cpu > 90 | | memory > 90  
disk < 10 | | network > 1000
```

- Combined logical operations

```
(cpu > 50 && memory > 50) | | disk < 20  
cpu > 80 && (memory > 1000 | | disk > 500)
```

4. Pattern Matching Operators

- LIKE operator

```
oname like 'prod-*'  
url like '*error*'  
message like 'WARN%'
```

- NOT LIKE operator

```
oname not like 'test-*'  
url not like '*debug*'
```

5. Built-in Functions

- Null check functions

```

isNull(value)      # Checks whether the value is null
isNotNull(value)   # Checks whether the value is not null
nvl(value, 0)      # Returns a default value if null
isEmpty(str)       # Checks whether the string is empty
isNotEmpty(str)    # Checks whether the string is not empty

```

- Aggregate functions

```

sum(cpu, memory, disk)    # Sum
avg(cpu, memory)          # Average
max(cpu, memory, disk)    # Maximum
min(cpu, memory, disk)    # Minimum
count(value1, value2, value3) # Count

```

- Math functions

```

round(cpu, 2)    # Rounds to 2 decimal places

```

- String functions

```

length(str)      # String length
startsWith(str, 'prefix') # Checks prefix
endsWith(str, 'suffix') # Checks suffix
indexOf(str, 'search') # String index
substring(str, 0, 10) # Substring
trim(str)        # Trims whitespace
replace(str, 'old', 'new') # String replacement
hasStr(str, 'search') # Checks string containment

```

- Conditional functions

- **if:** A conditional function that returns the `trueValue` if the condition evaluates to `true`, or the `falseValue` if it evaluates to `false`

Syntax

```

if(condition, trueValue, falseValue)

```


Example

```
if(value >= 90, 'High', 'Medium') == 'High'
```

- **decode:** Compares a value against conditions and returns the result of the matching condition, or a default value if no match is found

Syntax

```
decode(value, condition1, result1, condition2, result2, ..., conditionN, resultN, defaultValue)
```

Example

```
decode(value, 1, 'Low', 2, 'Medium', 3, 'High', 'Unknown') == 'Unknown'
```

- **in:** Returns `true` if the value matches any of the specified candidates; otherwise returns `false`

Syntax

