WhaTap

Log Monitoring

release date. 2026. 09. 10.

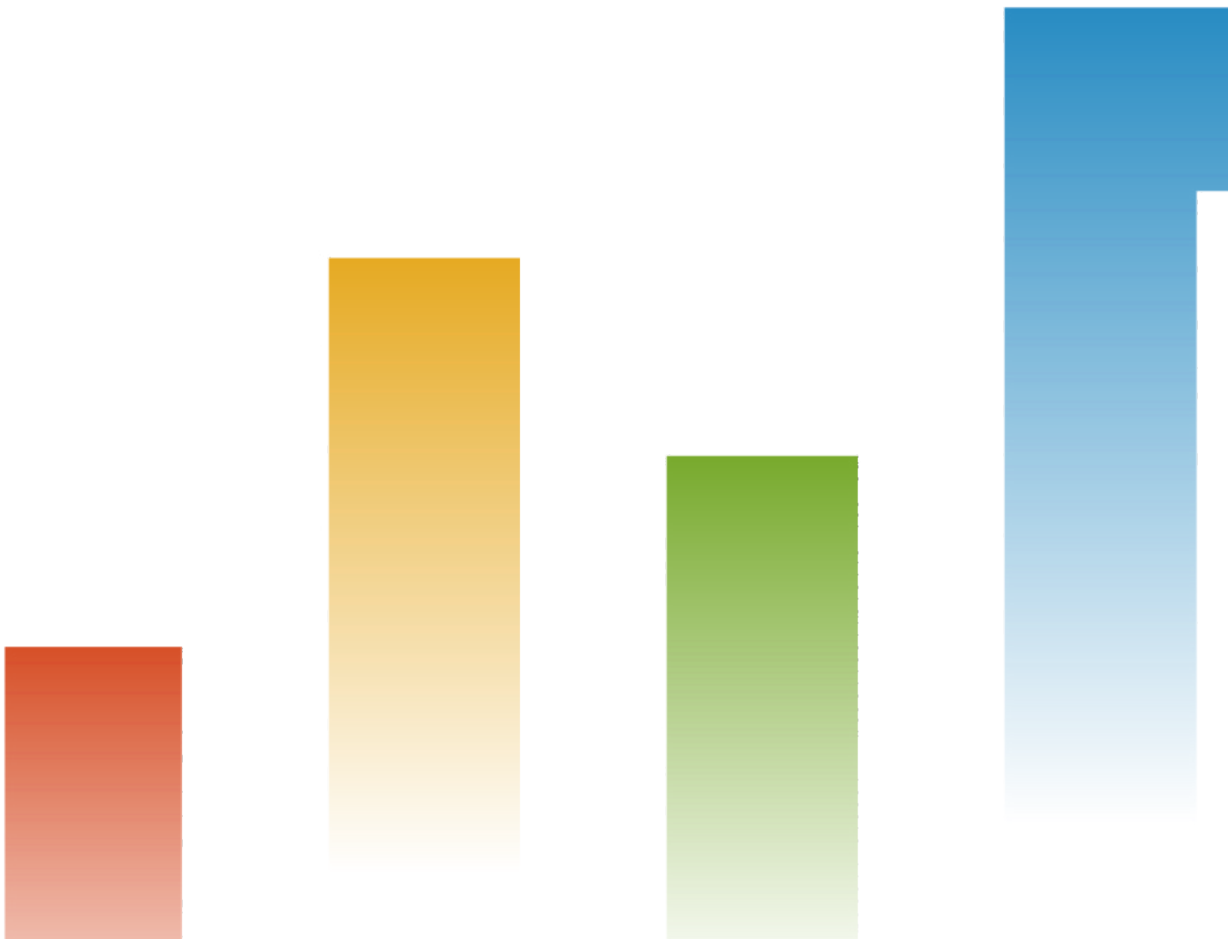


Table of Contents

Log 모니터링

로그는 애플리케이션 및 시스템에서 발생하는 이벤트와 메시지 등을 기록한 파일입니다. 이상 징후를 파악해 시스템 악화를 막거나 발생한 장애의 원인을 이해하고자 한다면 로그를 확인하는 것이 중요합니다.

최근 IT 서비스 구축 환경은 MSA(Microservice Architecture) 또는 Kubernetes 기반으로 전환되면서 관리 대상이 증가하고 있습니다. 일반적인 로그 모니터링은 서버에 접속해 `tail` 명령어나 편집기를 통해 확인합니다. 하지만 개별 서버에 일일이 접속하는 등의 단순한 방법을 사용하기 어렵습니다. 경우에 따라 수백 또는 수천 대의 서버에서 발생하는 로그를 어떻게 확인할 수 있을까요?

와탭 로그 모니터링 서비스를 통해 수많은 로그를 쉽게 관리할 수 있습니다.

> ⓘ 리눅스 `tail` 명령어는 시간에 따라 내용이 추가되는 로그 등을 확인하기 위한 용도로 많이 사용됩니다.

주요 특장점

- **중앙 통합 관리**

  와탭은 대량의 로그를 중앙에서 통합 관리합니다. 개별 서버에 접근하지 않아도 중앙에서 로그를 확인할 수 있어 편리합니다.

- **모든 로그 수집**

  로그를 선별해서 수집하는 경우 중요한 데이터가 누락될 수 있습니다. 와탭은 모든 로그를 수집하며, 수집된 로그들은 **라이브 테일**에서 실시간으로 확인할 수 있습니다.

- **가시성 확보**

  와탭의 다양한 차트에서 로그를 확인할 수 있습니다. 이를 통해 에러 및 이슈 정보에 대한 접근성을 높이고 장애 상황을 조기에 감지할 수 있게 합니다.

- **유연한 용량 관리**

  개별 서버에 로그 적재 시 로그로 인해 파일 시스템 용량이 과도하게 점유되는 문제가 발생할 수 있습니다. 와탭을 통해 로그를 중앙에 모은다면, 개별 서버에서 발생하는 로그 파일을 유지할 필요없이 중앙에 적재된 로그 데이터의 유지 기간만 관리하면 됩니다.

- **다양한 분석 관점**

장애 상황을 파악하고 예측하기 위해 다양한 관점으로 로그를 분석할 수 있어야 합니다. 와탭은 특정 태그를 포함한 로그 건수 추이를 확인하거나 필터링하여 분석할 수 있으며, 자주 사용하는 패턴은 차트로 저장해 언제든지 조회할 수 있습니다.

- **패턴 알림**

  장애 패턴을 알림으로 설정하여 문제를 예방하거나 빠르게 인지할 수 있습니다. 와탭은 개별 로그를 기준으로 특정 키워드가 포함되면 알림을 받는 **실시간 로그 알림**과, 특정 태그가 포함된 로그 건수 추이를 기준으로 알림을 받는 **복합 로그 알림**을 제공합니다.

로그 분석

라이브 테일

라이브 테일 메뉴에서 실시간으로 수집된 로그를 `tail` 명령어처럼 화면을 통해 흘러가는 로그를 조회할 수 있습니다.

> ⓘ 자세한 내용은 라이브 테일 문서를 참고하세요.

로그 트렌드

로그 트렌드 메뉴에서 수집되는 전체 로그 또는 특정 태그가 포함된 로그의 건수 추이를 확인할 수 있습니다. 로그 발생 건수는 장애 발생 및 해소 시점과 밀접하게 연관되므로, 로그 발생 건수 추이를 통해 장애 원인을 분석하고 대응 속도를 높일 수 있습니다.

> ⓘ 자세한 내용은 로그 트렌드 문서를 참고하세요.

로그 검색

로그 검색 메뉴에서 수집되는 전체 로그 또는 특정 태그가 포함된 로그를 조회할 수 있습니다. 특정 시간대나 서버에서 발생한 로그를 태그를 기준으로 조회하고 확인할 수 있습니다. 인접 로그 기능을 사용하면 특정 Error 또는 Exception 발생 전후의 로그를 함께 확인하여 원인 분석에 활용할 수 있습니다.

> ⓘ 자세한 내용은 로그 검색 문서를 참고하세요.

로그 탐색

로그 탐색 메뉴를 통해 로그 데이터를 빠르게 검색, 필터링, 분석할 수 있습니다. 필드 구조를 파악하고, 쿼리 기반 필터링을 통해 필요한 로그를 효율적으로 검색해, 결과를 데이터 시각화할 수 있습니다. 로그 탐색 기능은 Lucene 인덱스 기반으로 동작합니다.

> ⓘ 자세한 내용은 로그 탐색 문서를 참고하세요.

로그 모니터링 적용하기

와탭 로그 모니터링 서비스 이용을 위한 기본 적용 방법을 안내합니다. 와탭 로그 모니터링은 추가적인 에이전트를 구성하거나 로그에 맞추어 parser를 적용할 필요가 없습니다. 간단한 설정으로 빠르게 시작할 수 있습니다.

로그 모니터링의 작동 원리

일반적인 로그 통합 서비스는 수집기, 처리기, 저장소 그리고 UI 모듈로 이루어져 있습니다. 단계별 설정과 구성 작업을 필요로 하기에 각각의 모듈을 구축하는 과정이 번거로우며 추가 비용이 발생합니다.

와탭 로그 모니터링은 적용이 간단합니다. 기존의 모니터링 에이전트가 수집기 역할을 하기에 에이전트 옵션을 켜는 것만으로 로그 모니터링을 시작할 수 있습니다.

- Java PHP Python Go Server Kubernetes

- 출력된 파일에서 로그를 읽지 않고 Java 애플리케이션의 로그 라이브러리로 전달되는 로그를 직접 수집합니다.
- 로그를 직접 수집하기 때문에 파일 I/O를 유발하지 않아 시스템에 미치는 성능 영향이 매우 낮습니다.
- 트랜잭션 트레이스와 로그의 연결 추적성을 확보하여 트레이스에서 로그를 확인할 수 있습니다.