```
in(value, candidate1, candidate2, ..., candidateN)
```

Example

```
in(status, 200, 201, 204) == false
```

! Writing Guidelines

- Recommended
 - Use simple variable names without `${}` (e.g. `cpu`, `memory`)
 - Always use `${}` when special characters are included
 - Use parentheses to clearly separate complex expressions
 - Enter function names exactly as defined (case-sensitive)
- Avoid



- Incomplete expressions (e.g. `cpu > , memory && )`)
- Mismatched parentheses (e.g. `(cpu > 80 && cpu > 70) )`)
- Consecutive or invalid operators (e.g. `cpu >> 80 , cpu >< 80 )`)

Composite Metrics

Home > Select Project >  **Alert** > **Event Configuration** > **Composite metrics** tab

You can create events and send alert notifications using complex rules.

Default Options

- **Search:** Search for specific events by selecting an event name or an event recipient tag
- **JSON Download/Upload:** Edit event rules directly in the JSON editor, or download them as a file, modify them, and upload them again
- **Alert Message Settings:** Select detailed fields for alert messages and edit the content for each field
- **+ Add Event:** Add a new composite metrics event

Permissions

To configure **Composite metrics** events, you must have **Alert Settings** permission. You can grant this permission by clicking the  icon for the member in **Home > Select Project > Management > Integrated Member Management**.

Table | Composite Metrics Event List Layout

| Item | Description |
|--------------------------|--|
| Event Title | User-defined event name
- Up to 255 bytes |
| Condition | Event trigger conditions |
| Notify on Event Recovery | Whether to receive RECOVERED (normal) notifications when Critical or Warning events are resolved
– Enabled: Sends a RECOVERED notification when the state changes from in-progress to normal, and allows you to distinguish in-progress and resolved states in event history
– Disabled: Triggers a one-time event each time the threshold is exceeded |

| Item | Description |
|---|---|
| Event Recipient Tag | Users or groups that receive notifications for the event |
|  | Edit or delete the event |
|  | Event enablement status
– Enabled: Event is triggered when conditions are met
– Disabled: Event does not run and conditions are not evaluated |

Event Configuration

Composite metrics events define event conditions based on **MXQL**, the metric data query language. The **Create from Chart** feature provides a combo box to support MXQL auto-completion.

Add Event

1. Click the **Composite metrics** tab from  **Alert > Event Configuration**.
2. Click the [+ Add Event] button on the right side of the screen.
3. In the **Composite metrics** window, click **Create from Chart**.
 - a. You can use the provided templates.
4. In the **Event Configuration** screen, first query the event data, then configure the event conditions.

Query Event Data

Query event data to build charts. In the **Event Configuration** screen, select either the **Widget** or **Text** option under **Query event data** on the right to configure events.

- **Widget Text**

Using the options for configuring a time series chart, you can automatically complete the **MXQL** used for event configuration.

1. Select the date range for querying data.
2. Select the **Category** (target) for querying data.
3. Click + **Add Filter** under **Filter** to create event conditions.
 - a. In the filter window, select an expression, tag, and filter value.
 - b. Click + **Add** to add additional conditions.
 - c. To delete a condition, click the – icon next to it.
4. Under **Grouping**, select the grouped metric data. Multiple selections are supported.
5. Under **Time Unit**, set the time interval for grouped data by selecting `seconds` , `minutes` , or `hours` .
6. Under **Field**, select the fields to be used for event trigger conditions. Multiple selections are supported.

 MXQL entered in text mode is not compatible with widget mode.

An editor appears where you can directly modify **MXQL**.

1. Edit **MXQL** directly and click the **Query** button.
2. After selecting a date, you can view the results as a chart or graph.

Notification

Enter the basic information for alert notification settings.

- **Enable Event:** When the toggle is enabled, the event becomes active.
- **Level:** Select the event level (Critical, Warning, Info).
 - **Notify on Event Recovery:** When the toggle is enabled, a notification is sent when the event is resolved and enters the normal (RECOVERED) state.
- **Title:** Enter a title for the event rule.
- **Message:** Enter the content of the alert message delivered when the event is triggered. You can use variables in the message to include actual values at the time the event occurs.

Message example

✔ Variable Usage Rules

- Variable format: `${metricName}`
- Only metrics within the **same category as the configured category** can be used as variables.
- You can combine multiple variables to compose a detailed message.
- The list of available variables can be checked in **Metric Query**.

Event Trigger Conditions

- **Data Query Range:** Select the real-time data query window.
 - Available ranges: 30 seconds , 1 minute , 2 minutes , 5 minutes , 10 minutes , 30 minutes , 1 hour , 2 hours , 3 hours
- **Condition:** Enter the event trigger condition. Only fields included in the event data query can be used.

Additional Information

- **Interval:** The alert condition is evaluated at the selected interval (1 minute , 5 minutes , 10 minutes , 30 minutes , 1 hour , 12 hours).
- **Silence:** After an event is triggered, the same event is suppressed for the selected duration (unset , 1 minute , 5 minutes , 10 minutes , 30 minutes , 1 hour , 12 hours).
- **Event Active Time:** Configure tags to control when the event runs, such as working hours, non-working hours, or maintenance windows. If no tag is set, the event runs continuously for 24 hours while it is enabled.
 1. Click + **Add**.
 2. Under **Event Active Time**, click + **Create New Tag**.
 3. In **Create Event Active Time Tag**, select a **Tag Name**, **Days**, **Time**, and **Color**, then click the [Apply] button.
 4. Created tags appear in the tag list and can be applied by selecting the checkbox.
 5. To edit or delete a tag, click the  icon and modify or remove it in **Edit Event Active Time Tag**.

 When a tag is deleted, it is removed from all users to whom it is applied. However, tags that are

 currently in use by event rules cannot be deleted.

- **Event Recipient Tags:** When recipient tags are configured, notifications are sent to project members who have those tags. If no recipient tag is set, notifications are sent to all users in the project.

1. Click + **Add Tag** or the + icon.
2. Click + **Create New Tag** at the bottom of the **Event Recipient Tags** window.
3. In the **Create Tag** window, enter a tag name, select a color, and click the [Create Tag] button.
4. To edit or delete a created tag, click the  icon in the tag list.
5. Select the desired tag from the **Tag List** in the **Event Recipient Tags** window to apply it.

 When configuring events, selecting event recipient tags allows notifications to be sent to project members and third-party plugins associated with those tags. You can assign tags to project members and third-party plugins separately via **Alert > Notifications**.

Test Event Rules

You can test the configured conditions against historical data to estimate how many events would have been triggered.

1. Select the date for testing the event rules and click the [Confirm] button.
2. Click the [Run] button to check the number of triggered events.

Edit/Delete Event

1. Go to the **Alert > Notifications > Composite metrics** tab.
2. In the composite metrics list, click the  icon of the event you want to edit or delete.
3. In the **Notifications** screen, update the options and click the [Save] button.
 - a. To delete the selected event, click the [Delete] button in the upper right corner of the **Notifications** screen.

Event Sharing

You can save metric event settings as a JSON file to share configurations with other users or import settings created by others.

- JSON file name: `event-rules- YYYYMMDD .json`

Export

1. Click the [**JSON** ] button at the top right of the screen.
2. When the JSON editor opens, click the [ Export] button.
 - If you use the export function after searching for events, only the searched list will be downloaded as a JSON file.
3. After the JSON file is downloaded, share it with other users.

Import

1. Click the [] button at the top right of the screen.
2. Select the JSON file downloaded using the **Export** function.
3. When the JSON editor opens, choose either [Add to List] or [Overwrite].

 It is recommended to use this feature between projects of the same product type. You can import event settings from projects of different products, but they may not function correctly.

Edit in JSON Format

1. Click the [**JSON** ] button at the top right of the screen.
2. When the editor opens, modify the content according to the JSON format.
3. After completing the edits, click the [Save] button at the bottom of the screen.

 If the modified content does not conform to the JSON format, an error message will be displayed at the

! bottom of the screen and the content cannot be saved. The error message may vary depending on the type of formatting issue.

JSON error

Alert Message Settings

You can select detailed fields for the messages to be received and edit the content for each field.

Appendix

MXQL Syntax Guide

Composite metrics operates based on MXQL queries. For detailed syntax, refer to the [MXQL Syntax Guide](#).

Hitmap Patterns

Home > Select Project >  **Alert** > **Event Configuration** > **Hitmap pattern** tab

Hitmap pattern automatically detects anomalies based on time-based pattern changes in application execution ranges, such as response time distribution and call density.

Without manually configuring complex conditions or thresholds, the system analyzes Hitmaps collected in real time and automatically triggers alert notifications when patterns deviate from normal behavior. Hitmap pattern alerts are especially useful in the following situations.

- When a specific transaction becomes excessively concentrated during certain time periods compared to normal behavior
- When traffic distribution or response patterns differ from the usual baseline
- When pattern-based anomaly detection is required beyond what condition-based rules can capture

Event Configuration

Configure the options for the event patterns to be detected.

Activation

When the toggle is turned on, an alert notification is sent when the corresponding pattern is detected.

-  **Enabled:** An event is triggered when conditions are met.
-  **Disabled:** No event is triggered, and conditions are not evaluated.

Silence

After an event is triggered, the event is suppressed for the selected duration and is not recorded in **Event History**.

Event Recipient Tags

When recipient tags are configured, notifications are sent to project members who have those tags. If no recipient tag is set, notifications are sent to all users in the project.

1. Click **+ Add Tag** or the **+** icon.
2. Click **+ Create New Tag** at the bottom of the **Event Recipient Tags** window.
3. In the **Create Tag** window, enter a tag name, select a color, and click the **[Create Tag]** button.
4. To edit or delete a created tag, click the  icon in the tag list.
5. Select the desired tag from the **Tag List** in the **Event Recipient Tags** window to apply it.

Event Active Time

Configure tags to control when events run, such as working hours, non-working hours, or maintenance windows. If no tag is set, events run continuously for 24 hours while they are enabled.

1. Click **+ Add**.
2. Under **Event Active Time**, click **+ Create New Tag**.
3. In **Create Event Active Time Tag**, select a **Tag Name**, **Days**, **Time**, and **Color**, then click the **[Apply]** button.
4. Created tags appear in the tag list and can be applied by selecting the checkbox.
5. To edit or delete a tag, click the  icon and modify or remove it in **Edit Event Active Time Tag**.

 When a tag is deleted, it is removed from all users to whom it is applied. However, tags that are currently in use by event rules cannot be deleted.

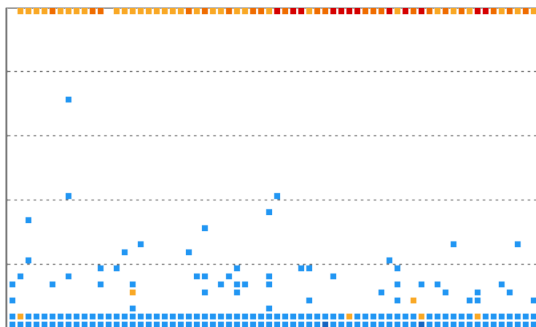
Hitmap Pattern Types

When you turn on the toggle for each pattern, an event is triggered when the corresponding pattern is detected.

Hitmap Horizontal Line Pattern

This pattern appears when multiple transactions complete within a similar response time range. Requests cluster around a specific latency band, which indicates increasing service delay or unstable responses from external resources.

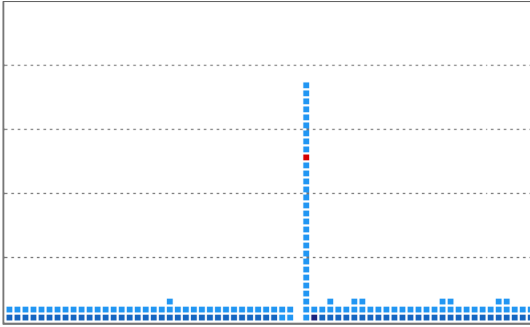
- Timeouts or delays occurring in external API, DB, or HTTPS calls
- Requests waiting for a certain duration before being processed all at once
- Repeated latency in shared I/O or network paths



Hitmap Vertical Line Pattern

In this pattern, transactions start at different times but finish simultaneously. Multiple requests are processed at the same moment, which indicates a bottleneck in a shared resource.

- Increased wait time due to shared resources such as DB locks or file locks
- Concurrent processing delays in queues or thread pools
- Temporary request suspension followed by simultaneous resumption due to system pauses such as GC pauses

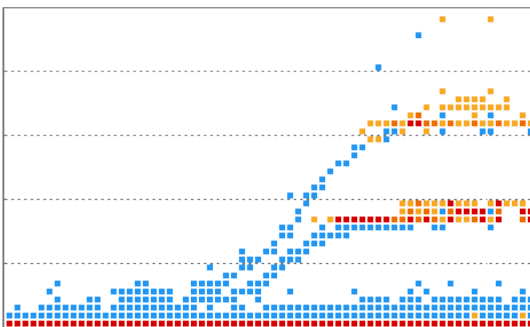


- ✓ A Hitmap vertical line pattern is a signal that lock contention or bottleneck resources should be inspected.

Hitmap Flying Pattern

This pattern appears when response times rise and fall in waves at regular intervals due to shortages in shared resources such as specific system components or logs.

- Shortages in shared resources such as GC, connection pools, or file handles
- Periodic latency factors such as log flushing or storage I/O
- Batch jobs or background jobs that periodically consume resources

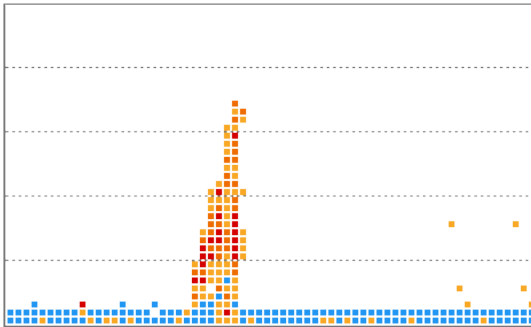


- ✓ A Hitmap flying pattern strongly indicates periodic bottlenecks or resource shortages.

Hitmap Overload Pattern

This pattern occurs when transactions become abnormally concentrated within a specific response time range. It appears when requests fail to distribute normally and instead pile up in a narrow band due to temporary load spikes or early signs of failure.

- Transactions accumulate in a queue due to temporary failures or delays and are then processed in bulk
- Backend service response latency
- Insufficient processing capacity caused by traffic concentration during specific time periods

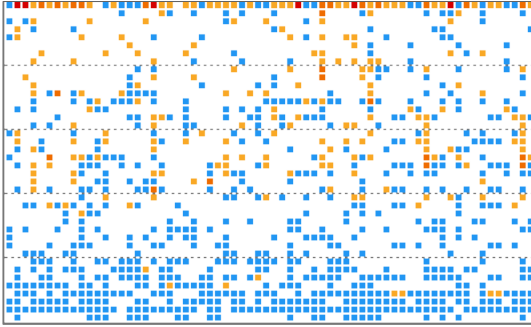


✓ An overload pattern indicates temporary service quality degradation or load balancing failure.

Hitmap Flood Pattern

This pattern indicates an overall increase in response times caused by excessive requests or load. It reflects a broad performance degradation across the service.

- Sudden spikes in user traffic or automated traffic
- System resource shortages such as CPU, memory, or GC pressure
- Overall response quality degradation due to backend bottlenecks



- ✓ A flood pattern requires immediate investigation.

Anomaly Detection

Home > Select Project >  **Alert** > **Event Configuration** > **Anomaly detection** tab

Anomaly detection identifies abnormal rising or falling patterns in metric data and triggers events when movements that differ from the usual behavior are detected.

Default Options

- **Search:** Search for specific events by selecting an event name or an event recipient tag.
- **+ Add Event Rule:** Add a new anomaly detection event.

Table | Anomaly Detection Event List Layout

| Item | Description |
|---|---|
| Event Message | Message sent when an event is triggered |
| Category | Unit used to classify metric data and serve as the basis for metric event configuration |
| Field | Metric field used to detect anomalies |
| Filter | Conditions that limit the data range subject to anomaly detection |
| Rising/Falling Pattern | Sensitivity of rising or falling patterns |
| Event Recipient Tag | Users or groups that receive notifications for the event |
|  | Edit event settings |
|  | Delete event |
|  | Event enablement status
- Enabled: Event is triggered when conditions are met
- Disabled: Event does not run and conditions are not evaluated |

Event Configuration

You can configure alert notifications to be sent when metric values rise or fall in a way that differs from their usual behavior.

Add Event Rules

1. Click the [+ Add Event Rule] button on the right side of the screen.
2. Configure **Event Basic Information**, **Metrics**, **Anomaly Detection**, and **Additional Information** in order.

Event Basic Information

1. Select the event level (**Critical**, **Warning**, Info).
 - This determines the event priority and alert level.
2. Enter the alert message to be delivered when the event is triggered.
 - You can use variables in the message to include actual values at the time the event occurs.

! Available Variables

- `$pcode` : Project code
- `$category` : Category
- `$field` : Field
- `$value` : Triggered value
- `$oname` : Event target

Metrics

1. Select a **Category**.
 - This category serves as the baseline for the metric data used to detect anomalies.

2. Select the **Field** to be used for the event trigger conditions. Multiple selections are supported.
3. Select filters that define the event condition targets used to narrow the event scope. Multiple selections are supported.

Anomaly Detection

1. Select the sensitivity levels for **Rising Pattern** and **Falling Pattern**.
2. Use the toggle buttons to enable or disable each pattern.
 - To detect only rising patterns: Rising Pattern **ON**, Falling Pattern **OFF**
 - To detect only falling patterns: Falling Pattern **ON**, Rising Pattern **OFF**

Additional Information

Event Active Time

Configure tags to control when an event runs, such as working hours, non-working hours, or maintenance windows. If no tag is set, the event runs continuously for 24 hours while it is enabled.

1. Click **+ Add**.
2. Under **Event Active Time**, click **+ Create New Tag**.
3. In **Create Event Active Time Tag**, select a **Tag Name**, **Days**, **Time**, and **Color**, then click the [Apply] button.
4. Created tags appear in the tag list and can be applied by selecting the checkbox.
5. To edit or delete a tag, click the  icon and modify or remove it in **Edit Event Active Time Tag**.

 When a tag is deleted, it is removed from all users to whom it is applied. However, tags that are currently in use by event rules cannot be deleted.

Event Recipient Tags

When event recipient tags are configured, notifications are sent to project members who have those tags. If no recipient tag is set, notifications are sent to all users in the project.

1. Click + **Add Tag** or the + icon.
2. Click + **Create New Tag** at the bottom of the **Event Recipient Tags** window.
3. In the **Create Tag** window, enter a tag name, select a color, and click the [Create Tag] button.
4. To edit or delete a created tag, click the  icon in the tag list.
5. Select the desired tag from the **Tag List** in the **Event Recipient Tags** window to apply it.

✓ When configuring events, selecting event recipient tags allows notifications to be sent to project members and third-party plugins associated with those tags. You can assign tags to project members and third-party plugins separately via **Alert > Notifications**.

Edit Event

1. Go to the **Alert > Event Configuration > Anomaly detection** tab.
2. In the **Anomaly Detection** list, click the  icon on the right of the event you want to edit.
3. In the **Anomaly Detection** window, update the options and click the [Save] button.

Delete Event

1. Go to the **Alert > Event Configuration > Anomaly detection** tab.
2. In the **Anomaly Detection** list, click the  icon on the right of the event you want to delete.
3. In the delete confirmation window, click the [Delete] button.

Real-time Log Events

Home > Select Project >  **Alert** > **Event Configuration** > **Real-time Log** tab

You can set alert notifications by filtering collected log data against conditions. Log monitoring must be activated to use log events. For more information on **activating log monitoring**, refer to the log documentation for each product.

- **Real-time log event:** An event is triggered when the configured event conditions are met in real-time collected logs.

Default options

- **Search:** Search for events by selecting an event name or event notification tag
- **JSON download/upload:** Event rules can be directly edited through the JSON editor, or downloaded as a file, modified, and re-uploaded
- **Notification message settings:** Select detailed items for the message to receive and edit the content for each item
- **+ Add event:** Add a new log event

Table | Real-time log event list configuration

| Item | Description |
|--------------|--|
| Event name | Event name used as the notification title
- Up to 255 bytes |
| Category | Log classification name (log folder name) |
| Search key | The field to search for in log data
- Example: To access an HTTP response status code, use the search key <code>status</code> |
| Search value | The actual value corresponding to the search key. An alert is sent when logs contain the specified value |

| Item | Description |
|---|---|
| | <ul style="list-style-type: none"> - Example: If the search key is <code>status</code> and the search value is <code>200</code>, an alert is triggered when logs containing HTTP status code 200 are collected |
| Pause event notifications | <p>Pauses the same event notification for a specified time after an event notification is triggered</p> <ul style="list-style-type: none"> - Available times: Disabled, 5 min, 10 min, 15 min, 20 min, 30 min, 1 hour, 2 hours, 3 hours, 6 hours, 12 hours, 1 day - If resolved notifications are enabled, no notification is sent for the same event during the selected time after receiving a normal (RECOVERED) status notification |
| Event target filtering | <p>Conditions that define the targets to which the event applies</p> <ul style="list-style-type: none"> - If no condition is set, alerts are evaluated for all collected data - Select input: Build conditions by selecting tags, operators, and values - Direct input: Manually enter conditions to specify targets |
| Event notification tags | User or group (notification tag) to receive the event notification |
|  | Edit and delete events |
|  | <p>Whether the event is active</p> <ul style="list-style-type: none"> - Enabled: Event occurs when conditions are met - Disabled: Event does not operate and conditions are not checked |

Event settings

Log events allow you to configure alert notifications when logs are collected in real time or when logs that meet specific conditions accumulate.

Add real-time log event

1. Click the **Real-time Log** tab in **Alert > Event Configuration**.

2. Click the [+ Add event] button on the right side of the screen.
3. Set the log event conditions in the **Add event rule** window.
 - **Event name:** Enter a name to use as the event title.
 - **Event activation:** When the toggle button is activated, an event occurs when conditions are met.
 - **Level:** Select the event level (**Critical**, **Warning**, **Info**).
 - **Message:** Enter the notification message to receive when the event is triggered. You can use variables in the message to include actual values at the time of event occurrence.

✓ **Variable usage rules**

- Variable format: `${metricName}`
 - Only metrics within the **same category as the configured category** can be used as variables.
 - You can combine multiple variables to compose a detailed message.
 - A list of available variables can be checked in **Metrics Query**.
- **Category:** Select the log classification name (log folder name) from the list or enter it directly. You must select a category before entering a search key and search value.
 - **Search key:** Enter the field you want to search for in log data.
 - Example: To access an HTTP response status code, use the search key `status`.
 - **Search value:** Enter the actual value corresponding to the **search key**. An alert is sent when logs contain the specified value.
 - Example: If the **search key** is `status` and the **search value** is `200`, an alert is triggered when logs containing HTTP status code 200 are collected.
 - **Event target filtering:** Enter conditions to define the targets to which the event applies. If no value is entered, alerts are evaluated for all collected data.
 - **Select input:** Select a tag, operator, and value to set the target conditions.

ⓘ For condition expression syntax and supported operators, refer to the [Condition](#)

! settings guide.

- **Direct input:** Enter conditions directly.
- **Pause event notifications:** Select a duration to temporarily suppress alerts. After the event is triggered, the same event will not occur during the selected time.
 - Available times: Disabled , 1 min , 2 min , 3 min , 5 min , 10 min , 20 min , 30 min , 1 hour , 3 hours , 6 hours , 12 hours
- **Event operation time:** Set tags to activate the event during specific time periods (work/non-work/maintenance hours, etc.). If no tag is set, the event operates 24 hours a day when activated.
 1. Click + **Add**.
 2. Click + **Create new tag** in **Event operation time**.
 3. In **Create event operation time tag**, select the **tag name, day of week, time, and color**, then click the [**Apply**] button.
 4. Created tags can be viewed in the tag list and are applied by selecting the checkbox.
 5. Click the  icon for the tag to modify or delete, and edit or delete the tag in **Edit event operation time tag**.

! Deleting a tag removes the tag for all users to whom it has been applied. However, tags in use by event rules cannot be deleted.

- **Event notification tags:** If notification tags are set for an event, notifications are sent to project members with those tags. If no recipient tags are set, notifications are sent to all users in the project.
 1. Click + **Add tag** or +.
 2. Click + **Create new tag** at the bottom of the **Event notification tags** window.
 3. In the **Create tag** window, enter the tag name, select a color, and click the [**Create tag**] button to create the tag.

4. Edit or delete created tags by clicking the  icon in the tag list.
5. Select the desired tag from the **tag list** in the **Event notification tags** window to apply it.

✓ When configuring events, you can send notifications to project members and 3rd-party plugins with the selected event notification tags. You can assign tags to project members and 3rd-party plugins respectively in **Alert notifications > Event notification settings**.

4. When configuration is complete in the **Add event rule** window, click the [Save] button.

Edit/delete events

1. Go to the **Real-time log** tab in **Alert > Event Configuration**.
2. Click the  icon for the event to edit or delete in the log event list.
3. Modify options in the **Event settings** screen and click the [Save] button.
 - a. To delete the selected event, click the [Delete] button in the upper right of the **Event settings** screen.

Event sharing

You can save metric event settings as a JSON file to share configurations with other users or import settings created by others.

- JSON file name: **event-rules-** YYYYMMDD **.json**

Export

1. Click the [**JSON** ] button at the top right of the screen.
2. When the JSON editor opens, click the [ **Export**] button.
 - If you use the export function after searching for events, only the searched list will be downloaded

as a JSON file.

3. After the JSON file is downloaded, share it with other users.

Import

1. Click the [] button at the top right of the screen.
2. Select the JSON file downloaded using the **Export** function.
3. When the JSON editor opens, choose either [Add to List] or [Overwrite].

 It is recommended to use this feature between projects of the same product type. You can import event settings from projects of different products, but they may not function correctly.

Edit in JSON Format

1. Click the [**JSON** ] button at the top right of the screen.
2. When the editor opens, modify the content according to the JSON format.
3. After completing the edits, click the [Save] button at the bottom of the screen.

 If the modified content does not conform to the JSON format, an error message will be displayed at the bottom of the screen and the content cannot be saved. The error message may vary depending on the type of formatting issue.

JSON error

Appendix

Trigger Conditions and Target Selection Guide

Metric alert trigger conditions and event target selection use the same syntax. However, trigger conditions use the field key as the variable, while target selection uses the tag key as the variable.

Level Behavior Guide

Event levels are categorized into Critical, Warning, and Info.

1. Priority-based Event Triggering

When multiple level conditions are met at the same time, only the event with the highest priority is triggered.

Example

```
Configuration:  
- Warning: CPU > 70%  
- Critical: CPU > 90%  
  
Current state: CPU 95%  
→ Result: Only the Critical event is triggered (Warning is suppressed)
```

2. Level Escalation

For events that maintain an in-progress state, when a higher-level condition is met while a lower-level event is already in progress, both levels enter the in-progress state.

Example

```
Configuration:  
- Warning: CPU > 70%  
- Critical: CPU > 90%  
  
Scenario:  
1) CPU 80% → Warning triggered (in progress)  
2) CPU 95% → Warning (in progress), Critical (in progress)
```

3. Level De-escalation

For events with an in-progress state, when the level transitions from a higher level to a lower level, the higher-level event is resolved while the lower-level event remains active.

Example

Configuration:

- Warning: CPU > 70%
- Critical: CPU > 90%

Scenario:

- 1) CPU 97% → Critical triggered
- 2) CPU 85% → Critical resolved, Warning remains
- 3) CPU 65% → Warning resolved

4. Execution Flow

- Warning: CPU > 70%
- Critical: CPU > 90%

Metric value: 60% → 75% → 92% → 85% → 60%

| | | | | | |
|--------|------|-------------|--------------|--------------|------------|
| State: | Info | Warning | Critical | Warning | Info |
| | | (triggered) | (triggered) | (maintained) | (resolved) |
| | | | Warning | | |
| | | | (maintained) | | |

- ⓘ Events that have an in-progress state remain active as long as their conditions are met and continue until those conditions are cleared.

Condition Configuration Guide

- ⓘ If a metric name starts with a number or contains special characters, you must wrap it in the `${metric_name}` format.

- Brackets `()`, `[]`
- Operators `+`, `-`, `*`, `/`, `%`
- Delimiters `:`, `@`, `#`, `,`, `.` (space)
- Other special characters `!`, `^`, `&`, `|`, `~`, `"`, `=`

Examples using `${}` :