> **Java 로그 라이브러리**
>
> 대표적인 Java 로그 라이브러리는 Apache Log4j, Logback 입니다.

Java Agent 2.1.1 버전부터 사용할 수 있습니다.

- 기존의 에이전트에 로그 수집 기능을 추가했습니다. 모니터링 에이전트가 로그 파일에 추가로 출력된 로그를 읽어 수집하는 방식을 활용합니다.

PHP Agent 2.3.2 버전부터 사용할 수 있습니다.

- 기존의 에이전트에 로그 수집 기능을 추가했습니다. 모니터링 에이전트가 로그 파일에 추가로 출력된 로그를 읽어 수집하는 방식을 활용합니다.
- 로그에 트랜잭션 ID를 출력하면, 트랜잭션 트레이스와 로그의 연결 추적성을 확보하여 트레이스에서 로그를 확인할 수 있습니다.

Python Agent 1.2.2 버전부터 사용할 수 있습니다.

- 기존의 에이전트에 로그 수집 기능을 추가했습니다. 모니터링 에이전트가 로그 파일에 추가로 출력된 로그를 읽어 수집하는 방식을 활용합니다.
- 기존의 에이전트에 로그 수집 기능을 추가했습니다. 모니터링 에이전트가 로그 파일에 추가로 출력된 로그를 읽어 수집하는 방식을 활용합니다.

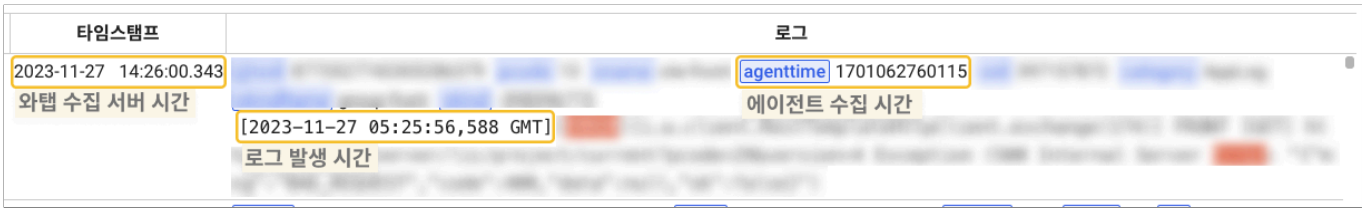
Server Agent 2.1.2 버전부터 사용할 수 있습니다.

- 쿠버네티스 컨테이너에 로그를 수집할 수 있습니다.

- 쿠버네티스 컨테이너 내부 애플리케이션의 로그를 수집할 수 있습니다.

> Kubernetes Agent 1.1.35 버전부터 사용할 수 있습니다.

로그 수집 시간 기준



로그 **수집 시간** 기준 시간이 **와탭 에이전트가 로그를 수집한 시간**에서 **와탭 수집 서버의 로그 처리 시간**으로 변경되었습니다.

일반적인 상황에서 변경 전과 큰 차이가 없으며, 기존 방식과 동일하게 로그를 검색할 수 있습니다. 다음과 같은 경우도 사용자의 추가적인 수정없이 로그 모니터링을 일관적으로 사용할 수 있습니다.

- NTP 사용 시, 모니터링 대상의 서버 시간이 표준 시간보다 과거 또는 미래 시간으로 설정되어있는 경우
- NTP 미사용 시, 2개 이상인 모니터링 대상의 서버 시간이 서로 다른 경우

타임셀렉터 조회 범위 지정

수집 시간 기준 시간 변경으로 타임셀렉터 역시 **와탭 수집 서버 시간**을 기준으로 동작합니다. 이에 따라 **에이전트 수집 시간**과 **와탭 수집 서버 시간**에 차이가 생겨 조회 범위에 포함되지 않는 로그가 발생할 수 있습니다. 이 경우 조회 범위를 넓게 지정 시 조회가 가능합니다.

로그 트렌드

로그 트렌드 추이 차트의 X축은 최소 1분 기준입니다. 로그 기준 시간이 변경된 후에도 차트를 통한 전체적인 추이 파악에 영향을 주지 않습니다.

로그 검색

로그 메시지에 로그 생성 시간을 나타내는 인덱스 `agenttime`이 추가되었습니다. `agenttime` 값을 통해 에이전트 수집 시간을

확인하세요.

> ⓘ UTC를 따르는 **와탭 수집 서버 시간**은 사용자 브라우저 시간에 따라 변환해 표기합니다.
>
> 예, 한국의 경우 UTC+9 기준으로 시간을 표기합니다.

단계별 기준 시간

로그는 3단계를 거쳐 수집됩니다. 단계마다 서로 다른 기준 시간이 사용될 수 있습니다.

와탭 에이전트 수집 시간과 **와탭 수집 서버 시간**은 큰 차이가 있습니다. 모니터링 대상의 서버 시간을 확인하세요. 모니터링 대상의 서버 시간이 과거 또는 미래 시간으로 설정된 경우 에이전트 수집 시간에 영향을 끼칩니다.

1. **로그 발생 시간**

   모니터링 대상의 시간 또는 Logging 정책에 따라 편차가 발생할 수 있습니다.

2. **에이전트 수집 시간**

   사용 중인 제품 또는 에이전트가 로그를 수집하는 방법 및 생성되는 로그에 따라 편차가 발생할 수 있습니다.

   - **Application**

     설정에 따라 로그 라이브러리 또는 로그 파일에서 실시간에 가깝게 로그를 수집합니다.

     > ⓘ 로그 라이브러리는 Java 제품만 지원합니다.

   - **Server**

     로그 파일에서 실시간에 가깝게 로그를 수집합니다.

   - **AWS Log**

     AWS Resource 정책에 따라 준 실시간 또는 수 분마다 로그를 수집합니다.

3. **와탭 수집 서버 시간**

   모니터링 대상 또는 로그 생성 방법과 상관없이 수집 서버에 저장되는 시간을 사용합니다.

로그 모니터링 적용하기

사용하는 애플리케이션에 따른 적용 방법은 다음과 같습니다. 로그 모니터링을 적용하기 전, 지원 버전을 먼저 확인하세요.

1. 지원하는 에이전트 버전을 확인하고 **업데이트**하세요.
2. 로그 모니터링 **옵션**을 설정하세요.
3. 로그 모니터링을 **활성화**하세요.

Java

Java 애플리케이션에서 로그를 수집하는 방법을 안내합니다.

PHP

PHP 애플리케이션에서 로그를 수집하는 방법을 안내합니다.

Node.js

Node.js 애플리케이션에서 로그를 수집하는 방법을 안내합니다.

Python

Python 애플리케이션에서 로그를 수집하는 방법을 안내합니다.

Go

Go 애플리케이션에서 로그를 수집하는 방법을 안내합니다.

Server

Server 애플리케이션에서 로그를 수집하는 방법을 안내합니다.

Kubernetes

쿠버네티스 컨테이너와 컨테이너 내부 애플리케이션의 로그를 수집하는 방법을 안내합니다.

Java

Java 애플리케이션에서 로그 수집을 위한 설정 방법을 안내합니다.

에이전트 업데이트

자바 에이전트 2.1.1 버전부터 가능합니다. 업데이트 방법은 다음 문서를 참고하세요.

에이전트 설정 확인

홈 화면 > **프로젝트 선택** > **로그** > **로그 설정**

1. 와탭 모니터링 서비스 초기 화면에서 프로젝트를 선택한 후, **로그** > **로그 설정** 메뉴로 이동하세요.
2. **로그 모니터링 시작하기** 탭의 **1. 에이전트 설정 확인**의 안내를 참고해 에이전트 설정을 확인하세요.

에이전트 설정하기

1. 프로젝트의 **관리** > **에이전트 설정** 메뉴를 클릭하세요.
2. **옵션 작성**의 **옵션 선택** 드롭 다운 목록에서 **직접 입력**을 클릭한 후, `logsink_enabled=true` 옵션을 작성하고 **적용** 버튼을 클릭해 추가하세요.
   - 키 입력창에 `logsink_enabled`을 작성하세요.
   - 값 입력창에 `true`을 작성하세요.
3. 로그 모니터링을 적용하기 위해 애플리케이션을 다시 시작하세요.

주요 옵션

- **hooklog_enabled** Boolean

  기본값 `false`

  Log 라이브러리를 후킹(Hooking)하여 로그 모니터링을 활성화합니다.

> 애플리케이션 실행 전, whatap.conf에 본 옵션이 활성화되어 있어야 이후 로그 모니터링의 On/Off를 `logsink_enabled` 설정을 통해 동적으로 제어할 수 있습니다. 애플리케이션 실행 전에 `logsink_enabled` 옵션이 `true` 로 설정된 경우 본 옵션을 별도로 설정하지 않아도 로그 모니터링이 가능합니다.
>
> 앞으로 로그 모니터링을 활용할 가능성이 있다면, 사전에 본 옵션을 꼭 설정할 것을 권장합니다.

- **hooklog_custom_methods**

  사용자 정의 로그를 등록합니다. 임의의 로그 프레임워크 내용을 전달합니다. 사이트에서 개별로 만든 로그 모듈의 로그를 추적할 때 사용합니다.

  Java

```java
package io.home.test;

public class MyLog {
  public void customLog(String log) { ... }
}
```

  whatap.conf

```
hooklog_custom_methods=io.home.test.MyLog.customLog
```

- **logsink_enabled** Boolean

  기본값 `false`

  Log 모니터링 기능을 On/Off 합니다.

> 애플리케이션 실행 전, whatap.conf에 `hooklog_enabled` 옵션이 설정되어 있으면 본 옵션을 통해 로그 모니터링의 On/Off를 동적으로 제어할 수 있습니다.