```

${cpu(xos)} > 50
${mem[0]} >= 100
${cpu-usage} > 80
${namespace:cpu} > 70
${metric name} > 60
${4xx_error} > 10

```

Usable without `${}` : Metric names that contain only alphabetic characters, underscores (`_`), or dots (`.`)

```

cpu > 80
cpu_usage > 50
CPUUtilization.Average > 0.8

```

1. Comparison Operators

Numeric Comparison

```

Greater than:      cpu > 80
Greater than or equal:  cpu >= 80
Less than:         memory < 1000
Less than or equal:  memory <= 1000
Equal to:          status == 200
Not equal to:       error != 0

```

String Comparison

```

status == 'OK'
region == "us-east-1"

```

2. Arithmetic Operators

- Basic arithmetic operations

```

Addition:    cpu + 10 >= 90
Subtraction: memory - 100 >= 500
Multiplication: cpu * 2 >= 100

```

```
Division:  disk / 1024 >= 100
Modulo:    value % 10 == 0
```

- Controlling precedence with parentheses

```
(cpu + memory) * 2 >= 200
(disk - used) / total >= 0.2
((cpu + memory) / 2) >= 50
```

- Negative values

```
cpu > -100
```

3. Logical Operators

- AND operator (&&)

```
cpu > 80 && memory > 1000
${cpu(xos)} > 50 && ${mem(xos)} > 60
```

- OR operator (| |)

```
cpu > 90 | | memory > 90
disk < 10 | | network > 1000
```

- Combined logical operations

```
(cpu > 50 && memory > 50) | | disk < 20
cpu > 80 && (memory > 1000 | | disk > 500)
```

4. Pattern Matching Operators

- LIKE operator

```
oname like 'prod-*'
url like '*error*'
```

```
message like 'WARN%'
```

- NOT LIKE operator

```
oname not like 'test-*'
url not like '*debug*'
```

5. Built-in Functions

- Null check functions

```
isNull(value)      # Checks whether the value is null
isNotNull(value)   # Checks whether the value is not null
nvl(value, 0)      # Returns a default value if null
isEmpty(str)       # Checks whether the string is empty
isNotEmpty(str)    # Checks whether the string is not empty
```

- Aggregate functions

```
sum(cpu, memory, disk)  # Sum
avg(cpu, memory)        # Average
max(cpu, memory, disk)  # Maximum
min(cpu, memory, disk)  # Minimum
count(value1, value2, value3) # Count
```

- Math functions

```
round(cpu, 2)          # Rounds to 2 decimal places
```

- String functions

```
length(str)           # String length
startsWith(str, 'prefix') # Checks prefix
endsWith(str, 'suffix') # Checks suffix
indexOf(str, 'search') # String index
substring(str, 0, 10)  # Substring
```

```
trim(str)          # Trims whitespace
replace(str, 'old', 'new') # String replacement
hasStr(str, 'search') # Checks string containment
```

- Conditional functions

- **if:** A conditional function that returns the `trueValue` if the condition evaluates to `true`, or the `falseValue` if it evaluates to `false`

Syntax

```
if(condition, trueValue, falseValue)
```

Example

```
if(value >= 90, 'High', 'Medium') == 'High'
```

- **decode:** Compares a value against conditions and returns the result of the matching condition, or a default value if no match is found

Syntax

```
decode(value, condition1, result1, condition2, result2, ..., conditionN, resultN, defaultValue)
```

Example

```
decode(value, 1, 'Low', 2, 'Medium', 3, 'High', 'Unknown') == 'Unknown'
```

- **in:** Returns `true` if the value matches any of the specified candidates; otherwise returns `false`

Syntax

```
in(value, candidate1, candidate2, ..., candidateN)
```

Example

```
in(status, 200, 201, 204) == false
```

Writing Guidelines

- Recommended
 - Use simple variable names without `${}` (e.g. `cpu`, `memory`)
 - Always use `${}` when special characters are included
 - Use parentheses to clearly separate complex expressions
 - Enter function names exactly as defined (case-sensitive)
- Avoid
 - Incomplete expressions (e.g. `cpu >` , `memory &&`)
 - Mismatched parentheses (e.g. `(cpu > 80 && cpu > 70)`)
 - Consecutive or invalid operators (e.g. `cpu >> 80` , `cpu >< 80`)

Composite Log Event Settings

Home > Select Project >  **Alert** > **Event Configuration** > **Composite Log** tab

You can set alert notifications by filtering collected log data against conditions. Log monitoring must be activated to use log events. For more information on **activating log monitoring**, refer to the log documentation for each product.

- **Composite log event:** An event is triggered when the number of logs that meet the log search conditions within the configured **data query range** reaches or exceeds the threshold count.

Permission

Alert configuration permission is required to configure composite metrics events. Alert configuration permission can be set by clicking the  icon for the member to grant permission in **Home Screen** > **Select Project** > **Management** > **Integrated Member Management**.

Default options

- **Search:** Search for events by selecting an event name or event notification tag
- **JSON download/upload:** Event rules can be directly edited through the JSON editor, or downloaded as a file, modified, and re-uploaded
- **Notification message settings:** Select detailed items for the message to receive and edit the content for each item
- **+ Add event:** Add a new log event

Table | Composite log event list configuration

| Item | Description |
|------------|---|
| Event name | Event name used as the notification title |
| Template | Composite log template |

| Item | Description |
|---|---|
| Category | Log classification name (log folder name) |
| Rule | Conditions that trigger the event |
| Additional notification when event is resolved | <p>Whether to receive notifications in normal (RECOVERED) status when Critical and Warning level events are resolved</p> <ul style="list-style-type: none"> - Enabled: Sends a RECOVERED notification when transitioning from in-progress → normal status, and the in-progress/resolved status can be distinguished in event history - Disabled: A one-time event occurs each time the threshold is exceeded |
| Pause event notifications | <p>Pauses the same event notification for a specified time after an event notification is triggered</p> <ul style="list-style-type: none"> - Available times: Disabled, 5 min, 10 min, 15 min, 20 min, 30 min, 1 hour, 2 hours, 3 hours, 6 hours, 12 hours, 1 day - If resolved notifications are enabled, no notification is sent for the same event during the selected time after receiving a normal (RECOVERED) status notification |
| Interval | Time range for querying log data |
| Event notification tags | User or group (notification tag) to receive the event notification |
|  | Edit and delete events |
|  | <p>Whether the event is active</p> <ul style="list-style-type: none"> - Enabled: Event occurs when conditions are met - Disabled: Event does not operate and conditions are not checked |

Event settings

The composite log event feature triggers alert notifications when the number of logs matching the log search conditions within the configured **data query range** reaches or exceeds the threshold count.

Add composite log event

1. Click the **Composite Log** tab in **Alert > Event Configuration**.
2. Click the [+ Add event] button on the right side of the screen.
3. Set the log event conditions in the **Add event rule** window. For details, refer to [Basic settings](#), [Detail settings](#), and [Additional settings](#).

Add composite log event

4. When configuration is complete in the **Add event rule** window, click the [Save] button.

Basic settings

- **Event name:** Enter a name to use as the event title.
- **Event activation:** When the toggle button is activated, an event occurs when conditions are met.
- **Level:** Select the event level (**Critical**, **Warning**, Info).

Detail settings

Configure [Template \(preset\)](#), [Message](#), [Category](#), [Filter](#), and [Event trigger conditions](#).

Template (preset)

Selecting a predefined preset automatically configures the **filter**, **query type**, and **event trigger conditions**. You can modify the settings after selecting a preset.

| Composite log template types | | |
|------------------------------|------------|--------------|
| Preset | Query type | Default rule |
| None | none | rows > 10 |
| 2xx status code count | count | count > 10 |

!

| Preset | Query type | Default rule |
|-------------------------------------|-------------|----------------------------------|
| 3xx status code count | count | count > 10 |
| 4xx status code count | count | count > 10 |
| 5xx status code count | count | count > 10 |
| Normal status code (2xx, 3xx) count | count_v2 | include_minus_exclude_count > 10 |
| Error status code (4xx, 5xx) count | count_v2 | include_minus_exclude_count > 10 |
| Error received count | count | count > 10 |
| Exception received count | count | count > 10 |
| Log count per second | rps | rps > 100 |
| Field aggregation (Aggregation) | aggr | avg > 1000 |
| Rate | rate | rate > 10 |
| Unique value count (Cardinality) | cardinality | cardinality > 100 |
| Percentile | percentile | p99 > 3000 |

Message

Enter the notification message to receive when an event occurs. You can use variables in the message to compose dynamic notifications.

- **Example:** Message `${host} error ${count} occurred` → Notification message web-01 error 15 occurred

Table | Message variables

| Variable | Description | Usage condition |
|-----------------------------|--|----------------------------|
| <code>\${count}</code> | Number of logs matching the condition | Always available |
| <code>\${groupField}</code> | Grouping field name | Grouped notifications only |
| <code>\${groupValue}</code> | Grouping field value | Grouped notifications only |
| <code>\${fieldName}</code> | Grouping field value (referenced directly by field name) | Grouped notifications only |

❗ The `${count}` variable can be used in all notifications regardless of grouping. The `${groupField}`, `${groupValue}`, and `${fieldName}` variables are only substituted when a grouping field is configured.

Category

Select the log classification name (log folder name) from the list or enter it directly. You can enter the search key and search value only after selecting a category.

Filter

Enter log search conditions using Lucene query syntax. Two input modes are supported.

- **Filter mode** : Build a query by selecting fields and values.
- **Code mode** : Enter Lucene queries directly.

Table | Lucene query examples

| Query | Description |
|---|---|
| <code>level:ERROR</code> | Logs where the level field is ERROR |
| <code>status_nxx:status_4xx OR status_nxx:status_5xx</code> | 4xx or 5xx status codes |
| <code>appender:accessLog</code> | Logs from the accessLog appender |
| <code>response_time.n:*</code> | Logs where the response_time.n field exists |

| Query | Description |
|-----------------------------|-----------------------------|
| host:web-01 AND level:ERROR | Error logs from host web-01 |

Event trigger conditions

Configure by query type. Additional settings are displayed depending on the selected template (preset) or query type.

Table | Event trigger condition key meanings

| Key | Meaning |
|-----------------------------|--|
| rows / count | Number of logs matching the filter condition |
| rps | Log count per second (count / interval in seconds) |
| avg | Average value of the target field |
| sum | Sum of the target field values |
| min | Minimum value of the target field |
| max | Maximum value of the target field |
| rate | Numerator count / denominator count × 100 (%) |
| cardinality | Number of unique values in the target field |
| p99 / p95 / p90 / p75 / p50 | Percentile values of the target field |
| include_minus_exclude_count | Include filter count – exclude filter count |

- **Count (count / none):** Counts the number of logs matching the configured filter conditions.

Table | Count settings

| Setting | Description |
|-------------------------|---|
| Filter | Log search condition (Lucene query) |
| Event trigger condition | <code>count</code> or <code>matching logs</code> <code>{operator}</code> <code>{threshold}</code> (e.g., <code>count > 10</code>) |

- **Log count per second (rps):** Calculates the number of logs per second (Requests Per Second) matching the configured filter conditions. The total count during the interval is divided by the interval in seconds.

Table | Log count per second settings

| Setting | Description |
|-------------------------|--|
| Filter | Log search condition (Lucene query) |
| Event trigger condition | <code>RPS</code> <code>{operator}</code> <code>{threshold}</code> (e.g., <code>rps > 0.5</code>) |

- **Field aggregation (aggr):** Calculates the aggregated value (average, sum, min, max) of the specified numeric field. The aggregation method can be selected from a dropdown in the event trigger condition.

Table | Field aggregation settings

| Setting | Description |
|-------------------------|---|
| Filter | Log search condition (Lucene query) |
| Target field | Numeric field to aggregate (e.g., <code>response_time.n</code>) |
| Event trigger condition | <code>avg / sum / min / max</code> <code>{operator}</code> <code>{threshold}</code> (e.g., <code>avg > 1000</code>) |

- **Rate:** Calculates the percentage (%) ratio of the numerator filter count to the denominator filter count. Useful for monitoring the ratio of specific conditions against all logs.

Table | Rate settings

| Setting | Description |
|-------------------------|---|
| Filter (numerator) | Log condition for the numerator (e.g., <code>level:ERROR</code>) |
| Denominator filter | Log condition for the denominator (e.g., <code>level:*</code>) |
| Event trigger condition | <code>Rate (%) {operator} {threshold}</code> (e.g., <code>rate > 10</code>) |

- **Example:** Filter `level:ERROR`, denominator filter `level:*`, rule `rate > 10` → Alert triggered when the error rate among all logs exceeds 10%
- **Unique value count (cardinality):** Counts the number of unique values in the specified field. Useful for detecting an abnormally large variety of values.

Table | Unique value count settings

| Setting | Description |
|-------------------------|---|
| Filter | Log search condition (Lucene query) |
| Target field | Field to calculate unique values for (e.g., <code>host</code>) |
| Event trigger condition | <code>Unique value count {operator} {threshold}</code> (e.g., <code>cardinality > 100</code>) |

- **Percentile:** Calculates the percentile value of the specified numeric field. Useful for monitoring tail latency in response times. Changing the percentile selection automatically updates the key in the event trigger condition.

Table | Percentile settings

| Setting | Description |
|--------------|---|
| Filter | Log search condition (Lucene query) |
| Target field | Numeric field to calculate percentile for (e.g., <code>response_time.n</code>) |
| Percentile | Select from P99, P95, P90, P75, P50 |

| Setting | Description |
|-------------------------|--|
| Event trigger condition | P99 / P95 / ... {operator} {threshold} (e.g., p99 > 3000) |

Table | Percentile meanings

| Percentile | Meaning |
|------------|---|
| P50 | Median. 50% of all data is at or below this value |
| P75 | 75% of all data is at or below this value |
| P90 | 90% of all data is at or below this value |
| P95 | 95% of all data is at or below this value |
| P99 | 99% of all data is at or below this value |

- **Include/exclude count (count_v2):** Calculates the net count by subtracting the exclude filter count from the include filter count.

Table | Include/exclude count settings

| Setting | Description |
|-------------------------|--|
| Filter (include) | Log conditions to include (e.g., status_nxx:status_4xx OR status_nxx:status_5xx) |
| Filter exclude | Log conditions to exclude (e.g., status_nxx:status_4xx) |
| Denominator filter | Denominator for ratio calculation (optional) |
| Event trigger condition | Include-exclude count {operator} {threshold} (e.g., include_minus_exclude_count > 10) |

- **Example:** Filter status_nxx:status_4xx OR status_nxx:status_5xx , filter exclude status_nxx:status_4xx → Alert triggered when the net 5xx count (4xx+5xx minus 4xx) exceeds 10

Group by field (Group By)

When a group by field is configured, alert conditions are evaluated independently for each value of that field. Select the field to group by from the dropdown. Clicking the dropdown shows a list of available fields for the current category with autocomplete.

- **Example:** Filter `level:ERROR` , group by field `host` , rule `rows > 3`

Table | Grouping comparison

| Item | Without grouping | With grouping |
|------------------|------------------------------------|---|
| Evaluation unit | All logs aggregated as one | Independent aggregation per group value |
| Alert occurrence | 1 alert | One per group meeting the condition |
| Alert title | Event title | [Group value] Event title |
| Stateful | State transition based on all logs | Independent state transition per group |

- `[web-01] Error alert` — 8 errors from web-01 → alert triggered
- `[web-02] Error alert` — 5 errors from web-02 → alert triggered
- `[api-01] Error alert` — 2 errors from api-01 → condition not met, no alert

❗ When **Additional notification when event is resolved** is enabled, the ABNORMAL → RECOVERED state transition occurs independently per group. For example, when errors from `web-01` are resolved, a `[web-01] RECOVERED` notification is triggered, with no effect on other hosts.

Additional settings

- **Interval:** Check alert conditions at the selected time interval (`unset` , `1 min` , `5 min` , `10 min` , `30 min` , `1 hour` , `12 hours`).
- **Mute:** After an event occurs, the same event is not triggered during the selected time (`unset` , `1 min` , `5 min` , `10 min` , `30 min` , `1 hour` , `12 hours`).

- **Event operation time:** Set tags to activate the event during specific time periods (work/non-work/maintenance hours, etc.). If no tag is set, the event operates 24 hours a day when activated.
 1. Click **+ Add**.
 2. Click **+ Create new tag** in **Event operation time**.
 3. In **Create event operation time tag**, select the **tag name**, **day of week**, **time**, and **color**, then click the **[Apply]** button.
 4. Created tags can be viewed in the tag list and are applied by selecting the checkbox.
 5. Click the  icon for the tag to modify or delete, and edit or delete the tag in **Edit event operation time tag**.

 Deleting a tag removes the tag for all users to whom it has been applied. However, tags in use by event rules cannot be deleted.

- **Event notification tags:** If notification tags are set for an event, notifications are sent to project members with those tags. If no recipient tags are set, notifications are sent to all users in the project.
 1. Click **+ Add tag** or **+**.
 2. Click **+ Create new tag** at the bottom of the **Event notification tags** window.
 3. In the **Create tag** window, enter the tag name, select a color, and click the **[Create tag]** button to create the tag.
 4. Edit or delete created tags by clicking the  icon in the tag list.
 5. Select the desired tag from the **tag list** in the **Event notification tags** window to apply it.

 When configuring events, you can send notifications to project members and 3rd-party plugins with the selected event notification tags. You can assign tags to project members and 3rd-party plugins respectively in **Alert notifications > Event notification settings**.

Simulation

You can simulate using historical data before saving the alert to check the expected number of alert occurrences.

1. After completing the alert configuration, select a time range at the bottom.
2. Click the [Simulation] button.
3. The chart displays data by time period and the number of alert occurrences.

! You can adjust the simulation time range to check alert occurrence patterns across different time periods. Data from the last 1–2 minutes may not be available due to index build delays.

4. When configuration is complete in the **Add event rule** window, click the [Save] button.

Create alert from Log Explorer

You can create alerts directly from Log Explorer.

1. Navigate to **Log > Log Search** menu.
2. Configure the category and search query.
3. Click the [Add event] button in the upper right of the screen.
4. The alert configuration form opens in a drawer, with the current category and search query automatically filled in.
5. Complete additional settings and click the [Save] button.
6. After saving, click the [Go] button to navigate to the alert list.

Edit/delete events

1. Go to the **Composite log** tab in **Alert > Event Configuration**.
2. Click the  icon for the event to edit or delete in the log event list.
3. Modify options in the **Event settings** screen and click the [Save] button.
 - a. To delete the selected event, click the [Delete] button in the upper right of the **Event settings** screen.

Event sharing

You can save metric event settings as a JSON file to share configurations with other users or import settings created by others.

- JSON file name: `event-rules-YYYYMMDD.json`

Export

1. Click the [**JSON** ] button at the top right of the screen.
2. When the JSON editor opens, click the [ Export] button.
 - If you use the export function after searching for events, only the searched list will be downloaded as a JSON file.
3. After the JSON file is downloaded, share it with other users.

Import

1. Click the [] button at the top right of the screen.
2. Select the JSON file downloaded using the **Export** function.
3. When the JSON editor opens, choose either [Add to List] or [Overwrite].

 It is recommended to use this feature between projects of the same product type. You can import event settings from projects of different products, but they may not function correctly.

Edit in JSON Format

1. Click the [**JSON** ] button at the top right of the screen.
2. When the editor opens, modify the content according to the JSON format.
3. After completing the edits, click the [Save] button at the bottom of the screen.

- ❗ If the modified content does not conform to the JSON format, an error message will be displayed at the bottom of the screen and the content cannot be saved. The error message may vary depending on the type of formatting issue.

JSON error

Transaction

Home > Select Project >  **Alert** > **Event Configuration** > **Transaction** tab

Transaction events provide fine-grained monitoring of transactions based on specific URLs, errors, status codes, and more. When transactions that meet the configured conditions are detected, an event is triggered.

Transaction data is analyzed in 1-minute intervals, and notifications are sent when the number of transactions that satisfy the conditions exceeds the defined threshold.

Latest Agent Versions

To use transaction events, you must install an agent that meets or exceeds the minimum supported version.

| Language | Minimum Supported Version |
|----------|---|
| Java | 2.2.50 or later |
| PHP | Not specified (use the latest version or contact support) |
| Node.js | 0.5.21 or later |
| Python | 1.7.8 or later |
| .NET | 2.3.5 or later |
| Go | Not specified (use the latest version or contact support) |

Default Options

- **Event Level:** Indicates the severity of the event and is classified into three levels: Info, **Warning**, and **Critical**.
- **Search:** Search for events by event name or event reception tag.
- **+ Add Event:** Add a new transaction event.

- **Resize Columns:** Drag the boundary of each column to adjust its width.

Table | Transaction Event List Structure

| Item | Description |
|---|--|
| No. | Event sequence number |
|  | Whether the event is enabled
- Enabled: The event is triggered when conditions are met
- Disabled: The event does not operate and conditions are not evaluated |
|  | Edit or delete event settings |
| Event Name | User-defined event name
- Up to 255 bytes |
| Condition Type | Type of event trigger condition (elapsed time, error occurrence, pattern only) |
| Aggregation Basis | Criteria used to aggregate transaction counts |
| Target Pattern | Include/exclude patterns to which the event applies (URL, status code, etc.) |
| Trigger Count | Number of occurrences required to trigger the event
- Aggregation period: fixed at 1 minute (based on each minute from :00 to :59 seconds; the period cannot be changed)
- The event is triggered when transactions that meet the condition occur at least the configured number of times within that 1-minute period |
| Pause | Temporarily suppresses repeated occurrences of the same event after an alert is sent
- Available durations: Disabled, 5 min, 10 min, 15 min, 20 min, 30 min, 1 hr, 2 hr, 3 hr, 6 hr, 12 hr, 1 day
- If recovery notifications are enabled, no alert is sent for the same event during the selected time after a RECOVERED notification is received |

| Item | Description |
|------------------|--|
| Event Recipients | Users or groups that receive the event notification (reception tags) |

Event Configuration

Transaction events are configured through the steps of **event conditions**, **target patterns**, and **basic information and notification settings**.

With event configuration, you can detect specific transaction patterns and receive alerts.

Add Event

To add a transaction event, go to **Event Configuration > Transaction** tab, then click the [+ Add Event] button in the upper-right corner of the screen.

| How to Add a Transaction Event |
|--|
| Step 1. Define Event Conditions: Define the criteria that trigger the event. |
| Step 2. Select Event Target: Select the monitoring targets where the event applies. |
| Step 3. Basic Information and Notification Settings: Configure the event name, message, and recipients. |

Define Event Conditions

Configure the conditions under which a transaction event is triggered.

Condition Type

Select the criteria used to trigger the event.

- **Elapsed Time:** Transactions that take longer than the specified time (ms).
 - **Elapsed Time Threshold:** When the condition type is Elapsed Time, enter the value in milliseconds. An

event is triggered when a transaction takes longer than the configured time (ms).

- **Error Occurred:** Transactions in which an error occurs.
- **Pattern Only:** Transactions that match a specific pattern (such as hitmap patterns).

Aggregation Criteria

Select the unit by which alert conditions are aggregated.

- Aggregation criteria: per agent, per URL of each agent, per error class of each agent, per business of each agent, per URL, per error class, per business, or overall.

Trigger Count (Required)

Configure how many times the condition must be met before a notification is sent.

Example: trigger count **1** sends a notification immediately if even one transaction meets the condition within 1 minute, while trigger count **3** requires at least 3 transactions meeting the condition to accumulate within the same 1 minute before a notification is sent.

Pause

Select a pause duration to prevent excessive alert notifications.

- After an event occurs, the same event will not trigger again during the configured time period.
- Available durations: 1 minute , 5 minutes , 10 minutes , 15 minutes , 30 minutes , 1 hour , 2 hours , 3 hours , 6 hours , 12 hours , 1 day

Event Operation Time

Configure the time periods during which the event is active. This allows you to **suppress alerts during non-business hours or maintenance windows**. If no tag is configured, the event runs continuously for 24 hours while enabled.

1. Click + **Add**.
2. In Event Operation Time, click + **Create New Tag**.
3. In **Create Event Operation Time Tag**, select the **tag name, days, time, and color**, then click the [Apply]

button.

4. The created tag appears in the tag list. Select the checkbox to apply it.
5. To edit or delete a tag, click the  icon and update or remove it in **Edit Event Operation Time Tag**.

 When a tag is deleted, it is removed from all alerts where it is applied. However, tags currently in use by event settings cannot be deleted.

Select Event Target

Configure patterns for transaction filtering.

Pattern Configuration

- You can enter multiple patterns, separated by pressing **Enter**.
- If no pattern is configured, events apply to all transactions in the project.
- Patterns of the same type are applied with an **OR** condition.
- Patterns of different types are applied with an **AND** condition.
- You can use wildcards (*****) in patterns. Up to two wildcard positions are allowed.
- Each pattern must include at least one character other than the wildcard (*****) and slash (**/**).

Be careful when combining slashes (**/**) and wildcards (*****)

- `/api/*` : `/api` alone does not match. A path must follow `/api/` (e.g.,  `/api/users`,  `/api`)
- `/api*` : `/api` itself also matches (e.g.,  `/api`,  `/api/users`,  `/apitest`)

Match Patterns

Filter data that matches the specified items. After being applied, match patterns are displayed as **blue tags**. You can enter up to 10 patterns for each item.

| Item | Description |
|----------------|--|
| Operation Name | Operation name pattern |
| URL | Transaction URL pattern
- Example: <code>/api/*</code> , <code>*/users/*</code> |
| Error Class | Error class name pattern
- Example: <code>*Exception</code> , <code>*Error</code> |
| Error Message | Error message content pattern |
| Status Code | HTTP status code pattern
- Example: <code>4**</code> , <code>5**</code> |

Exclude Patterns

Exclude data that matches the specified items from the filtering targets. After being applied, exclude patterns are displayed as **red tags** and use the same format as match patterns. You can enter up to 10 patterns for each item.

Pattern Examples

- Exact Match