- **logsink_trace_enabled** Boolean

  기본값 `false`

  Log에 트랜잭션 ID를 삽입하여, 트랜잭션 트레이스의 로그 탭을 노출할지 여부를 지정합니다.

로그 모니터링 활성화

홈 화면 > **프로젝트 선택** > **로그** > **로그 설정**

로그 모니터링 시작하기 탭의 **2. 로그 모니터링 활성화**에서 와탭 로그 모니터링을 활성화 또는 비활성화 할 수 있습니다.

- 로그 모니터링 활성화: 토글 버튼을 켜면 로그 모니터링이 활성화되고, 활성화한 날부터 15일 동안 무료로 체험하실 수 있습니다.
- 로그 모니터링 비활성화: 토글 버튼을 끄면 로그 모니터링이 비활성화되고, 로그를 더 이상 저장하지 않습니다.

> **로그 모니터링 활성화 권한**
>
> 에이전트 설치 후 프로젝트에 대한 **수정 권한**이 있는 사용자만 로그 모니터링을 활성화할 수 있습니다. 권한에 대한 자세한 내용은 다음 문서를 참고하세요.

PHP

PHP 애플리케이션에서 로그 수집을 위한 설정 방법을 안내합니다.

에이전트 업데이트

PHP 에이전트 2.3.2 버전부터 가능합니다. 업데이트 방법은 다음 문서를 참조하세요.

에이전트 설정 확인

홈 화면 > **프로젝트 선택** > **로그** > **로그 설정**

1. 와탭 모니터링 서비스 초기 화면에서 프로젝트를 선택한 후, **로그** > **로그 설정** 메뉴로 이동하세요.
2. **로그 모니터링 시작하기** 탭의 **1. 에이전트 설정 확인**의 안내를 참고해 에이전트 설정을 확인하세요.

설정 파일 경로 확인

로그 모니터링을 원하는 파일의 **설정 파일 경로**를 확인하세요. CLI 환경에서 확인하거나 웹에서 확인할 수 있습니다.

CLI 환경에서 확인하기

```
$ php -i | grep 'Scan'
Scan this dir for additional .ini files => /etc/php8.0.d
```

> ⓘ 해당 경로가 "(None)"인 경우 설정 파일의 경로는 /usr/whatap/php 입니다.

웹에서 확인하기

CLI 환경과 Apache 환경의 PHP 설정이 다르면 웹에서 `phpinfo()` 함수의 결과를 확인하세요.

명령어 입력

설정 파일 경로를 포함한 명령어를 입력하면 로그 모니터링이 시작됩니다.

```
export LOGFILES=/some/path/file1,/some/other/file2
echo "whatap.logsink_enabled=true" | sudo tee -a [설정파일경로]/whatap.ini
echo "whatap.logsink.files=$LOGFILES" | sudo tee -a [설정파일경로]/whatap.ini
```

로그 모니터링 활성화

홈 화면 > 프로젝트 선택 > 로그 > 로그 설정

로그 모니터링 시작하기 탭의 **2. 로그 모니터링 활성화**에서 와탭 로그 모니터링을 활성화 또는 비활성화할 수 있습니다.

- 로그 모니터링 활성화: 토글 버튼을 켜면 로그 모니터링이 활성화되고, 활성화한 날부터 15일 동안 무료로 체험하실 수 있습니다.
- 로그 모니터링 비활성화: 토글 버튼을 끄면 로그 모니터링이 비활성화되고, 로그를 더 이상 저장하지 않습니다.

> **로그 모니터링 활성화 권한**
>
> 에이전트 설치 후 프로젝트에 대한 **수정 권한**이 있는 사용자만 로그 모니터링을 활성화할 수 있습니다. 권한에 대한 자세한 내용은 다음 문서를 참고하세요.

Node.js

Node.js 애플리케이션에서 로그 수집을 위한 설정 방법을 안내합니다.

에이전트 업데이트

Node.js 에이전트 0.5.1 버전부터 가능합니다. 업데이트 방법은 다음 문서를 참조하세요.

에이전트 설정 확인

홈 화면 > **프로젝트 선택** > **로그** > **로그 설정**

1. 와탭 모니터링 서비스 초기 화면에서 프로젝트를 선택한 후, **로그** > **로그 설정** 메뉴로 이동하세요.
2. **로그 모니터링 시작하기** 탭의 **1. 에이전트 설정 확인**의 안내를 참고해 에이전트 설정을 확인하세요.

에이전트 설정하기

1. 프로젝트의 **관리** > **에이전트 설정** 메뉴를 클릭하세요.
2. **옵션 작성**의 **옵션 선택** 드롭 다운 목록에서 **직접 입력**을 클릭한 후, `logsink_enabled=true` 옵션을 작성하고 **적용** 버튼을 클릭해 추가하세요.
   - 키 입력창에 `logsink_enabled` 을 작성하세요.
   - 값 입력창에 `true` 을 작성하세요.
3. 로그 모니터링을 적용하기 위해 애플리케이션을 다시 시작하세요.

주요 옵션

- **logsink_enabled** Boolean

  기본값 `false`

  로그 모니터링 활성화 여부를 설정합니다.

- **logsink_trace_txid_enabled** Boolean

  기본값 `true`

  로그에 트랜잭션 ID를 삽입하여 트랜잭션 트레이스의 로그 탭 노출 여부를 설정합니다.

- **logsink_limit_content_enabled** Boolean

  기본값 `true`

  로그 메시지의 최대 크기 제한 여부를 설정합니다.

- **logsink_limit_content_length** Number

  기본값 `10,000`

  로그 메시지의 최대 길이를 설정합니다. 이 값은 `logsink_limit_content_enabled` 옵션이 `true`로 설정된 경우에만 적용됩니다.

로그 모니터링 활성화

홈 화면 > **프로젝트 선택** > **로그** > **로그 설정**

로그 모니터링 시작하기 섹션의 **로그 모니터링 활성화** 탭에서 토글 버튼으로 와탭 로그 모니터링을 활성화 또는 비활성화할 수 있습니다.

- 로그 모니터링 활성화: 토글 버튼을 켜면 로그 모니터링이 활성화되고, 활성화한 날부터 15일 동안 무료로 체험하실 수 있습니다.
- 로그 모니터링 비활성화: 토글 버튼을 끄면 로그 모니터링이 비활성화되고, 로그를 더 이상 저장하지 않습니다.

> **로그 모니터링 활성화 권한**
>
> 에이전트 설치 후 프로젝트에 대한 **수정 권한**이 있는 사용자만 로그 모니터링을 활성화할 수 있습니다. 권한에 대한 자세한 내용은 다음 문서를 참고하세요.

Python

Python 애플리케이션에서 로그 수집을 위한 설정 방법을 안내합니다.

에이전트 업데이트

Python 에이전트 1.2.2 버전부터 가능합니다. 업데이트 방법은 다음 문서를 참조하세요.

에이전트 설정 확인

홈 화면 > **프로젝트 선택** > **로그** > **로그 설정**

1. 와탭 모니터링 서비스 초기 화면에서 프로젝트를 선택한 후, **로그** > **로그 설정** 메뉴로 이동하세요.
2. **로그 모니터링 시작하기** 탭의 **1. 에이전트 설정 확인**의 안내를 참고해 에이전트 설정을 확인하세요.

로그 수집 활성화

설정 파일 경로(WHATAP_HOME)를 포함한 명령어를 입력하면 로그 수집이 바로 시작됩니다.

```
export LOGFILES={로그파일전체경로},...
echo "logsink_enabled=true" | sudo tee -a {설정파일경로}/whatap.conf
echo "logsink.files=$LOGFILES" | sudo tee -a {설정파일경로}/whatap.conf
```

로그와 웹 트랜잭션 연동

1.3.6 이후 버전

1.3.6 버전부터 다음의 방법으로 트랜잭션과 로그 연동을 설정할 수 있습니다. 현재 와탭은 Python의 logging, loguru 라이브러리를 지원하고 있습니다. 사용하는 Python Log 라이브러리에 따라 whatap.conf를 구성하세요.

- logging 모듈

logging

```
echo "trace_logging_enabled=true" | sudo tee -a {설정파일경로}/whatap.conf
```

- loguru 모듈

loguru

```
echo "trace_loguru_enabled=true" | sudo tee -a {설정파일경로}/whatap.conf
```

1.3.6 미만 버전

트랜잭션 별로 발생한 로그를 별도로 조회할 수 있도록 트랜잭션 아이디 `{txid}` 를 로그에 출력합니다. 와탭 모니터링에서 Python LogRecord에 `{txid}` 를 자동 주입하여 포매터 설정 시 로그 파일에 `{txid}` 를 출력할 수 있도록 합니다.

```
settings.py
...
LOGGING = {
...
    'formatters': {
        ...
    },
    'handlers': {
        ...
    },
    'loggers': {
        ...
        '{로거이름}': {
            'handlers': [...],
        ...
        },
    },
}
```

```
try:
    import whatap.trace.mod.logging as whatap_logging
    if whatap_logging.logging_injection_processed:
        LOGGING['formatters']['whatap.formatter']={
            '()': 'django.utils.log.ServerFormatter',
            'format': '[{server_time}] -- {{ "@txid" : "{txid}" }} -- {message}',
            'style': '{',
        }
        LOGGING['handlers']['whatap']={
            'level': 'DEBUG',
            'class': 'logging.handlers.RotatingFileHandler',
            'filename': os.path.join(BASE_DIR, 'logs','whatap_log.log'),
            'formatter': 'whatap.formatter',
        }
        LOGGING['loggers'][{로거이름}]['handlers'].append('whatap')
except:
    pass
...
```

로그 모니터링 활성화

홈 화면 > **프로젝트 선택** > **로그** > **로그 설정**

로그 모니터링 시작하기 탭의 **2. 로그 모니터링 활성화**에서 와탭 로그 모니터링을 활성화 또는 비활성화할 수 있습니다.