```
Pattern: "hello"
"hello" → Match
"Hello" → No match (case-sensitive)
```

- Prefix Match

```
Pattern: "hello*"
"hello" → Match
"helloworld" → Match
"hi" → No match
```

- Suffix Match

```

Pattern: "*world"
"world"    → Match
"helloworld" → Match
"hello"    → No match

```

• Substring Match

```

Pattern: "*ell*"
"hello"    → Match (h'ell'o)
"excellent" → Match (exc'ell'ent)
"welcome"  → No match (does not contain "ell")

```

• Composite Patterns

```

Pattern: "hello*world"
"hellobeautifulworld" → Match

Pattern: "hello*test*"
"hellomytestcase"     → Match

Pattern: "*test*world"
"mytestcaseworld"     → Match

```

Table | Practical Usage Examples

| Pattern Type | Examples |
|----------------------|--|
| URL Patterns | <ul style="list-style-type: none"> - <code>/api/*</code>: All URLs that start with <code>/api/</code> - <code>/users/*</code>: All URLs that contain <code>users</code> - <code>/api/v1/users</code>: Only the URL that matches exactly |
| Error Class Patterns | <ul style="list-style-type: none"> - <code>Exception</code>: All errors that end with <code>Exception</code> - <code>java.lang.*</code>: All classes that start with <code>java.lang</code> - <code>Timeout*</code>: All errors that contain <code>Timeout</code> |
| Status Code Patterns | <ul style="list-style-type: none"> - <code>4**</code>: All 4xx status codes - <code>404</code>: Only the 404 status code |

| Pattern Type | Examples |
|--------------|------------------------------|
| | - 5** : All 5xx status codes |

Pattern Limitations

- **Required characters:** Each pattern must include at least one character other than the wildcard (*). A pattern cannot consist of wildcards only.
 -  Not allowed: **, *
 -  Allowed: a*, a, a*, /api*
- **Empty strings:** Empty strings or null values are not matched.
- **Consecutive wildcards:** Consecutive * characters are treated as a single wildcard.
- **Complex patterns:** Patterns containing three or more wildcards are not supported.
- **No regular expressions:** Character classes ([a-z]), quantifiers (+ , ?), and grouping (()) are not supported.
- **Case sensitivity:** All pattern matching is case-sensitive.
- **No escaping:** There is no way to escape characters for literal matching.

Basic Information and Notification Settings

Configure the basic information of the event and how notifications are delivered.

Event Activation

Configure whether notifications are sent when an event occurs.

-  **Enabled:** Notifications are sent when the configured conditions are met.
-  **Disabled:** Notifications are not sent even if the configured conditions are met.

Event Name (Required)

Enter the event name to be used as the notification title.

Event Level (Required)

Select the event level (Info, **Warning**, **Critical**).

Event Recipients

Specify the members who will receive event notifications.

- **Receive All** **Receive by Tag**

Notifications are sent to all targets (members and channels) with event reception enabled in the project.

Notifications are sent to project members who have the selected tag.

1. Select **Receive by Tag**.
2. In the **Receiver Tags** section, click [+ Add Tag] or [+].
3. Click + **Create New Tag** at the bottom of the **Event Receiver Tags** window.
4. In the **Create Tag** window, enter a tag name, choose a color, and click [Create Tag].
5. To edit or delete a created tag, click the  icon in the tag list.
6. In the **Event Receiver Tags** window, select the desired tag from the **Tag List** to apply it.

Receiver Tags

Receiver tags are a feature for managing notification recipients (users or channels) at a group level. When you assign receiver tags to an event, notifications are delivered only to users who have those tags.

If no receiver tag is set, notifications are sent to all users in the project.

Reference. Example of Event Behavior

This is an example of an actual transaction event occurrence.

Configuration Conditions



- Aggregation criteria: By agent
- Occurrence count: 2 times
- Pause: 5 minutes
- URL patterns: `/api/posts/**` , `/delete/*` , `/*test*/`
- Error class patterns: `*TimeoutException*` , `*NullPointerException`
- Status code pattern: `4**`
- Excluded error class: `*IOException`
- Excluded status code: `5**`

Example of a Triggered Notification Message

The transaction information in the event message displays the values from the most recent transaction that matched the conditions. The event message includes both the configured conditions and the details of the actual transaction that triggered the alert, allowing you to clearly identify which conditions caused the notification.

```
(ignored count: 3)
[Per Agent] limit: 2, count: 3
[Condition]
url patterns: /api/posts/**, /delete/*, */test/*
error class patterns: *TimeoutException*, *NullPointerException
status patterns: 4**
exclude error class patterns: *IOException
exclude status patterns: 5**
[Transaction]
url: /api/posts/test/timeout
txid: 408093121470636540
status: 408
error class: java.util.concurrent.TimeoutException
```

Explanation of Event Message Composition

- **(ignored count: n):** The number of transactions that met the conditions but did not trigger a notification during the pause period. This item is not displayed if there are no ignored transactions during the pause period.
- **Aggregation Information:** Displayed differently depending on the configured aggregation method.

- **By Agent:** [Per Agent] limit: n, count: n
- **By URL:** [Per Url] limit: n , followed by a list of each URL and its occurrence count
- **By Error Class:** [Per Error Class] limit: n , followed by a list of each error class and its occurrence count
- **By Context:** [Per Context] limit: n , followed by a list of each context and its occurrence count
- **By Agent & URL:** [Per Agent & Url] limit: n , followed by a list of each URL and its occurrence count
- **By Agent & Error Class:** [Per Agent & Error Class] limit: n , followed by a list of each error class and its occurrence count
- **By Agent & Context:** [Per Agent & Context] limit: n , followed by a list of each context and its occurrence count
- **Total:** [Total] limit: n, count: n
- **[Condition]:** Displays the conditions that were configured at the time the event was triggered. Only configured patterns are shown.
 - **context patterns:** Context match patterns
 - **url patterns:** URL match patterns
 - **error class patterns:** Error class match patterns
 - **error message patterns:** Error message match patterns
 - **status patterns:** Status code match patterns
 - **exclude context patterns:** Context exclusion patterns
 - **exclude url patterns:** URL exclusion patterns
 - **exclude error class patterns:** Error class exclusion patterns
 - **exclude error message patterns:** Error message exclusion patterns
 - **exclude status patterns:** Status code exclusion patterns
- **[Transaction]:** Detailed information about the most recent transaction that met the conditions.
 - **url:** Transaction URL
 - **txid:** Unique transaction identifier
 - **status:** HTTP response status code (not displayed for non-HTTP transactions)
 - **error class:** Error class (not displayed if no error class exists)
 - **error message:** Error message (not displayed if no error message exists)

 Notes

- Transaction events are aggregated and processed in 1-minute intervals.
- If the condition type is **Pattern Only**, at least one match or exclusion pattern must be configured.
- Pattern matching is case-sensitive.
- Notification messages include detailed information about the most recent transaction that met the conditions.

Container

Home > Select project >  **Alert** > **Application Event Configuration** classic

Alert notifications are sent based on event conditions such as container resource usage, active transactions, and error situations.

- ! The classic event configuration is a separated menu that allows continued use of the legacy event configuration method.
For more advanced features, use [Event configuration](#).

Default options

Inactive agent alert

An event occurs when no data is received from an agent for the configured period. You can enable or disable this feature using the toggle button.

-  **Enabled:** An inactive agent event occurs when no data is received for a certain period.
-  **Disabled:** No event occurs regardless of data reception status.

- ! The legacy inactive agent check interval can be configured under **Management > Agent list**.

Send notification when event is resolved

After an event occurs due to threshold conditions, you can choose whether to send a **Recovered** notification when the state returns to normal. Use the toggle button to enable or disable this feature.

-  **Enabled:** Events operate in a state based manner. When thresholds are exceeded, the status becomes ongoing, and when values recover below thresholds, the status changes to recovered.
-  **Disabled:** A one time event occurs each time the threshold is exceeded.

Duplicate agent alert

If two or more agents with the same name run simultaneously, normal monitoring may be affected, so an event is generated. However, duplicate agent events do not occur again within the configured time window.

- To check whether agent names are duplicated, review the related options in the **Event Configuration** menu.
- Available durations: 30 seconds , 1 minute , 3 minutes , 5 minutes , 10 minutes

Conditions

You can configure event trigger conditions per item. Input formats vary depending on the item.

- **Server CPU / Server memory / Server disk**
- **Active transactions / Transaction errors**
- **Transaction response time**

Condition (%)

Set two threshold levels: **Warning** and **Critical**.

Alerts are triggered when thresholds are exceeded, with **Critical** taking priority.

- Left value: Warning threshold (**Warning**)
- Right value: Critical threshold (**Critical**)

Condition (count)

Enter the count used as the event trigger threshold. If the count exceeds the configured value for the specified duration, an event occurs.

Condition (count)

Configure the condition. An event is triggered when the number of transactions exceeding the configured response time reaches or exceeds the specified count.

Condition (response time)

Select the baseline response time.

- Available times: 1 second , 2 seconds , 3 seconds , 5 seconds , 10 seconds , 15 seconds , 30 seconds , 1 minute , 2 minutes , 3 minutes , 5 minutes , 10 minutes , 30 minutes , 1 hour

- Example: If the count threshold is **10**, the response time threshold is **1 second**, and the duration is **5 seconds**, then an event occurs when 10 or more transactions exceed 1 second and the condition persists for 5 seconds.

Duration

An event occurs when the configured condition persists for the selected duration.

- Available durations: 5 seconds , 10 seconds , 15 seconds , 30 seconds , 1 minute , 2 minutes , 3 minutes , 5 minutes , 10 minutes , 30 minutes , 1 hour

Silence

After an event occurs, no additional events with the same condition occur during the selected silence period. Setting an appropriate silence duration helps prevent excessive alerts.

- Available durations: 1 minute , 2 minutes , 3 minutes , 5 minutes , 10 minutes , 30 minutes , 1 hour , 2 hours , 3 hours , 6 hours , 12 hours , 1 day

Event enablement

You can control event behavior using the toggle button.

-  **Enabled:** An event occurs when the configured conditions are met.
-  **Disabled:** Conditions are not evaluated.

Event configuration

You can configure events for the following items to monitor system status in real time. After configuring each item, click the **Save** button at the top right of the screen.

Server CPU / Server memory / Server disk

An event occurs when server resource usage exceeds the configured threshold (%). You can set different thresholds for **Warning** and **Critical** levels.

- Example: If CPU usage is set to **Warning**: 80% and **Critical**: 90%, a **Critical** event occurs when usage exceeds 90%.

Active transactions

Monitors the number of transactions currently being processed. An event occurs when the count exceeds the configured threshold and persists for the configured **duration**. This is useful for detecting situations with a large number of concurrent transactions.

- Default occurrence count: 100

Transaction errors

Transactions that do not complete normally are treated as errors. An event occurs when the error count exceeds the configured threshold and persists.

- Collection interval: 5 seconds

Transaction response time

Events are configured based on the time required for each transaction to complete (response time). You can detect service slowdowns relative to expected response time.

- Example: If the response time threshold is 10 seconds and the count threshold is 10
"An event occurs when 10 or more transactions exceed 10 seconds and the condition persists for 5 seconds."

Event Notification Settings

Home > Select Project >  **Alert** > **Notifications**

Event Notification Settings per User

 Information about other members in **Event notification settings per-user** can only be viewed by members with the **Member Management** permission. For all other members, required information is masked.

- Mobile device management for bulk event notification receipt settings and access control can be managed in **Account Information**.
- For detailed information about member permissions, refer to the [following document](#).

Notification Methods

Alerts are available via [Email](#), [SMS](#), [WhatsApp](#), and [Mobile](#).

In the **Event notification settings per-user** section, select the checkbox for the desired notification method to receive alerts. If you clear the checkbox for a notification method, alerts will not be sent through that channel.

 All users except the project super administrator can change only their own notification settings.

Email Alerts

Email alerts are sent to the email address provided during sign-up.

SMS Alerts

You can receive alerts via SMS.

ⓘ The SMS alert feature is available only for paid projects.

1. Click your **profile icon** at the top right of the screen, then go to **Manage Accounts**.
2. In **Account Information**, click the [General Mobile Phone] button in the **User Phone Number** section.
3. Enter the **phone number** that will receive the verification code.
 - Only **Korean mobile phone numbers** can be registered to receive SMS alerts.
4. Click the [Save] button.

✔ Change or delete the registered phone number

- To change the phone number, click the [Change] button. SMS verification is required.
- To delete the phone number, click the [Delete] button. After deletion, you will no longer receive alerts via SMS.

WhatsApp Alerts

You can receive alerts via WhatsApp.

ⓘ The WhatsApp alert feature is available only for paid projects.

1. Click your **profile icon** at the top right of the screen and go to **Manage Accounts**.
2. In **Account Information**, click the [WhatsApp] button in the **User Phone Number** section.
3. Enter the **phone number** that will receive the verification code.
4. Click the [Send Verification Code] button.
5. Enter the 6-digit verification code sent to your WhatsApp application.
 - If you do not receive the verification code within 10 minutes, click the [Resend Verification Code] button.
6. Click the [Verify] button.

Mobile Alerts

Users who need to receive mobile alerts must install the WhaTap application on their mobile device and log in.

- Download the WhaTap application – [Android](#)
- Download the WhaTap application – [iOS](#)

Repeated Alerts (Escalation) Settings

If the alert condition is not resolved after an alert is triggered, you can configure repeated alert intervals based on the time of the initial alert. This setting applies only to **Critical** level alerts.

For example, to repeat alerts immediately (0 minutes), 1 hour later, and 1 day later after the initial alert, enter `0,1H,1D` in the **Repeated Alerts (Escalation)** column.

1. Go to **Alert > Notifications** and navigate to the **Event notification settings per-user** section.
2. Configure the settings in the **Repeated Alerts (Escalation)** field.
 - **M**: minutes, **H**: hours, **D**: days. If the unit is omitted, the value is interpreted as minutes.
 - Enter a number or a number with a unit (**M**, **H**, **D**). If the input format is invalid, an error message is displayed.
3. Click the [Save] button. The settings will not be applied unless they are saved.

Event Reception Tags

When configuring events, you can select **event reception tags** to send alerts to project members and 3rd-party plugins associated with those tags. If no event reception tag is set, alert notifications are sent to **all project members**.

Add / Edit / Delete Event Reception Tags

 Event reception tags that are **currently applied to an event** cannot be deleted.

1. Go to **Alert > Notifications**.

2. In the user list under the **Event notification settings per-user** section, click + **Add Tag** or +.
3. In the **Event reception tag** window, click + **Create New Tag**.
4. In the **Create Tag** dialog, enter a tag name, select a color, and click the [Create Tag] button.
5. To edit or delete a created tag, click the  icon in the tag list.
6. Select the desired tag from the **Tag List** in the **Event reception tag** window to apply it.

Remove Event Reception Tags

1. Go to **Alert > Notifications** and open the **Event notification settings per-user** section.
2. Click + in the **Event reception tag** field for the user whose event reception tag you want to remove.
3. In the **Event reception tag** window, click the **X** on the tag you want to remove. The tag will be removed.

Receive Alerts Without Reception Tags

When configuring events, you can select event reception tags to send alerts to project members and 3rd-party plugins associated with those tags.

- To receive alerts **without event reception tags**, enable the **Receive alerts without reception tags** option.
- If you want to receive **only alerts that have event reception tags**, disable this option.

 To stop receiving all alerts, disable this option and make sure that no **Event reception tag** is selected.

Selecting the event reception mode

In the **Event notification settings per-user** section, you can select the event alert reception scope per member or third-party channel from the following three modes. The reception mode is selected in the **Event reception tags** column of each member or channel row.

| Mode | Description |
|----------------------------------|--|
| Receive all alerts | Receive all alerts regardless of the event reception tags |
| Receive alerts for selected tags | Receive only alerts that have the selected event reception tags. Select at least |

| Mode | Description |
|-------------------|--|
| only | one tag |
| Receive no alerts | Receive no event alerts. The selected tags are preserved without being deleted |

- In the **Receive alerts for selected tags only** mode, select at least one tag to receive alerts in the tag selection area.
- To also receive alerts that have no event reception tag specified, select the **Unclassified** virtual tag in the tag selection area. The **Unclassified** tag can be selected or deselected like a normal tag, but the tag itself cannot be edited or deleted.

ⓘ In the **Receive no alerts** mode, the corresponding member or channel receives no event alerts at all. The selected tags are preserved without being deleted, so if you switch back to the **Receive alerts for selected tags only** mode, you can use the previously selected tag settings as is.

Select Notification Language

You can choose the language in which you receive alert notifications.

1. Go to **Alert > Notifications**.
2. In the user list under the **Event notification settings per-user** section, click the ► button at the far left of the user whose language you want to configure.
3. Select the **Notification Language**.

Day and Time-based Notification Settings

You can configure whether a user receives notifications by day of the week and time of day.

1. Go to **Alert > Notifications**.
2. In the user list under the **Event notification settings per-user** section, click the ► button at the far left of the user whose day and time settings you want to configure.

3. Select the days and/or enter the time ranges during which you want to receive alerts.

- Settings can be configured separately for each notification method.

3rd-party Plugins

You can receive alert notifications through external applications.

- **Supported applications:** Slack, Telegram, Teams, Teams Workflows, Jandi, AlertNow, Webhook, Webhook JSON, Opsgenie, PagerDuty, LINE, ilert

ⓘ Internal messengers that are not included in WhaTap Labs' supported scope can be integrated via standard **Webhook** or **Webhook JSON**.

Table | 3rd-party Plugin List

| Item | Description |
|------------------------------|---|
| Plugin Name | Name of the plugin |
| Auth Key | Authentication key |
| Auth Value | Authentication value |
| Receive Level | Notification level: Warning, All, or Critical |
| Repeated Alerts (Escalation) | Re-notification interval from the initial alert time if the issue is not resolved
- Applies only to Critical alerts |
| Smart Message (AI) | AI-powered notifications available when adding the Telegram plugin
- Supported languages: Korean, English, Japanese |
| Event Reception Tag | Whether to receive alerts for specific members or teams in the project |
| Delete | Delete the 3rd-party plugin |

Add a 3rd-party Plugin

1. Go to **Alert > Notifications**.
2. In the **3rd-party Plugins** section, click **Add**.
3. In the **Add 3rd-party Plugin** window, select the desired service tab.
4. Follow the instructions for the selected service (see [How to add 3rd-party plugin notifications](#)).
5. Click **Test Receive** to verify the configuration.
6. When finished, click **Add** or **Register**.

How to Add 3rd-party Plugin Notifications

Slack

- ⓘ In the **Add 3rd-party Plugin** window, click the [**Add To Slack**] button on the **Slack** tab to register **WhaTapNotiBot** to your Slack team and channel.

Telegram

- ⓘ
1. In **Telegram**, search for **@WhaTapApmNotiBot**.
 2. Select the bot and click the [**Start**] button.
 3. In the chat window, enter the `/license` command followed by the access key of the project for which you want to receive alerts.

Telegram

ⓘ Available commands

- `/help` : View the list of commands
- `/license` : Enter the access key of the project to receive alerts

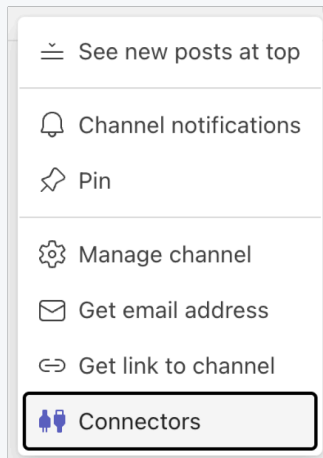


- `/remove` : Enter the project code for which you want to stop receiving alerts
- `/list` : View the list of project codes currently receiving alerts

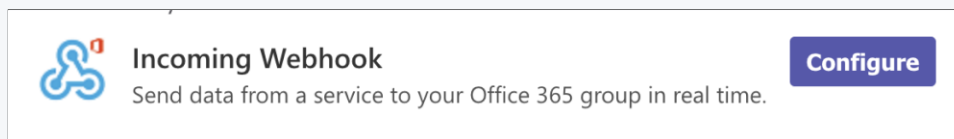
Teams



1. Go to the **connector** of the channel where you want to receive alerts.



2. Add a **Webhook** and click the [Configure] button.



3. Enter a webhook name and click the [Create] button.



Incoming Webhook

[Send feedback](#)

The Incoming Webhook connector enables external services to notify you about activities that you want to track. To use this connector, you'll need to create certain settings on the other service, which needs to support a webhook that's compatible with the [Office 365 connector format](#).

Fields marked with * are mandatory

To set up an Incoming Webhook, provide a name and select Create. *

Customize the image to associate with the data from this Incoming Webhook.

[Upload Image](#)



Default Image

Create

Cancel

4. Copy the generated **Webhook URL**.

Copy the URL below to save it to the clipboard, then select Save. You'll need this URL when you go to the service that you want to send data to your group.

<https://whatap.webhook.office.com/webhook>



Url is up-to-date.

Done

Remove

5. Return to the **Add a third-party plug-in** window in the WhaTap Monitoring service, and enter values for **Channel Name** and **Webhook URL** at the bottom of the **Teams** tab.

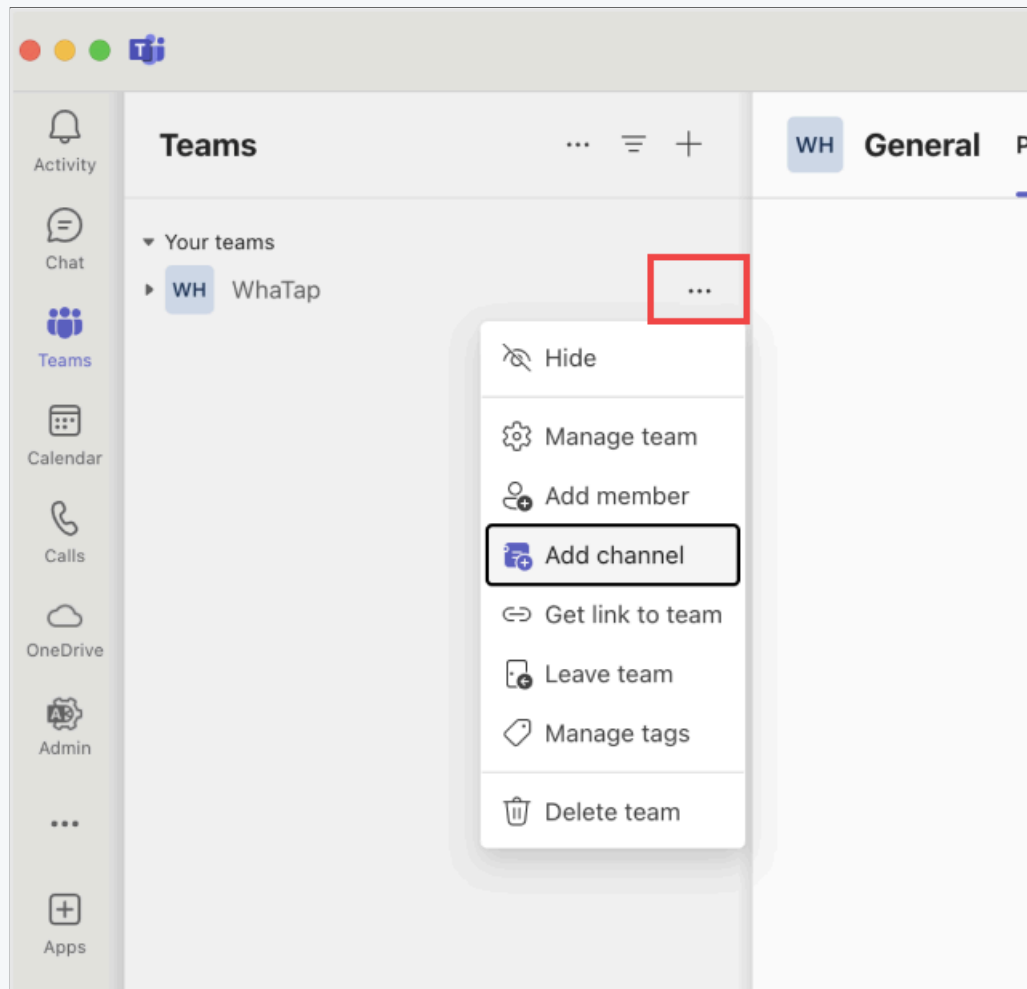
6. Click the [**Registration**] button.

Teams Workflows



- ⓘ To use workflows, you need an account that can access **Microsoft Teams** and a **Business plan** subscription.

1. In the **Teams** menu, click **More actions (...)** > **Add channel**.



2. In the **Create a channel** window, enter the **Channel name** and **Description**. For **Channel type**, select **Standard**, then click the [Create] button.



Create a channel

Channel name *

WhaTap Event Channel

Description

Help others find the right channel by providing a description

Choose a channel type * ⓘ

Select

Standard
Everyone on the team has access.

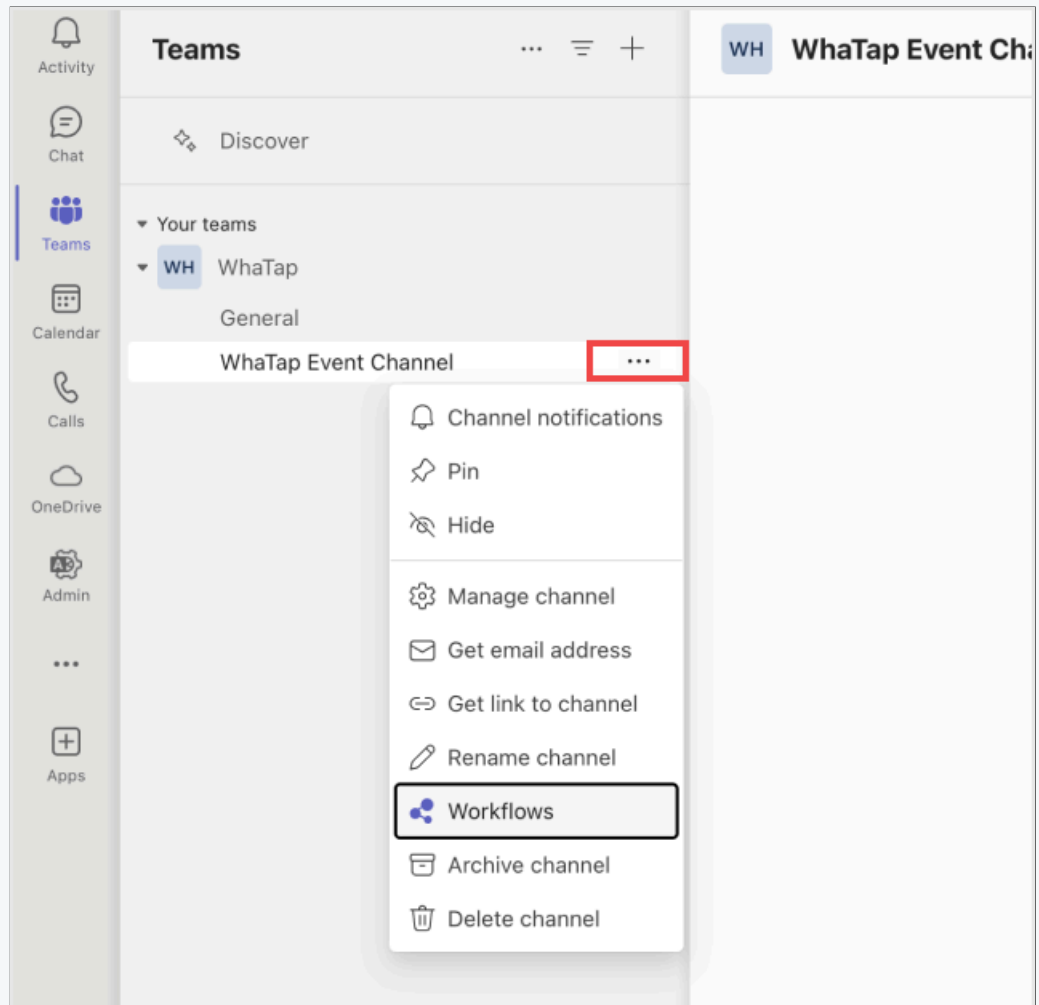
Shared
People or teams in your org or outside your org can have access.

Private
Specific people on the team have access.

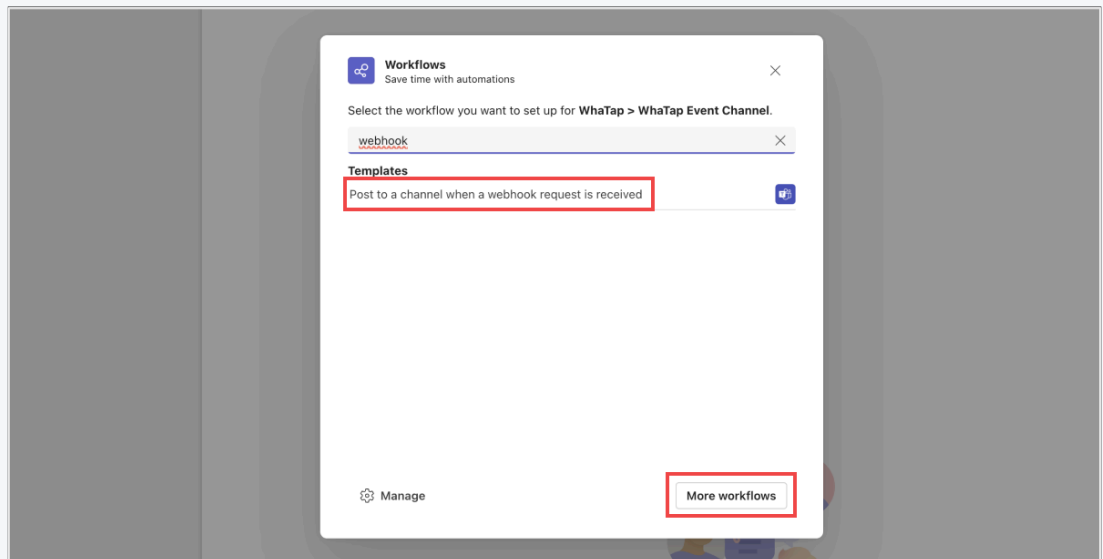
Cancel Create

ⓘ If you select **Shared** as the channel type, workflows cannot be used. If you select **Private**, workflows can be added but notifications cannot be received.

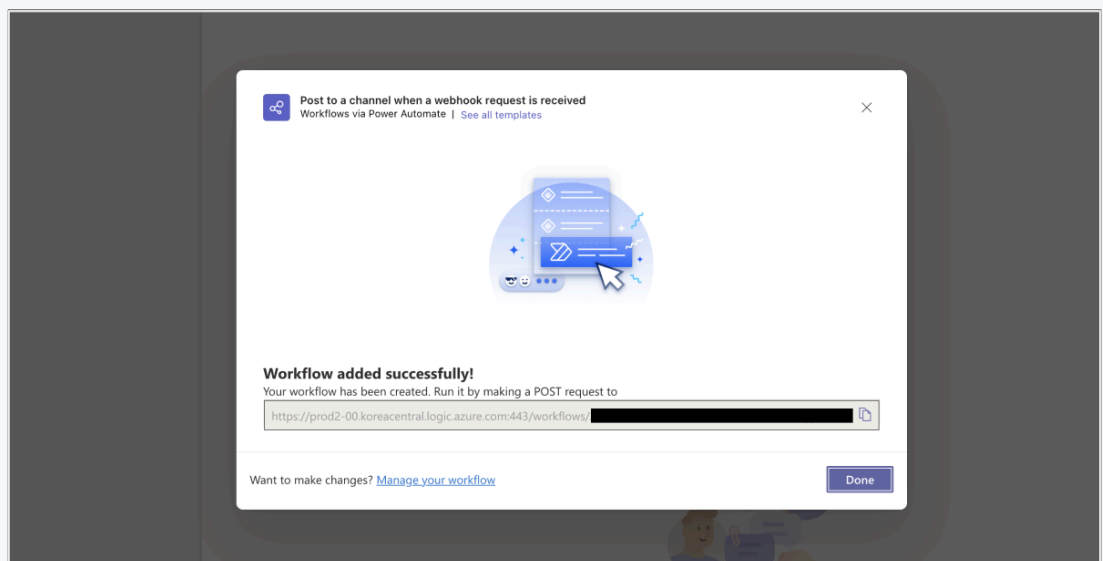
3. Click **More actions (...)** of the created channel, then select **Workflows**.



4. In the **Workflows** dialog, select **Post to a channel when a webhook request is received** from the workflow list, then click the [Add workflow] button.



5. Check the **Name** and **Connection** status, then click the [Next] button.
6. Review the **Details**, then click the [Add workflow] button.
7. The workflow has been added. Copy the generated URL.



8. Return to the **Add a third-party plug-in** window in the WhaTap Monitoring service, and enter values for **Channel Name** and **Webhook URL** at the bottom of the **Teams Workflows** tab.



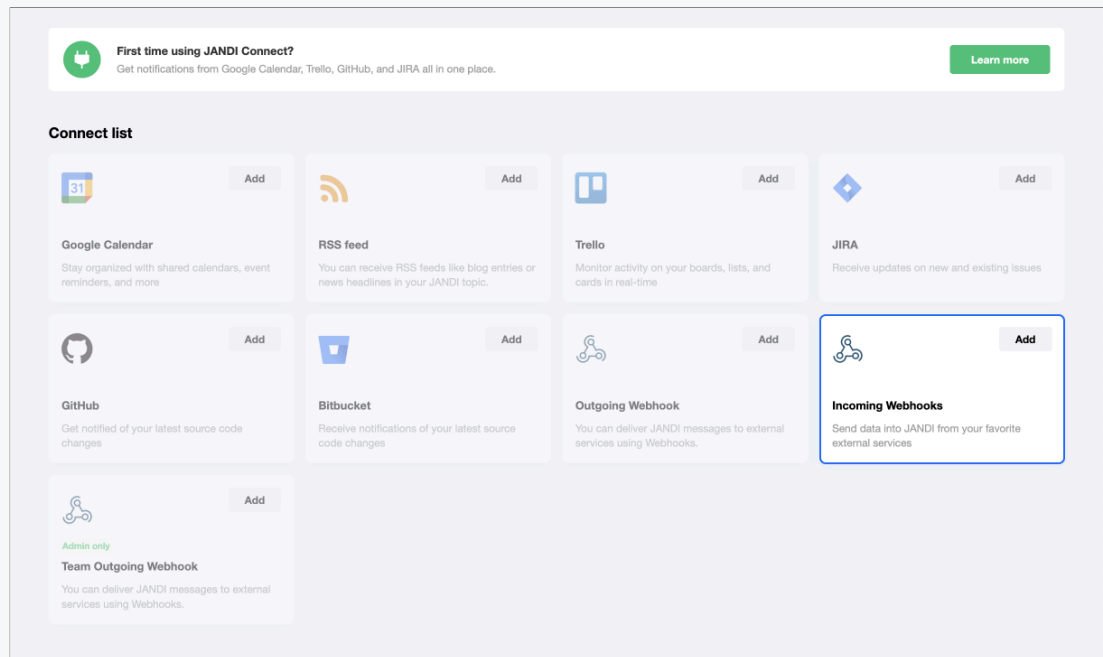
9. Click the [**Registration**] button.



Jandi



1. Go to **Jandi Connect settings** in the topic or chat where you want to receive notifications.
2. In the **Connect list**, find **Webhook Receiver (Incoming Webhook)** and click the [**Connect**] button.



3. Click the [**Add Webhook Receiver**] button.
4. Copy the Webhook URL from the **Register Webhook URL** section.



Connect settings

Profile settings

Change the name or profile image that appears for messages from this integration

 Incoming Webhooks

Topic / JANDI selection

Please select a topic or DM to receive integration messages. You change it after saving it.

JANDI

Webhook URL Registration

Webhook URL

Copy the generated Webhook URL and register with the service.

 <https://wh.jandi.com/connect-api/webhook/31257583/d3...>

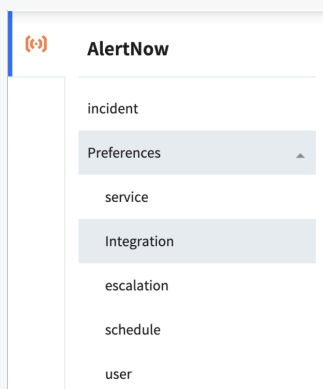
Copy

- Return to the **Add a third-party plug-in** window in the WhaTap Monitoring service, and enter values for **Channel Name** and **Webhook URL** at the bottom of the **Jandi** tab.
- Click the [**Registration**] button.

AlertNow



- Access the **AlertNow** service, then go to **Settings > Integration**.



- On the **Integration** screen, click the [Create Integration] button.
- From the list of integration types, select **WhaTap**.
- On the **Create Integration** screen, configure the **Name** and **Naming Rules**.

Integration

Entire

Integration name

Enter integration name

Integration type

WhaTap

Set naming convention

Define and manage incident naming rules.

naming convention

[[level]][title] {projectName} {oname} {oid}

Preview rule settings

Incident names are limited to 3500 bytes in length.

[Warning]CPU Used > 10 % OTE-SERVER ote-front-330494549

sample data