- 로그 모니터링 활성화: 토글 버튼을 켜면 로그 모니터링이 활성화되고, 활성화한 날부터 15일 동안 무료로 체험하실 수 있습니다.
- 로그 모니터링 비활성화: 토글 버튼을 끄면 로그 모니터링이 비활성화되고, 로그를 더 이상 저장하지 않습니다.

> **로그 모니터링 활성화 권한**
>
> 에이전트 설치 후 프로젝트에 대한 **수정 권한**이 있는 사용자만 로그 모니터링을 활성화할 수 있습니다. 권한에 대한 자세한 내용은 다음 문서를 참고하세요.

Go

Go 애플리케이션에서 로그 수집을 위한 설정 방법을 안내합니다.

에이전트 설정 확인

홈 화면 > **프로젝트 선택** > **로그** > **로그 설정**

1. 와탭 모니터링 서비스 초기 화면에서 프로젝트를 선택한 후, **로그** > **로그 설정** 메뉴로 이동하세요.
2. **로그 모니터링 시작하기** 탭의 **1. 에이전트 설정 확인**의 안내를 참고해 에이전트 설정을 확인하세요.

로그 수집 활성화

아래 명령어를 입력하면 로그 수집이 바로 시작됩니다.

```
export LOGFILES=/some/path/file1,/some/other/file2
echo "logsink_enabled=true" | sudo tee -a /usr/whatap/agent/whatap.conf
echo "logsink.files=$LOGFILES" | sudo tee -a /usr/whatap/agent/whatap.conf
```

로그 모니터링 활성화

홈 화면 > **프로젝트 선택** > **로그** > **로그 설정**

로그 모니터링 시작하기 탭의 **2. 로그 모니터링 활성화**에서 와탭 로그 모니터링을 활성화 또는 비활성화할 수 있습니다.

- 로그 모니터링 활성화: 토글 버튼을 켜면 로그 모니터링이 활성화되고, 활성화한 날부터 15일 동안 무료로 체험하실 수 있습니다.
- 로그 모니터링 비활성화: 토글 버튼을 끄면 로그 모니터링이 비활성화되고, 로그를 더 이상 저장하지 않습니다.

> **로그 모니터링 활성화 권한**
>
> 에이전트 설치 후 프로젝트에 대한 **수정 권한**이 있는 사용자만 로그 모니터링을 활성화할 수 있습니다. 권한에 대한 자세한 내용은 다음 문서를 참고하세요.

Server

서버 애플리케이션에서 로그 수집을 위한 설정 방법을 안내합니다.

에이전트 업데이트

서버 에이전트 2.1.2 버전부터 가능합니다. 업데이트 방법은 다음 문서를 참고하세요.

에이전트 설정 확인

홈 화면 > **프로젝트 선택** > **로그** > **로그 설정**

1. 와탭 모니터링 서비스 초기 화면에서 프로젝트를 선택한 후, **로그** > **로그 설정** 메뉴로 이동하세요.
2. **로그 모니터링 시작하기**의 **1. 에이전트 설정 확인**의 **OS 선택** 탭에서 **리눅스(shell)** 또는 **윈도우(Powersehll, 관리자 권한)** 를 선택하세요.
3. **기본 설치** 또는 **카테고리와 함께 설치**를 참고해 에이전트 설정을 확인하세요.

> ⓘ 다음 설정은 에이전트 재시작이 필요합니다.

Linux Shell

- **기본 설치**

  Linux Shell

```
export LOGFILES=/some/path/file1,/some/other/file2
echo "logsink.files=$LOGFILES" | sudo tee -a /usr/whatap/infra/conf/whatap.conf
```

- **카테고리와 함께 설치**

  Linux Shell

```
cd /usr/whatap/infra
sudo mkdir extension

cat <<EOL | sudo tee extension/logsink.conf >> /dev/null
[[inputs.logsink]]
 category = "serverlog"
 ## 로그 발생량 통계 별도 데이터로 전송 여부
 stats_enabled = true
 ## 로그 발생량 통계 카테고리
 stats_category = "logsink_stats"
 ## 로그 파일 경로(path)에 별표(*)가 포함되어 제외할 로그 비대상 파일명 설정
 excludeNames = [ ".gz",".zip" ]
 [[inputs.logsink.file]]
   ## 로그 파일 지정 시, 날짜 패턴(strftime.org) 지정 가능
   path = "/some/path/%Y-%m-%d/.log"
   disabled = false
   encoding = "euc-kr"

 [[inputs.logsink.file]]
   path = "/some/other/log"
   disabled = false
   encoding = "utf-8"

 [[inputs.logsink.file]]
   ## 줄 단위 로그에서 해당 키워드 검색 시, 이전 로그에 병합
   nowrap_keywords = ["Caused by:", "Test"]

 [[inputs.logsink.file]]
 ## 프로젝트 코드 및 프로젝트 라이선스 입력 시, 로그를 같은 리전의 원하는 프로젝트로 전송 가능
 # pcode = 프로젝트 코드
 # license = "프로젝트 라이선스"

 [[inputs.logsink.file]]
 # whatap_host_ip = "수집 서버 아이피1/수집 서버 아이피2"
 # whatap_host_port = 6600

 [[inputs.logsink.file]]
 path = "/tmp/mixed.log"
 disabled = false
```

```
  encoding = "utf-8"
  ## [{"A":1,"B":2},{"C":3,"D":4},...] 형태를 각각의 로그로 분리할 수 있다.{"A":1,"B":2} {"C":3,"D":4}
  multiline_pattern = "simple_json_array"
EOL

sudo service whatap-infra restart
```

Windows Powershell

- **기본 설치**

  Windows Powershell

```
$LOGFILES="c:\\whatap\\logs\\%Y-%m-%d\\*.log,c:\\whatap\\logs\\*.log"
Add-Content "c:\Program Files\WhatapInfra\whatap.conf" -Value "logsink.files=$LOGFILES"
```

- **카테고리와 함께 설치**

  Windows Powershell

```
# 관리자 권한 필요
New-Item -type "Directory" -Path "C:\Program Files\WhatapInfra\extension"

$contentToAdd = @"
[[inputs.logsink]]
category = "serverlog"
## 로그 발생량 통계 별도 데이터로 전송 여부
stats_enabled = true
## 로그 발생량 통계 카테고리
stats_category = "logsink_stats"
## 로그 파일 경로(path)에 별표(*)가 포함되어 제외할 로그 비대상 파일명 설정
excludeNames = [ ".gz",".zip" ]
[[inputs.logsink.file]]
## 로그 파일 지정 시, 날짜 패턴(strftime.org) 지정 가능
path = "c:\\whatap\\logs%Y-%m-%d_.log"
disabled = false
encoding = "euc-kr"

```

```
[[inputs.logsink.file]]
## 줄 단위 로그에서 해당 키워드 검색 시, 이전 로그에 병합
nowrap_keywords = ["Caused by:", "Test"]

[[inputs.logsink.file]]
## 프로젝트 코드 및 프로젝트 라이선스 입력 시, 로그를 같은 리전의 원하는 프로젝트로 전송 가능
# pcode = 프로젝트 코드
# license = "프로젝트 라이선스"

[[inputs.logsink.file]]
# whatap_host_ip = "수집 서버 아이피1/수집 서버 아이피2"
# whatap_host_port = 6600

[[inputs.logsink.file]]
path = "/tmp/mixed.log"
disabled = false
encoding = "utf-8"
## [{"A":1,"B":2},{"C":3,"D":4},...] 형태를 각각의 로그로 분리할 수 있다.{"A":1,"B":2} {"C":3,"D":4}
multiline_pattern = "simple_json_array"

"@

New-Item -path "C:\Program Files\WhatapInfra\extension" -name "logsink.conf" -type "file" -value $contentToAdd
-Force

Restart-Service "Whatap Infra"
```

옵션 설정

- **stats_enabled** : 수집 현황 데이터의 수집 여부를 설정합니다. (기본값: false)
  - true 로 설정해야 하며, stats_category 에서 설정한 카테고리(logsink_stats)로 통계 데이터가 발생합니다.
- **stats_category** : 수집 현황 데이터를 저장할 메트릭스 카테고리를 설정합니다. logsink_stats 로 값을 설정해야 합니다.
  통계 데이터 필드는 다음과 같습니다.

  저장되는 통계 데이터 필드: file checkInterval encoding filepos checkedLocalTime lastupdatedLocalTime fileSize error firstCheck transferBytes

- `excludeNames` : 로그 파일 경로(path)에 별표(`*`)를 포함한 경우 로그 비대상 파일을 제외하도록 파일명을 설정할 수 있습니다. 쉼표(`,`)를 구분자로 이용해 복수 설정할 수 있습니다.

  Example

  ```
  excludeNames = [ ".gz",".zip" ]
  ```

- `nowrap_keywords` : 줄 단위 로그 검색 시 해당 옵션값으로 지정한 키워드가 검색될 경우 이전 로그에 병합합니다.
- `pcode` : 로그를 전송할 같은 리전의 프로젝트 코드를 입력합니다.
- `license` : 로그를 전송할 같은 리전의 프로젝트의 라이선스 코드를 입력합니다.

> ⓘ `pcode` , `license` 옵션의 경우 서버 에이전트 2.7.4 버전 이상부터 지원합니다.