```
{
  "metricName": "cpu",
  "pcode": 188,
  "level": "Warning",
  "metricValue": "26.888296",
  "oid": "-330494549",
  "title": "CPU Used > 10 %",
  "message": "MEMORY is too high 98%",
  "uid": "4dfe0b7686-c4335f0ef4c1cfa45754",
  "metricThreshold": "18.0",
  "oname": "ote-front",
  "time": 1592887274971,
  "projectName": "OTE-SERVER",
  "status": "on"
}
```

56/1600 byte

5. Click the [Confirm] button to create the integration.
6. Check the list of created integrations and select the newly created integration.

Integration

Create integration

Enter search term

Last update: 2024-02-21 11:31:17

| Integration name | Integration type | Integration key | service name |
|------------------|------------------|-----------------|--------------|
| whatap-test | WhaTap | | whatap-test |

7. On the details screen, click the copy icon on the right side of the URL to copy it.

Integration

whatap-test

Basic Information

| | |
|------------------|--|
| Integration type | WhaTap |
| Integration key | fda527a5140 |
| URL | https://alertnowitgr.opsnow.com/integration/whatap/v1/ |
| constructor | Lisa |

Copy to clipboard

8. Return to the **Add a third-party plug-in** window in the WhaTap Monitoring service, and enter values for **Channel Name** and **Webhook URL** at the bottom of the **AlertNow** tab.
9. Click the [**Registration**] button.

Webhook



1. Register the Webhook that will receive notifications. The alert `title` and `text` are sent to the registered URL via the **POST** method.