윈도우 이벤트 로그 옵션 설정

윈도우 이벤트 로그 수집 시 다음과 같이 옵션을 설정할 수 있습니다.

```
# 관리자 권한 필요
New-Item -type "Directory" -Path "C:\Program Files\WhatapInfra\extension"

$contentToAdd = @"
[[inputs.win_eventlog]]
    category = "win_event_log"
    stats_category = "win_event_log_stats"
    stats_enabled = true
    enabled = true
    [[inputs.win_eventlog.file]]
        #true | false
        enabled = true
        # Application, Security, Setup, System, ForwardedEvents
        file = "Application"
        #1: Information, 2: Warning 3: Error 4: Audit Success 5 Audit Fail
        #event_type =
        #event id
        #event_id =
        #event source name
        #source_name = ""
```

```
        #프로젝트 코드 및 프로젝트 라이선스 입력 시, 로그를 같은 리전의 원하는 프로젝트로 전송 가능
        #pcode = 프로젝트 코드
        #license = "프로젝트 라이선스"
        #whatap_host_ip = "수집 서버 아이피1/수집 서버 아이피2"
        #whatap_host_port = 6600

"@

New-Item -path "C:\Program Files\WhatapInfra\extension" -name "win_eventlog.conf" -type "file" -value $contentToAdd -Force

Restart-Service "Whatap Infra"
```

- 카테고리 지정(`category`) 필수

  예, `win_event_log`

- 통계 카테고리 지정(`stats_category`) 필수

  예, `win_event_log_stats`

- 통계 카테고리 On/Off(`stats_enabled`) 필수

  예, `true` 혹은 `false`

- 수집 기능 On/Off(`enabled`) 필수

  예, `true` 혹은 `false`

- 파일별 수집 기능 On/Off(`enabled`) 필수

  예, `true` 혹은 `false`

- 파일(`file`) 필수

  예, `Application` , `Security` , `Setup` , `System` , `Forwarded`

- 이벤트 타입(`event_type`) 비필수

  예, `1` , `2` , `3` , `4` , `5`

> ⓘ **이벤트 타입**
>
> 1. Information
> 2. Warning
> 3. Error
> 4. Audit Success
> 5. Audit Fail

- 이벤트 아이디(`event_id`) 비필수
- 이벤트 소스 이름(`source_name`) 비필수
- 프로젝트 코드(`pcode`) 비필수
- 프로젝트 라이선스 (`license`) 비필수

> ⓘ **에이전트 지원 버전**
>
> - 윈도우 이벤트 로그 수집의 경우 서버 에이전트 2.5.2 버전 이상부터 지원합니다.
> - 윈도우 이벤트 로그 옵션 중 `pcode` , `license` 옵션의 경우 서버 에이전트 2.7.4 버전 이상부터 지원합니다.

다중 파일 지정 및 분리

다중 파일 지정

여러 로그 파일을 동일한 카테고리로 설정하려면, 각 로그 파일 경로를 `[[inputs.logsink.file]]` 항목에 추가하고, `path` 설정을 통해 파일 경로를 지정합니다. 또한, `category` , `stats_enabled` , `stats_category` 등의 옵션을 사용하여 로그 발생량 통계 설정과 같은 부가적인 설정을 할 수 있습니다.

/root/test1 및 /root/test2 파일을 동일한 카테고리(`serverlog`)로 지정한 예시입니다.

```
[[inputs.logsink]]
 category = "serverlog"
 stats_enabled = true
```

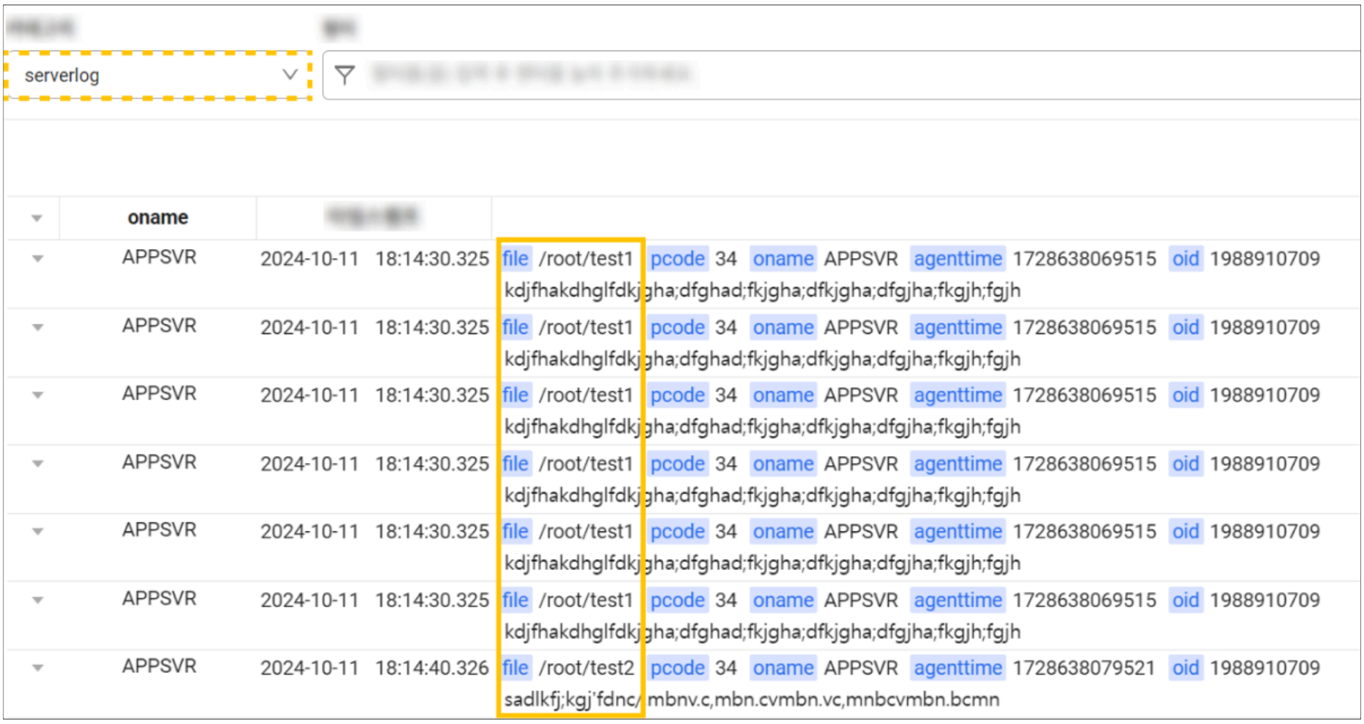
```
  stats_category = "logsink_stats"
  excludeNames = [ ".gz",".zip" ]

[[inputs.logsink.file]]
  path = "/root/test1"
  disabled = false
  encoding = "utf-8"

[[inputs.logsink.file]]
  path = "/root/test2"
  disabled = false
  encoding = "utf-8"
```

> ⓘ `category` , `stats_enabled` , `stats_category` , `nowrap_keywords` 등의 옵션에 대한 내용은 옵션 설정를 참고하세요.

해당 예시의 경우 **라이브 테일** 메뉴에서 다음과 같이 확인할 수 있습니다.



파일 분리

서로 다른 카테고리로 로그 파일을 지정하려면, logsink{숫자}.conf 파일을 각각 생성한 후, 각 파일의 `category` 항목에 서로 다른값을 설정하세요.

logsink.conf와 logsink**1**.conf를 통해 서로 다른 로그 카테고리를 설정한 예시입니다.

logsink.conf

```
# cat logsink.conf
[[inputs.logsink]]
 category = "serverlog1"
 stats_enabled = true
 stats_category = "logsink_stats"
 excludeNames = [ ".gz",".zip" ]

[[inputs.logsink.file]]
 path = "/root/test1"
 disabled = false
 encoding = "utf-8"
```

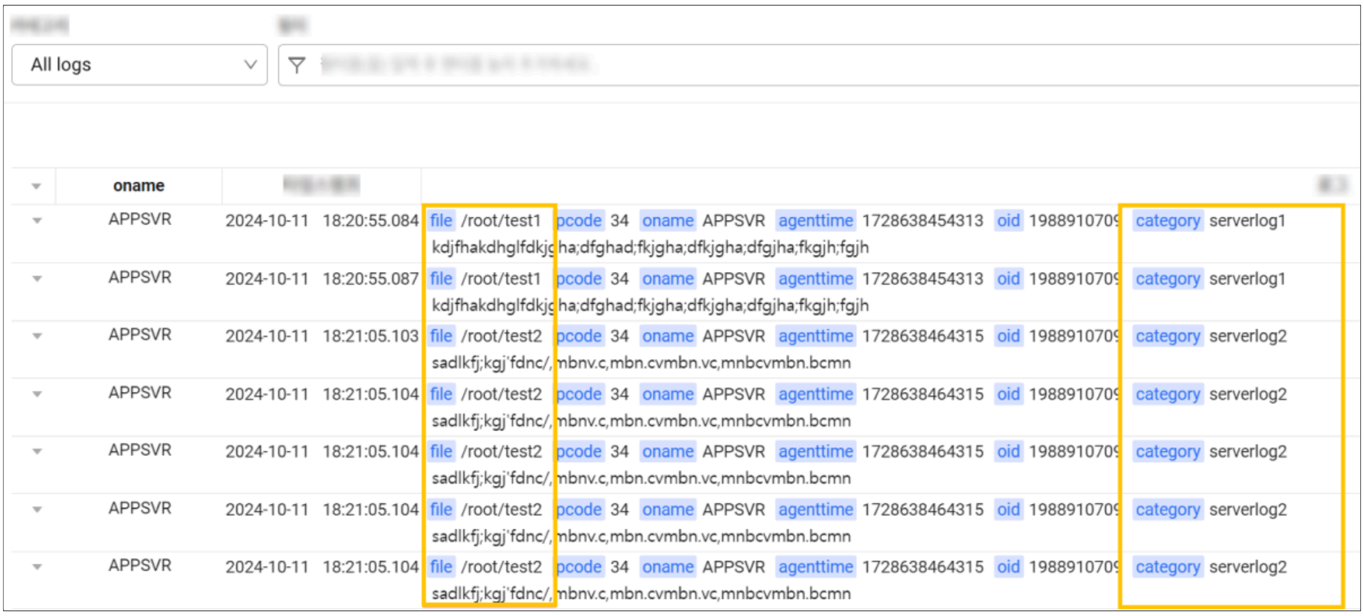
logsink1.conf

```
#cat logsink1.conf
[[inputs.logsink]]
 category = "serverlog2"
 stats_enabled = true
 stats_category = "logsink_stats"
 excludeNames = [ ".gz",".zip" ]

[[inputs.logsink.file]]
 path = "/root/test2"
 disabled = false
 encoding = "utf-8"
```

ⓘ `category` , `stats_enabled` , `stats_category` , `nowrap_keywords` 등의 옵션에 대한 내용은 옵션 설정를 참고하세요.