```
{
  "title": "Alert Title",
  "text": "Sample warning message"
}
```

2. Enter values for **Webhook Name** and **Webhook URL**.
3. Click the [**Registration**] button.

Webhook JSON



1. Register the Webhook that will receive notifications. Alert data is sent to the registered URL in **JSON format** via the **POST** method. For substitution variables that can be included in the JSON payload, see **Available Variables**.
 - `${something}` represents a substitution value provided by WhaTap. Refer to **Available Variables** for the full list and descriptions.
 - The event status (`status`) is categorized as **event triggered** (`on`) or **event recovered** (`off`).
 - The `level` is categorized as `None` , `Info` , `Warning` , or `Critical` .

```
{
  "pcode": "${pcode}",
  "projectName": "${projectName}",
  "time": "${time}",
  "oid": "${oid}",
  "oname": "${oname}",
  "title": "${title}",
  "message": "${message}",
  "level": "${level}",
  "status": "${status}",
}
```



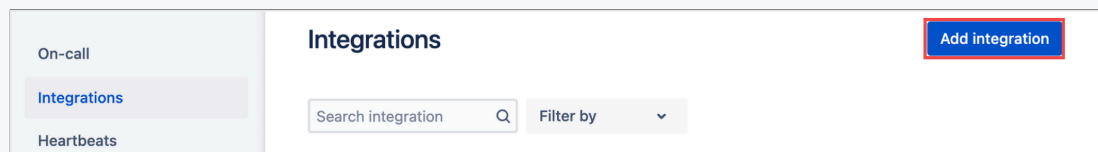
```
"metricName": "${metricName}",
"metricValue": "${metricValue}",
"metricThreshold": "${metricThreshold}",
"uuid": "${uuid}",
"okind": "${okind}",
"onode": "${onode}"
}
```

2. Enter values for the **Webhook Name** and **Webhook URL** fields.
3. To add a header value, click + **Add** in **Add user header value**, then enter the key and value.
4. Click the [**Registration**] button.

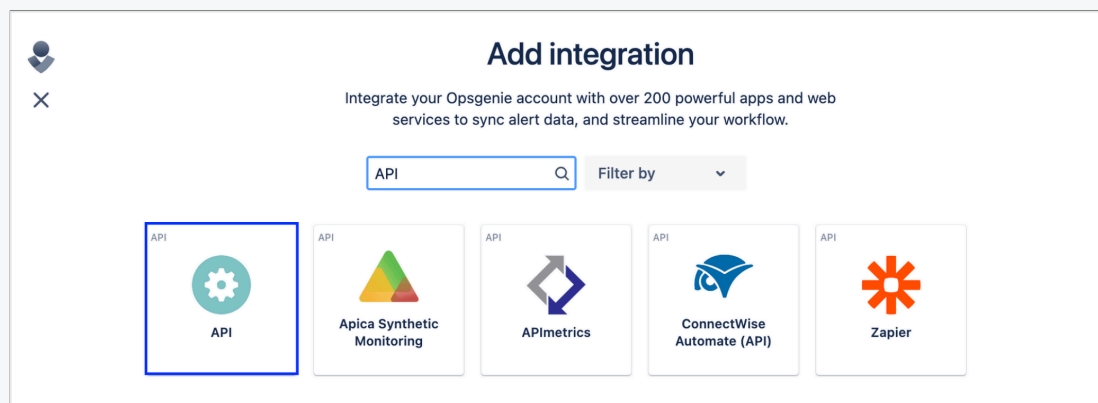
Opsgenie



1. In the **Teams** menu, select the team that will receive notifications.
2. Go to the **Integration** menu and click the [Add Integration] button.



3. In the integration list, search for and select **API**.





4. Set the API name, assign a team, configure the required permissions, and then click the [Save Integration] button.

5. Return to the **Add a third-party plug-in** window in the WhaTap monitoring service, and enter values for **Name** and **API Key** at the bottom of the **Opsgenie** tab.
6. Click the [**Registration**] button.

PagerDuty



1. Sign up for a [PagerDuty](#) account and log in.
2. In PagerDuty, go to **Services > Service Directory**, then click the [+ New Service] button.

PagerDuty

3. Enter the service information (**Name**) and click the [Next] button.

PagerDuty

4. Configure the **Escalation Policy**, then click the [Next] button.



PagerDuty

5. Configure **Alert Grouping** and **Transient Alerts**, then click the [Next] button.

PagerDuty

6. In **Integrations**, select **Event API V2**, then click the [Create Service] button.

PagerDuty

7. On the created service page, copy the **Integration Name** and **Integration Key** values.

PagerDuty

8. Return to the **Add a third-party plug-in** screen in the WhaTap monitoring service and enter the **Integration Name** and **Integration Key** values.
9. Click the [Register] button.

ⓘ If you do not receive messages or the plugin is not registered, check whether the **Integration Name** or **Integration Key** values are correct.

LINE



1. Install the LINE app.
2. Scan the QR code below to add the official **WhaTapLabs** channel as a friend.

Add LINE 3rd-party plugin

3. Use the following commands to link your project with LINE and configure notifications.

Commands

```
/help # Show available commands
/list # View the list of linked projects
/license [ACCESS_KEY] # Add a new project linkage
/remove [PROJECT_CODE] # Remove a linked project
/language [ko|ja|en] # Set notification language (Korean/Japanese/English)
```



- You can find the project access key under **Management > Project Management**.
- Using the `/language` command in LINE applies **only to the current LINE chat room**. It does not change the notification language for other 3rd-party plugins or projects.

Examples

```
/license XXXXXXXXXX # Link a project
/language ko # Set language to Korean
/list # Check linked projects
```



- **Project access key:** Available under **Management > Project Management** in the WhaTap project.
- **Language settings:** The `/language` command applies only within the LINE chat room and does not affect other notification channels.

ilert



1. In the ilert console, click [+ Create alert source] at the top right of the **Alert sources** page.
2. In the **Integration type** step, enter `WhaTap` in the **Filter integrations** search field at the bottom to search.
3. From the search results, select the **WhaTap** tile and click the [Next] button.
4. Enter the alert source name and assign a team, then click the [Next] button.
 - You can create a new escalation policy or select an existing one.
5. Select the alert grouping method and click the [Continue setup] button.
 - The initially selected **Do not group alerts** option can be changed later.
6. On the next screen, you can configure additional settings such as **alert templates and priorities**. For now, click the [Finish setup] button to complete the setup.
7. When the setup is complete, an **API key** or **Webhook URL** will be generated on the completion screen.



8. Copy the generated **Webhook URL** from the **Webhook JSON** tab and paste it into the **Webhook URL** input field below.
9. Click the [Register] button.

Preventing Alert Storms

If a large number of alerts occur within a short period, alert notifications are temporarily paused for a specified duration. For example, if 20 events occur within 5 minutes, alert notifications will be suspended for the next 5 minutes. Once the configured **Pause Duration** has elapsed, the alert-storm prevention feature is automatically released.

1. Go to **Alert > Notifications > the Avoids a large number of alerts** section.
2. Click the toggle button to **Enable** or **Disable** the feature.
 -  **Enabled:** Temporarily pause alerts when an alert storm is detected
 -  **Disabled:** Resume alert notifications
3. Configure the **Detection Time**, **Detection Count**, and **Pause Duration**.
 - If the number of events exceeds the **Detection Count** within the **Detection Time**, alert notifications are paused for the configured **Pause Duration**.



When alerts are blocked due to an alert storm, a notification banner is displayed. To immediately release the alert-storm block, click the **Stop Alert Storm Prevention** button in the banner.

Event Notification Format

Common Alert Template

The common template can be applied in most cases. It allows you to receive alerts in a consistent format across various environments.

- Product Type: Application, Database, Kubernetes
- Event Type: Application Alerts, Database Alerts, Metrics Alerts
- Event Channel: SMS, Mobile, 3rd-party plugins, Plugins

Event Title

Event title format

```
[Level][Platform][ProjectName][ApplicationName][EventTitle]
```

Event title example

```
[Info][JAVA][Application Project][TC-0-1-8081][CRITICAL_HIGH_MEMORY]
```

❗ If ApplicationName is not set, it is omitted from the event title.

- Platform is displayed as one of the following.
 - JAVA
 - NODEJS
 - PYTHON
 - PHP
 - DOTNET
 - GO
 - POSTGRESQL



- ORACLE
- MYSQL
- MSSQL
- BSM_JAVA
- CLOUDWATCH
- TIBERO
- KUBERNETES
- KUBE_NS
- URLCHECK
- URLCHECK_ADMIN
- CUBRID
- ALTIBASE
- CLUSTER
- REDIS
- MONGODB
- VR
- RUM

Event Message

This section describes the information that can be included in the event message. If `optional` is set to `false`, the item is always included in the message. If `optional` is set to `true`, the item is displayed only when the data is available.

Event message example