해당 예시의 경우 **라이브 테일** 메뉴에서 다음과 같이 확인할 수 있습니다.



로그 모니터링 활성화

홈 화면 > 프로젝트 선택 > **로그** > **로그 설정**

로그 모니터링 시작하기 섹션의 **로그 모니터링 활성화** 탭에서 토글 버튼으로 와탭 로그 모니터링을 활성화 또는 비활성화할 수 있습니다.

- 로그 모니터링 활성화: 토글 버튼을 켜면 로그 모니터링이 활성화되고, 활성화한 날부터 15일 동안 무료로 체험하실 수 있습니다.
- 로그 모니터링 비활성화: 토글 버튼을 끄면 로그 모니터링이 비활성화되고, 로그를 더 이상 저장하지 않습니다.

> **로그 모니터링 활성화 권한**
>
> 에이전트 설치 후 프로젝트에 대한 **수정 권한**이 있는 사용자만 로그 모니터링을 활성화할 수 있습니다. 권한에 대한 자세한 내용은 다음 문서를 참고하세요.

Kubernetes

쿠버네티스 컨테이너와 컨테이너 내부 애플리케이션의 로그 수집을 위한 설정 방법을 안내합니다.

에이전트 업데이트

쿠버네티스 에이전트 1.1.35 버전부터 가능합니다. 업데이트 방법은 다음 문서를 참조하세요.

에이전트 설정 확인

홈 화면 > **프로젝트 선택** > **로그** > **로그 설정**

1. 와탭 모니터링 서비스 초기 화면에서 프로젝트를 선택한 후, **로그** > **로그 설정** 메뉴로 이동하세요.
2. **로그 모니터링 시작하기** 탭에서 **에이전트 설정 및 로그 모니터링 활성화**의 **1. 에이전트 설정 확인**의 안내를 참고해 에이전트 설정을 확인하세요.

컨테이너 로그 수집 활성화

쿠버네티스 컨테이너에 로그를 수집하려면 **1. 에이전트 설정 확인**에서 **로그 설정 적용하기** 버튼을 클릭하세요.

컨테이너 내부 애플리케이션 로그 수집 활성화

Java 2.1.1, Python 1.2.2 버전부터 가능합니다. 쿠버네티스 컨테이너 상에서 실행되는 애플리케이션의 로그를 수집할 수 있도록 다음을 참고하세요.

1. **로그 > 로그 설정 > 로그 모니터링 시작하기** 탭으로 이동하세요.
2. **1. 에이전트 설정 확인** 하단의 **에이전트 설정** 버튼을 클릭하세요.
3. **에이전트 설정**에서 **옵션 작성**의 옵션 선택 드롭 다운 목록에서 **직접 입력**을 클릭하세요.
4. 에이전트 설정 명령어의 키 `logsink_enabled` 와 값 `true` 를 입력창에 입력하세요.

```
logsink_enabled=true
```

5. **적용** 버튼을 선택하세요. 쿠버네티스 컨테이너 내 애플리케이션의 로그를 수집할 수 있습니다.

> - Java 애플리케이션 로그 수집에 대한 자세한 내용은 다음 문서를 참고하세요.
> - Python 애플리케이션 로그 수집에 대한 자세한 내용은 다음 문서를 참고하세요.
> - Go 애플리케이션 로그 수집에 대한 자세한 내용은 다음 문서를 참고하세요.

로그 모니터링 활성화

홈 화면 > 프로젝트 선택 > **로그** > **로그 설정**

로그 모니터링 시작하기 섹션의 **로그 모니터링 활성화** 탭에서 토글 버튼으로 와탭 로그 모니터링을 활성화 또는 비활성화할 수 있습니다.

- 로그 모니터링 활성화: 토글 버튼을 켜면 로그 모니터링이 활성화되고, 활성화한 날부터 15일 동안 무료로 체험하실 수 있습니다.
- 로그 모니터링 비활성화: 토글 버튼을 끄면 로그 모니터링이 비활성화되고, 로그를 더 이상 저장하지 않습니다.

> **로그 모니터링 활성화 권한**
>
> 에이전트 설치 후 프로젝트에 대한 **수정 권한**이 있는 사용자만 로그 모니터링을 활성화할 수 있습니다. 권한에 대한 자세한 내용은 다음 문서를 참고하세요.

쿠버네티스 로그 모니터링 카테고리 안내

쿠버네티스 관련 다양한 로그를 확인할 수 있습니다. 와탭 쿠버네티스에서 모니터링 카테고리를 제공합니다.

> 설정에 따라 중복된 로그 내용이 저장될 수 있으니 반드시 중복 여부를 확인하시기 바랍니다.

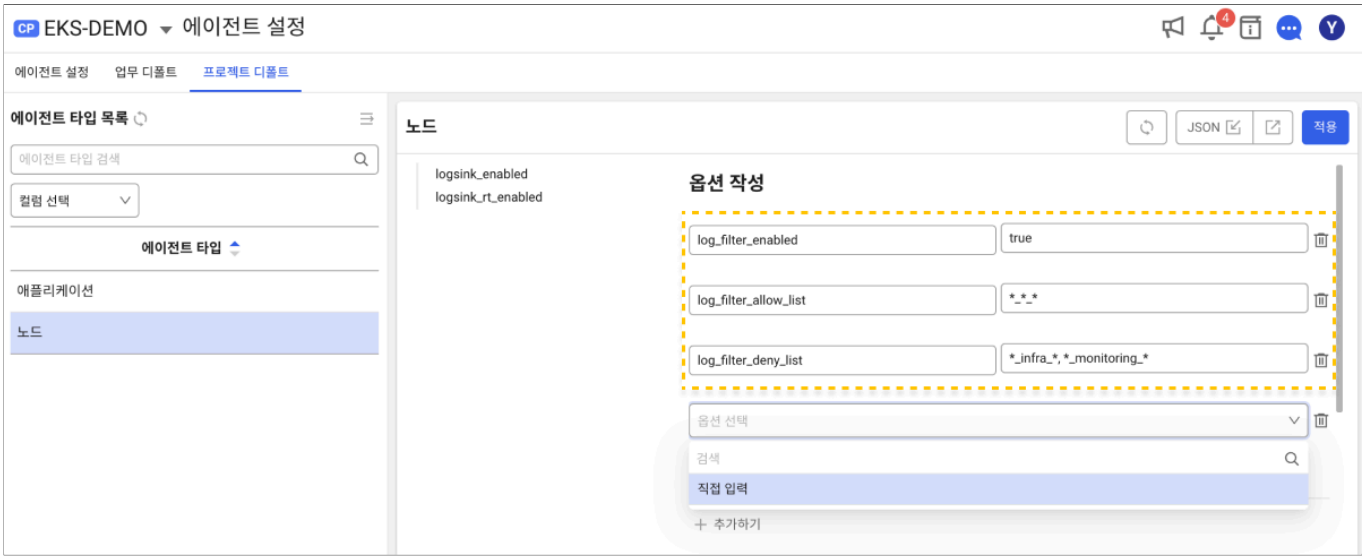
카테고리	설명
`#K8sEvent`	• 쿠버네티스에서 발생하는 이벤트가 저장된 로그 • 사용자 설정과 무관하게 기본 생성
`#WhatapEvent`	• 와탭 이벤트 설정에 의해 발생한 이벤트가 저장된 로그 • 사용자 설정과 무관하게 기본 생성
`containerStdout`	• 컨테이너 Standard Out 로그 • 사용자 설정 시 생성 • 노드 에이전트에 `logsink_enabled=true` 설정 추가 시
`AppLog`	• 컨테이너 내 애플리케이션 로그 • 사용자 설정 시 생성 • 애플리케이션 에이전트에 `logsink_enabled=true` 설정 추가 시

containerStdOut 로그 필터 설정

홈 화면 > 프로젝트 선택 > **관리** > **에이전트 설정**

쿠버네티스 로그에서 필요한 정보를 선별하기 위한 필터링 옵션을 안내합니다. 필터링을 통해 허용하거나 제외할 로그를 설정할 수 있습니다.

1. **관리** > **에이전트 설정**으로 이동하세요.
2. **프로젝트 디폴트** 탭에서 **옵션 작성** 방식으로 **직접 입력**을 클릭하세요.



로그 네이밍 규칙

쿠버네티스 containerStdOut 로그는 노드의 /var/log/containers/ 경로 하위에 생성됩니다. 다음과 같은 네이밍 규칙을 따르고 있습니다.

```
<podName>_<namespace>_<containerName>-<containerId>.log
```

로그 필터 옵션

- **log_filter_enabled** bool

  기본값 `false`

  로그 필터 사용 여부를 설정합니다.

- **log_filter_allow_list** list

  허용할 로그의 리스트를 설정합니다. 필터링 적용 시 포함할 항목을 의미합니다.

- **log_filter_deny_list** list

  제외할 로그의 리스트를 설정합니다. 필터링 적용 시 제외할 항목을 의미합니다.

로그 필터 동작

`log_filter_enabled` 옵션이 활성화된 경우만 로그 필터가 동작합니다. 해당 옵션의 값이 `true`로 설정된 상태에서 `log_filter_allow_list` 및 `log_filter_deny_list`를 통해 로그를 선택적으로 수집할 수 있습니다.