```
Project Name : Application Project
Project Code : 3
Agent Name : TC-0-1-8081
Message : RECOVERED: Memory is too high. less than 10%
```

Event Time : 2022-04-12 18:53:24 +0900
 Event OFF Time : 2022-04-12 18:53:24 +0900
 Alert Type : APPLICATION_MEMORY
 Metric Name : memory
 Metric Value : 20
 Metric Threshold : 10
 Stateful : true

Table | Event Message Components

| En | Ko | Supported Alert Types | Description |
|------------------|-----------------------|-----------------------|--|
| Project Name | Project name | All | - |
| Project Code | Project code | All | - |
| Agent Name | Agent name | All (optional) | oname |
| Message | Event message | All | - |
| Alert Type | Alert type | All | See the AlertType table below |
| Event Time | Event occurrence time | All | In 2022-04-13 10:40:49 +0900, +0900 indicates the GMT offset |
| Event Off Time | Event recovery time | All (optional) | In 2022-04-13 10:40:49 +0900, +0900 indicates the GMT offset |
| Metric Name | Metric name | All (optional) | Name of the metric used to evaluate the event condition |
| Metric Value | Metric value | All (optional) | Included when the metric value exceeds the threshold |
| Metric Threshold | Metric threshold | All | Threshold value used to determine the event |

| En | Ko | Supported Alert Types | Description |
|-----------------|------------------------------|--------------------------|---|
| | | (optional) | condition |
| Stateful | Recovered-event notification | All (optional) | <code>true</code> if recovered-event notification is enabled; otherwise <code>false</code> |
| Event Rule | Event trigger condition | Metrics alerts | - |
| Target Filter | Event target filter | Metrics alerts | Evaluates event conditions only for metrics collected from specific targets |
| Repeat Count | Event repeat count | Metrics alerts | The event is triggered when the condition is met the specified number of times within the repeat duration |
| Repeat Duration | Event repeat duration | Metrics alerts | Time window used to evaluate the repeat count |
| Receiver | Receiver | Metrics alerts | - |
| Query | MXQL query | Composite metrics alerts | - |
| Rule | Event trigger condition | Composite metrics alerts | - |
| Query Period | Query period | Composite metrics alerts | - |
| Query Interval | Query interval | Composite | - |

| En | Ko | Supported Alert Types | Description |
|-----------------|-------------------|--------------------------|--|
| | | metrics alerts | |
| Silent Time | Silent time | Composite metrics alerts | - |
| Query | URL | Exception alerts | URL of the request that caused the exception |
| TXID | Transaction ID | Exception alerts | - |
| Class | Error class name | Exception alerts | - |
| Log Message | Log message | Server – file log alerts | - |
| Log File | Log file path | Server – file log alerts | - |
| IP | IP address | All server alerts | - |
| CPU | CPU usage | All server alerts | Snapshot at the time the event occurred |
| CPU_load1 | CPU load (1 min) | All server alerts | Snapshot at the time the event occurred |
| CPU_loadPerCore | CPU load per core | All server | Snapshot at the time the event occurred |

| En | Ko | Supported Alert Types | Description |
|---|-----------------------------|-----------------------|---|
| | | alerts | |
| Memory | Memory usage | All server alerts | Snapshot at the time the event occurred |
| Swap | Swap usage | All server alerts | Snapshot at the time the event occurred |
| Disk Name Used Percent Free Size IO Percent | Disk performance | All server alerts | Snapshot at the time the event occurred |
| Name Bps Pps | Network traffic performance | All server alerts | Snapshot at the time the event occurred |
| Message Time Name | Acknowledgement information | All server alerts | - |

ⓘ This section displays as much information as possible that can be provided for the event.

- **AlertType** is displayed as one of the following values.

| AlertType | Description |
|--------------------------------|--------------------------------------|
| APPLICATION_CPU | Application CPU alert |
| APPLICATION_MEMORY | Application memory alert |
| APPLICATION_DISK | Application disk alert |
| APPLICATION_ACTIVE_TRANSACTION | Application active transaction alert |
| APPLICATION_ERROR_TRANSACTION | Application error transaction alert |



| AlertType | Description |
|------------------------------|--|
| APPLICATION_SLOW_TRANSACTION | Application slow transaction (response time) alert |
| METRICS | Metrics alert |
| COMPOSITE_METRICS | Composite metrics alert |
| ANOMALY | Anomaly detection alert |
| LOG_REALTIME | Real-time log alert |
| COMPOSITE_LOG | Composite log alert |
| SERVER_REBOOT | Server reboot alert |
| SERVER_NO_DATA | Server no-data alert |
| SERVER_PORT | Server port alert |
| SERVER_NETWORK_IOPS | Server network IOPS alert |
| SERVER_NETWORK_BPS | Server network BPS alert |
| SERVER_DISK_IO | Server disk I/O alert |
| SERVER_DISK_QUOTA | Server disk usage alert |
| SERVER_DISK_INODE | Server inode alert |
| SERVER_CPU | Server CPU alert |
| SERVER_MEMORY | Server memory alert |
| SERVER_CPU_STEAL | Server CPU steal alert |



| AlertType | Description |
|-----------------------|---------------------------------------|
| SERVER_MEMORY_SWAP | Server swap memory alert |
| SERVER_LOG_FILE | Server log file alert |
| SERVER_WINDOW_EVENT | Server Windows event alert |
| SERVER_OFF | Server alert OFF notification |
| SERVER_ACKNOWLEDGE | Server acknowledge notification |
| SERVER_PROCESS_COUNT | Server process count alert |
| SERVER_PROCESS_CPU | Server process CPU alert |
| SERVER_PROCESS_MEMORY | Server process memory alert |
| SERVER_PROCESS_OFF | Server process alert OFF notification |
| AGENT_ACTIVE | Agent activated alert |
| AGENT_INACTIVE | Agent deactivated alert |
| AGENT_REACTIVATED | Agent reactivated alert |
| URL | URL alert |
| TOO_MANY_EVENT | Too many events alert |
| CLOUD_WATCH | CloudWatch alert |
| EXCEPTION | Exception alert |

Event History And Monitoring

Home > Select Project >  **Alert** > **Event History**

Event History allows you to view events that have occurred in the project and analyze their detailed information. It supports Elasticsearch DSL queries, enabling you to find events that match specific conditions quickly and accurately.

Viewing Event History

Click the **event title** in the search results to move to a screen where you can analyze detailed data for the time period when the event occurred. This allows you to comprehensively understand the situation before and after the event.

1. Go to **Alert > Event History**.
2. In the **Event History** screen, use **Time Selection** to set the period for viewing event records.
3. Enter a query in the filter input field and click the [Magnifying Glass] button.
 - If no query is entered, all events are displayed.

Column Settings

You can choose which columns to display or hide.

1. In the **Event History** screen, click the [ Column Settings] button at the top left.
2. In the **Column Settings** window, select the columns you want to show or hide.
 - You can change the column order in **Displayed Items**.
 - You can search for specific columns using the **Column Name Search** field.

| Column Item | Description |
|-------------|------------------------|
| Event Name | The title of the event |

| Column Item | Description |
|---------------------|---|
| Message | The event message or snapshot information |
| Event Status | The current status of the event |
| Agent Name | The name (Oname) of the agent where the event occurred
- Displayed as blank if the event is not associated with a specific agent |
| Agent Group | The agent type (OkindName) where the event occurred
- Displayed as blank if the event is not associated with a specific agent |
| Agent Node | The server (OnodeName) of the agent where the event occurred
- Displayed as blank if the event is not associated with a specific agent |
| Event Occurred Time | The time when the event occurred |
| Event Resolved Time | The time when the event was resolved
- Displayed as - for events without a resolved state |

3. After completing the settings, click the [Confirm] button to apply the column configuration.

↓ CSV Download

You can download the query results as a CSV file.

1. Go to **Alert > Event History** and click the [CSV] button.
2. The list of events that match the current filter conditions is saved as a CSV file.

View Only Events in Progress

You can view only events that are currently in progress.

1. Go to **Alert > Event History** and click the [View Only Events in Progress] button.
2. The following condition is automatically added to the filter query and applied to the search.

Stateful: **true** and Status : **"ON"**

Event Search

- **Precise Search:** Specify exact conditions to retrieve only the events you need
- **Composite Conditions:** Search across multiple fields at once to handle complex conditions in a single query
- **Flexible Patterns:** Supports various search methods such as wildcards and partial matches
- **High Performance:** Fast searches powered by indexed data

Searchable Fields

Field names and values are not case sensitive.

- **Field Name Tag:** Event types that support the field
- **Field Type:** The data type of the field
 - : Number
 - : String
 - : Boolean
 - : Date

Event Information Fields

| Field Name | Type | Description | Example Value |
|------------|--------|------------------------|--|
| Title | String | Event title | "Database Connection Error" |
| OffTitle | String | Event recovery title | "RECOVERED: Database Connection Error" |
| Message | String | Event message | "Connection timeout occurred" |
| OffMessage | String | Event recovery message | "RECOVERED: Connection timeout occurred" |

| Field Name | Type | Description | Example Value |
|-----------------|--------|-------------------------------------|---|
| Level | String | Current event level | Critical, Warning, Info |
| OriginLevel | String | Original event level | Critical, Warning, Info |
| Status | String | Event status | ON, OFF, CANCEL, ACKNOWLEDGE, MAINTENANCE, DISABLED |
| ActCount | Number | Number of occurrences | 2 |
| MetricName | String | Metric name | "memory" |
| MetricValue | String | Metric value | "85.5" |
| OffValue | String | Recovery value | "72.8" |
| MetricThreshold | String | Threshold value | "80" |
| alertType | String | Event type | "BASIC", "METRICS", "TRANSACTION", etc. |
| alertId | String | Unique identifier of the event rule | "zf3uojer0fv4v7" |

Type-specific Supported Fields

ⓘ These fields are supported only for specific event types. If the event type does not match, the value may be empty.

| Field Name | Type | Event Type | Description | Example Value |
|------------|--------|----------------|--------------------|---------------|
| eventRule | String | Basic, Metrics | Event trigger rule | "memory ≥ 80" |
| field | String | Real- | Log search | "content" |

| Field Name | Type | Event Type | Description | Example Value |
|-------------|--------|---------------|------------------|--|
| | | time Log | key | |
| keyword | String | Real-time Log | Log search value | "Error" |
| logCategory | String | Real-time Log | Log category | "AppLog" |
| logContent | String | Real-time Log | Log content | "00:00000:00009:2025/04/06 23:03:52.55 server Error: 1601, Severity: 17, State: 3\n00:00000:00009:2025/04/06 23:03:52.55 server There are not enough 'user connections' available to start a new process." |

Target Fields

| Field Name | Type | Description | Example Value |
|------------|--------|------------------------------|---------------------|
| Oid | Number | Agent unique identifier | -98765432 |
| Oname | String | Agent name | "web-server-01" |
| Okind | Number | Agent type unique identifier | 867318026 |
| OkindName | String | Agent type name | "web-server" |
| Onode | Number | Agent node unique identifier | 334634079 |
| OnodeName | String | Agent node name | "production-node-1" |

Unique Event Fields

| Field Name | Type | Description | Example Value |
|------------|--------|--------------------------------|---------------------------|
| Id | Number | Unique identifier of the event | 5768121 |
| UUID | String | Unique event identifier | "550e8400-e29b-41d4-a716" |

State / Flag Fields

| Field Name | Type | Description | Example Value |
|-------------|---------|----------------------------------|---------------|
| Stateful | Boolean | Whether the event is state based | true, false |
| Disabled | Boolean | Whether the event is disabled | true, false |
| Escalation | Boolean | Whether escalation is applied | true, false |
| SystemEvent | Boolean | Whether it is a system event | true, false |

Query Syntax by Search Type

! For the basic search syntax, refer to the [WhaTap log search query syntax](#) documentation.

1. Keyword Search

Events that contain the string "Database Connection"

```
title: "Database Connection"
```

2. Multiple Value Search

Retrieves events that include any one of the specified values.

```
title: Database Connection
```

Events whose current event level is Info or Warning

```
level: info warning
```

3. Pattern Search

You can search using patterns with wildcards.

Find events that occurred on specific agents using a name pattern

```
oname: web-*-prod
```

タイトルが Connection で終わるイベントを検索

```
title: *Connection
```

4. Complex Condition Search

You can combine multiple conditions to perform precise searches.

Events with a Critical level and a title containing Database

```
level: critical and title: Database
```

Warning or Critical events from a specific agent

```
oname: web-server-01 and level: Warning Critical
```

5. OR Condition Search

Retrieves events that satisfy **at least one** of the specified conditions.

Events where the title or message contains a specific keyword

```
title: Connection or message: Connection
```

6. Exclusion Conditions

You can exclude specific conditions from the search.

All events except those with an Info level

```
not level: info
```

Real-time Notifications

The **Alert status** feature displays pop-up notifications when events configured in **Event Configuration** occur. You can receive alerts immediately when an event is triggered, allowing you to quickly recognize issues and respond. You can also review notifications that have already occurred or were missed, helping you understand and resolve situations where real-time response was difficult.

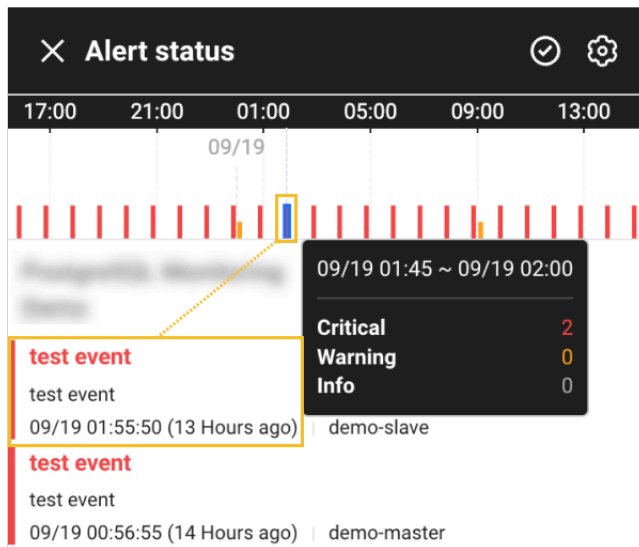
Viewing Real-time Notifications

Real-time notifications appear as pop-ups in the upper-right corner of the screen. You can check the time, title, and message of each notification.

- To close an individual notification, click the  button in the upper-right corner of the message.
- To close all pop-up notifications displayed on the screen, click **Close All Notifications**.

Viewing Notification History

1. Click the  icon at the top of the screen after selecting a project.
2. The **Alert status** window appears.
 - You can view up to 100 notifications from the past 24 hours for the current project.
 - Selecting an individual event item navigates you to the related menu.
 - In the time-based chart within the **Alert status** window, selecting a specific time range takes you to the corresponding event items.



- To view the full notification history, click **View All Notification History**. This navigates to **Alert > Event History**.

ⓘ If the **Alert status** list is empty, the **View All Notification History** button is displayed. Disable the **Receive notifications without reception tags** option.

Resolving Events in Bulk

You can resolve all ongoing events at once from the **Alert status** list. The bulk resolution process may take approximately 30 seconds to 1 minute to complete.

- When using the **Add Acknowledge** feature, a large number of recovery (resolved) notifications may be generated for currently ongoing events.
- Even after using the **Add Acknowledge** feature, events may occur again if the event rules remain enabled and the data still exceeds the threshold.
- Recovery notifications can be disabled by editing the event in **Alert > Event Configuration**. Disable **Send additional notification when the event is resolved** or **Recovered notification**.

How to use:

1. Click the  icon at the top of the screen after selecting a project.
2. When the **Real-time Notifications** window appears, click the  icon in the upper-right corner.
3. When the message `Resolve all ongoing events in the real-time notification list.` appears, click the **Resolve** button.

Configuring Real-time Notifications

1. Click the  icon at the top of the screen after selecting a project.
2. When the **Alert status** window appears, click the  icon in the upper-right corner.
3. Configure each option in the real-time notification settings window.
 - **Popup notifications:** Displays notifications as pop-ups even when the **Alert status** window is closed.
 - **Show resolved events:** Allows resolved events to be displayed in the **Alert status** list.
 - **Preview notification sound:** Lets you preview browser notification sounds. Click the  button next to **Critical** or **Warning**. If you cannot hear the sound, check your browser settings.

System Events

WhaTap Monitoring provides common system events that are automatically detected and generated across the platform. This document explains the types of events generated by the system, their trigger conditions, and recommended response actions.

EVENT_BLOCKED_BY_SYSTEM

If an excessive number of events occurs within a short period of time in a project, the WhaTap system detects this situation and blocks further events. When events are blocked, the system delivers the following message to the user. If you encounter this event, review and adjust your event configuration settings.

 **Critical**



Event Message

Excessive events detected, event blocked for stability. name: server agent

| | |
|---------------------|--------------------------------|
| Project No | 8 |
| Project Name | new server |
| Target | server agent |
| Event Title | EVENT_BLOCKED_BY_SYSTEM |
| Event Time | (GMT+0900) 2024-11-18 17:58:58 |

Log In to WhaTap

Events are blocked as a result of excessive events in the server agent of new server project. If no events occur in server agent for a period of time, it is automatically unblocked.

Blocking Criteria

Events are blocked based on the following criteria:

1. Agent-level blocking

If **10 events** occur within **1 minute** from a single agent, all subsequent events from that agent are blocked.

2. Project-wide blocking triggered by multiple agents

If **3 or more agents** are blocked within **1 minute** according to condition 1, events from the entire project are blocked.

3. Project-level blocking

If **100 events** occur within **1 minute** in a single project, all events from that project are blocked.

How to Release the Block

When events are blocked by the system, no events must occur from the affected agent or project for **1 minute**.

- Disable all configured events from **Alert > Event Configuration**. **Review your event settings and adjust thresholds to prevent excessive event generation.**
- Before configuring events, use the **Simulation** feature to estimate how many events may occur.

Laboratory

This menu provides new or experimental features to users. Through  **Laboratory** or **Sitemap** in the side menu, you can access it. The provided features may differ depending on the product.

 The functionality of the **Laboratory** menu is currently in Beta version and may cause unexpected errors. Be careful when using the **Laboratory** menu. It is not recommended to use the menu in handling important data or in the operating environment. If you have any feedback or issues, feel free to contact the support team at support@whatap.io.

- **MXQL Data**

MXQL is a query language to flexibly query WhaTap's performance data (metrics). It is used to comprehensively search and use the metrics collected from multiple agents in a project. For more information about **MXQL**, see [the following](#).

- **Pending Pod Status**

It provides useful features for checking the status and error messages of Pods and reviewing the cluster resource status, network policy, scheduler log, and such.

 This function is supported in Kubernetes master agent 1.7.0 or later.

- **Alert Message**

It provides a customization function to allow to modify the alert messages that are delivered to project members via email or third-party plugins. For more information, see [the following](#).

 In the side menu, if the  **Laboratory** menu does not appear, you can access it through **Sitemap**.
Under the side menu, select . If the **Sitemap** window appears, select a desired menu in the **Laboratory** section on the lower right of the window.