로그 필터는 블랙리스트 기반으로 작동합니다. 예를 들어 `log_filter_enabled` 활성화 후 `log_filter_allow_list` 옵션에 추가할 항목을 설정하지 않을 경우 기본적으로 모든 컨테이너의 로그 수집이 차단됩니다. 또한 로그 필터는 allow 규칙 보다 deny 규칙의 우선 순위가 높습니다. 동일한 항목이 allow와 deny에 모두 설정된 경우, deny 규칙이 우선 적용되어 해당 로그는 수집되지 않습니다.

로그 필터 예시

단일 네임스페이스 로그 수집

- 허용 **log_filter_allow_list** `*_infra_*`

`infra` 네임스페이스에 존재하는 모든 컨테이너가 수집 대상으로 특정됩니다.

다중 네임스페이스 로그 수집

- 허용 **log_filter_allow_list** `*_infra_*, *_monitoring_*`

`infra`, `monitoring` 네임스페이스에 존재하는 모든 컨테이너가 수집 대상으로 특정됩니다.

특정 단어를 포함하는 Pod 제외

- 허용 **log_filter_allow_list** `*_*_*`
- 제외 **log_filter_deny_list** `*prod*_*_*`

모든 컨테이너가 수집 대상으로 특정되지만, `podName`에 `prod`가 포함된 로그는 제외됩니다.

특정 단어를 포함하는 Pod 로그 수집

- 허용 **log_filter_allow_list** `*prod*_*_*`

`podName`에 `prod` 가 포함된 파드의 로그만 수집됩니다.

다중 네임스페이스 로그 수집 및 특정 단어 포함하는 Pod 제외

- 허용 **log_filter_allow_list** `*_infra_*, *_monitoring_*`
- 제외 **log_filter_deny_list** `*prod*_*_*`

`infra`, `monitoring` 네임스페이스의 로그가 수집되고, `podName`에 `prod`가 포함된 로그는 제외됩니다.

컨테이너 이름 내 특정 단어로 시작하는 컨테이너 로그 제외

- 허용 **log_filter_allow_list** `*_*_db-*`

모든 네임스페이스에서 `containerName`이 `db`로 시작하는 컨테이너의 로그는 제외됩니다.

라이브 테일

> ⓘ **로그 조회** 권한이 없을 경우 해당 메뉴에 진입할 수 없습니다.

홈 화면 > 프로젝트 선택 > **로그** > **라이브 테일**

라이브 테일 메뉴에서 서버 콘솔에 접속하지 않고, 로그 데이터 스트림을 모니터링 화면상에서 바로 확인할 수 있습니다. 복잡한 대량의 로그도 **필터**와 **하이라이트** 기능을 활용해 선별하고 빠르게 인지할 수 있습니다. 로그 데이터 조회 주기는 2초입니다.

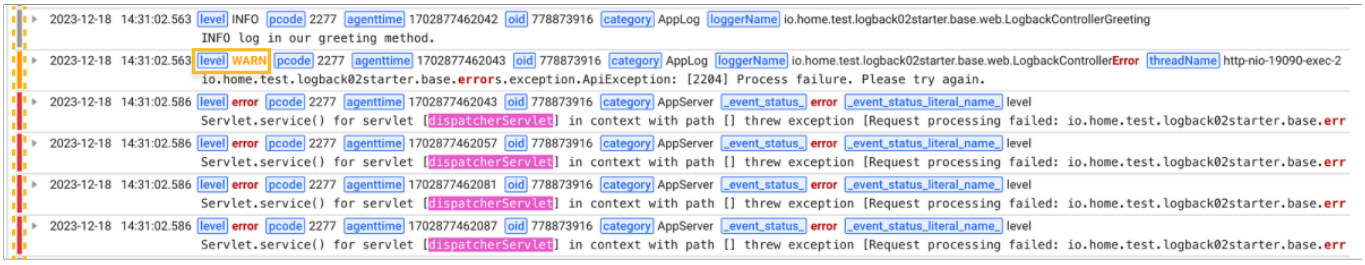
주요 용어는 다음과 같습니다.

- **Category**: 로그의 수집 및 조회 단위
- **Content**: 로그 메시지
- **Search Key**: 로그 파서 설정을 통해 생성되는 검색 키
- **Tag**: 수집된 로그를 검색할 수 있는 태그

> ⓘ 로그 테이블 컬럼 가장자리를 드래그해 컬럼 너비를 수정할 수 있습니다.

에이전트 옵션

에이전트 옵션이 설정된 경우 로그 레벨을 수집해 로그 레벨 기준 색상이 다음과 같이 표시됩니다.



에이전트 옵션 설정

```
# whatap.conf
weaving=log4j-2.17
weaving=logback-1.2.8
```

- Java 에이전트 2.2.22 버전 이후부터 위빙 설정에 log4j-2.17 또는 logback-1.2.8 설정 시 사용할 수 있습니다. 에이전트 재시작이 필요합니다.
- 로그 레벨은 파싱된 키워드 중 `level`, `type` 기준으로 판별합니다. `level`, `type`으로 파싱된 키가 존재하고 파싱 값이 FATAL, CRITICAL, ERROR, WARN, WARNING, INFO를 포함할 경우 로그 레벨 색상을 표시합니다.

필터 적용하기

필터를 적용하면 입력한 조건에 맞는 로그를 필터링합니다. 복수의 필터를 입력할 수 있습니다. 필터의 태그가 같은 경우 OR(`||`)로, 그렇지 않은 경우는 AND(`&&`)로 적용됩니다.

입력 창에 값을 직접 입력하거나 **필터** 입력 창을 클릭해 필터를 지정할 수 있습니다. 필터 태그는 `검색 키`, `연산자`, `검색 값`의 순서로 입력합니다. **검색** 버튼을 선택하면 필터가 적용된 데이터를 로그 목록에서 조회할 수 있습니다.

> **가이드 UI**
>
> 다음과 같이 입력 창 아래 **가이드 UI**를 제공합니다. 입력 창 또는 필터 태그에 마우스 커서가 있을 때 ESC 키를 통해 **가이드 UI**를 닫을 수 있습니다.

검색 키, 연산자, 검색 값 입력

- `검색 키` 입력 시 일반 인덱스, 예약어 인덱스, 숫자만 입력할 수 있는 인덱스를 구분해 추천 값을 제공합니다
- `연산자` 입력 시 일반 인덱스 검색 키의 경우 `==`, `!=` 옵션을 하단에 안내합니다. 숫자만 입력할 수 있는 인덱스의 경우 `>`, `<`, `<=`, `>=`, `==`, `!=` 옵션을 제공합니다.
- `검색 값` 입력 시 일치 검색(`>`, `<`, `<=`, `>=`, `==`)일 때 파란색으로, 제외 검색(`!=`)일 때 붉은색으로 하이라이팅합니다. 대소문자 구분 옵션을 활용해 검색할 수 있습니다.

- 필터 태그가 2줄 이상 길어지는 경우 ∧ **접기** 아이콘을 선택해 접어둘 수 있습니다.
- 필터 태그 입력 후 입력 창에서 Shift와 Tab 키를 동시에 사용하여 이전 필터 태그로 이동할 수 있습니다.
- 필터 태그 입력 후 입력 창에서 Tab 키를 통해 다음 필터 태그로 이동할 수 있습니다.

필터 태그 추가

입력 창에 텍스트 입력 후 Enter 또는 Tab 키를 누르거나, 입력창 하단의 가이드 UI에서 추천 값을 클릭하거나 방향키로 선택해 추가할 수 있습니다.

필터 태그 제거

Backspace 키 또는 태그의 **X** 아이콘을 눌러 태그를 삭제할 수 있습니다. 입력창 오른쪽의 **X** 아이콘을 클릭해 모든 태그를 한 번에 삭제할 수 있습니다.

필터 즐겨찾기

필터 **즐겨찾기** 기능을 제공합니다. 입력 창 오른쪽의 ☆ **즐겨찾기** 아이콘을 선택하여 원하는 필터 검색 조건을 즐겨찾기에 **추가**, **제거** 및 기존 즐겨찾기를 선택할 수 있습니다. **즐겨찾기**는 최대 50개까지 저장됩니다.

필터 적용 예외 상황

- 숫자만 입력할 수 있는 인덱스(`.n` 으로 끝나는 `검색 키`)를 입력한 태그에서 `검색 값` 은 숫자만 입력할 수 있습니다.
- 중복된 `검색 키` , `검색 값` 은 입력할 수 없습니다.
- `검색 키` , `검색 값` 중 하나라도 없는 태그가 존재할 때 검색할 수 없습니다. 유효하지 않는 태그의 경우 회색으로 표시합니다.

> ⓘ **라이브 테일** `검색 키` 로 `category` 를 입력할 수 없습니다.
> 입력된 필터 값 아래에 있는 수식(expression)은 로그 데이터 조회 시 적용될 필터 수식 미리 보기입니다.

미파싱 키워드 필터 적용

로그에서 파싱되지 않은 즉 인덱스가 생성되지 않은 키워드를 포함한 로그를 조회할 수 있습니다. 이 경우 지정 범위 내 모든 로그를 Full Scan합니다. 그렇기 때문에 인덱스가 생성된 키와 비교해 검색 속도가 다소 떨어질 수 있습니다. 정형화된 로그 데이터의 경우 **로그 파서 설정**을 통해 인덱스 키 값을 활용해 검색하는 것을 권장합니다.

1. **카테고리**를 선택하세요.
2. **필터** 입력창에 `content` 기준 띄어쓰기 후 검색을 원하는 키워드를 입력하세요.
   예시, `content *select*`
3. 🔍 **검색** 버튼을 클릭해 로그를 조회하세요.

> ⓘ **라이브 테일**의 경우 모든 로그 검색이 가능해 카테고리를 지정할 필요가 없습니다.
> 파서 설정에 대한 자세한 내용은 다음 문서를 참조하세요.

필터 수정

필터에 값을 입력한 뒤 입력한 값을 클릭하면 해당 값을 수정할 수 있습니다.

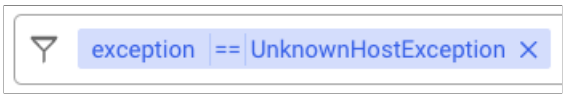
- 입력 창에 텍스트 재입력해 수정할 수 있습니다.
- 입력 창 아래 **가이드 UI**를 통해 추천 값을 선택해 수정할 수 있습니다.

검색 키(Search Key)

다음 이미지에서 파란색 박스 부분은 파싱(parsing)된 검색 키입니다. 검색 키는 **로그 설정**의 **로그 파서 설정** 탭에서 파싱 로직을 등록해 설정할 수 있습니다.

필터 입력 문법

태그는 검색 키와 검색 값으로 구성되어있습니다. 다음의 예시에서 검색 키는 `exception` , 검색 값은 `UnknownHostException` 입니다. 해당 예시는 수집한 로그 데이터 중 IP 주소와 도메인 주소가 매칭되지 않아 서버를 호스트에 연결할 수 없을 경우 발생하는 예외(`UnknownHostException`)가 포함된 로그 데이터를 조회합니다.



검색 키 종류

검색 키 종류	검색 키 포멧	의미	검색 키와 검색 값 예시	검색 예시
문자열 키워드	keyword	파일 이름	- 키: fileName - 값: /data/whatap/logs/yard.log	fileName:/data/whatap/logs/yard.log
숫자 키워드	keyword.n	응답시간	- 키: response_time.n - 값: 2945	response_time.n>=2945
예약어 키워드 (사전 정의 키워드)	@keyword	트랜잭션 ID	- 키: @txid - 값: 85459614215434144	-
로그 본문 키워드	content	로그 본문	- 키: content - 값: 사용자 입력값	content: `*ERROR*`

Content 검색 키

- Content 검색 키는 인덱싱되지 않은 로그의 본문을 대상으로 검색합니다. 예를 들어 `content *ERROR*` 와 같이 입력하는 경우 로그 본문 중 `ERROR` 를 포함한 로그를 검색합니다.
- 어떤 키워드로 인덱싱을 걸어야하는지 모르는 경우 Content 검색 키를 활용해 문제가 되는 키워드를 포함한 로그를 식별합니다. 이후 **로그 설정** 메뉴의 로그 파서 설정을 통해 해당 키워드로 파서를 설정해 인덱스를 생성하는 방식으로 검색 속도를 향상시킬 수 있습니다.

공통 문법

문법 종류	설명	예시
`==searchValue`	검색 값과 일치하는 로그를 검색합니다.	`exception==RuntimeExceptionexception`
`!=searchValue`	검색 값을 제외한 로그를 검색합니다.	`exception!=RuntimeException`

문법 종류	설명	예시
`*searchValue`	검색 값으로 끝나는 로그를 검색합니다.	`word==*hello`
`searchValue*`	검색 값으로 시작하는 로그를 검색합니다.	`word==hello*`
`*searchValue*`	검색 값으로 중간에 포함된 로그를 검색합니다.	`word==*hello*`
`*search*Value*`	검색 값으로 포함된 로그를 검색합니다.	`word==*he*llo*`
`re:{regex}`	정규표현식에 매칭되는 로그를 검색합니다.	`caller==re:^i\.w\.a\.w\.s\.v\.r\.`
`**`	검색 키에 해당하는 모든 로그를 검색합니다.	

검색 키가 숫자 키워드(keyword.n)인 경우 문법

다음의 문법은 검색 키가 `keyword.n` 형식인 경우에만 지원합니다.

- 검색 값으로는 숫자만 올 수 있습니다.
- `.n` 키워드의 값에는 prefix를 붙이지 않습니다. `.n` 이 아닌 키워드는 모두 prefix를 붙여야합니다.

  예, `+>searchValue` 는 유효하지 않습니다.

문법 종류	설명	예시
`>searchValue`	검색 값보다 큰 값이 포함된 로그를 조회합니다.	`response_time.n>3000`
`>=searchValue`	검색 값보다 크거나 같은 값이 포함된 로그를 조회합니다.	`response_time.n>=3000`
`==searchValue`	검색 값보다 같은 값이 포함된 로그를 조회합니다.	`response_time.n==3000`
`!=searchValue`	검색 값보다 다른 값이 포함된 로그를 조회합니다.	`response_time.n!=3000`
`<searchValue`	검색 값보다 작은 값이 포함된 로그를 조회합니다.	`response_time.n<3000`
`<=searchValue`	검색 값보다 작거나 같은 값이 포함된 로그를 조회합니다.	`response_time.n<=3000`

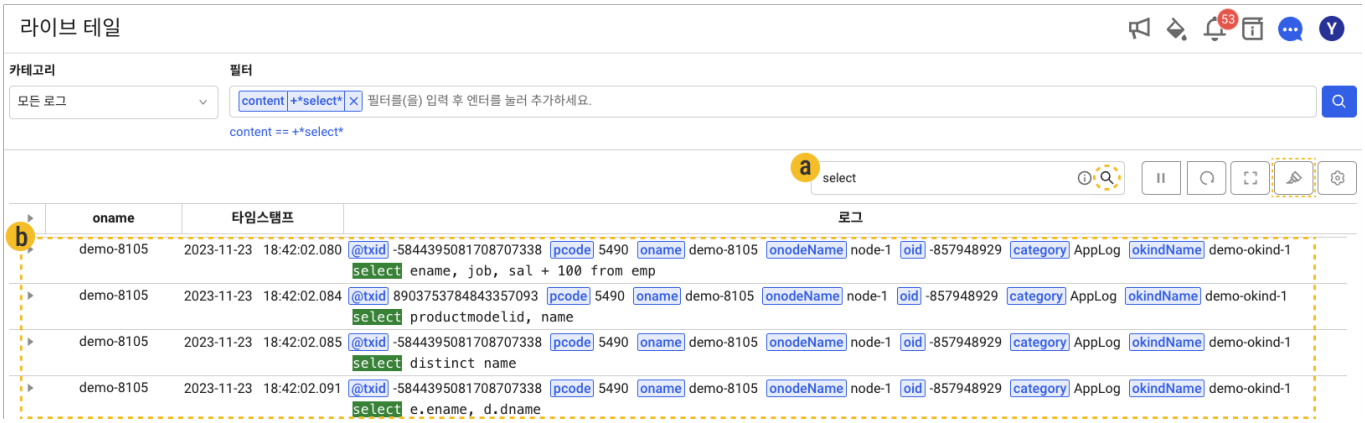
개인정보 조회

로그 개인정보 조회 권한이 있을 경우, **개인정보 표시** 토글 버튼 활성화를 통해 비식별화 처리한 개인정보를 표시할 수 있습니다.

> ⓘ
> - **로그 개인정보 조회** 권한에 대한 자세한 내용은 다음 문서를 참고하세요.
> - **로그 개인정보 비식별화** 기능에 대한 자세한 내용은 다음 문서를 참고하세요.

콘텐츠 하이라이트

로그의 콘텐츠 중 원하는 키워드를 손쉽게 식별하기 위해 하이라이트 기능을 제공합니다. 단일 또는 복수 키워드로 필터링할 수 있습니다.



1. 키워드 입력창에 하이라이트를 원하는 키워드를 입력하세요.
   - 예시, `select`
2. 로그 목록의 Content 내 검색한 키워드가 하이라이팅 됩니다.
   - **로그 전체 화면** 아이콘을 선택하면 **로그**와 **타임스탬프**를 전체 화면에서 확인할 수 있습니다.
   - 하이라이트된 키워드 간 이동은 단축키로 조작할 수 있습니다.
     - `Enter` : 다음 row로 이동
     - `Shift+Enter` : 이전 row로 이동

- ∧ / ∨ 아이콘: row 이동
- `Cmd` + `F` (Mac) / `Ctrl` + `F` (Windows): 키워드 검색

복수 키워드 조건

복수 키워드로 하이라이팅을 할 경우 다음과 같이 작성합니다.

입력 문자열	설명	결과
`a b c`	띄어쓰기로 각 키워드를 구분합니다.	a, b, c
`"Whatap is good."`	띄어쓰기를 키워드에 포함하고 싶은 경우 `''` 또는 `""`로 감쌉니다.	Whatap is good.
`"Whatap\\ is good."`	""로 감싸진 키워드에서 `\`를 포함할 경우, `\\`로 입력해야합니다.	Whatap\ is good.

하이라이트 색상 설정

하이라이팅할 키워드 및 색상을 설정할 수 있습니다.

- 기본적으로 로그 레벨에 따른 하이라이팅(FATAL, ERROR, WARN, SEVERE, CRITICAL)이 적용되어 있습니다.
- 설정한 내용은 **프로젝트 단위**로 저장됩니다.

1. 아이콘을 클릭하세요.
2. **하이라이트** 창에서 추가적으로 색상 설정을 원하는 키워드를 입력창에 입력하세요.
3. 입력창 왼쪽 색상 클릭 시 색을 선택할 수 있습니다.

테이블 설정

컬럼 설정

컬럼을 추가하거나 순서를 설정할 수 있습니다. 화면 오른쪽에서 **컬럼 설정** 버튼을 클릭하세요.

컬럼 추가

태그를 선택하여 테이블에 컬럼을 추가할 수 있습니다. **log** 컬럼 선택을 해제할 경우 **로그 표시 상세 설정**을 확인할 수 없습니다. 반드시 한 개 이상의 컬럼을 선택하세요.

- **제공되는 컬럼**: 로그, 수집 시간, content(로그 메시지)

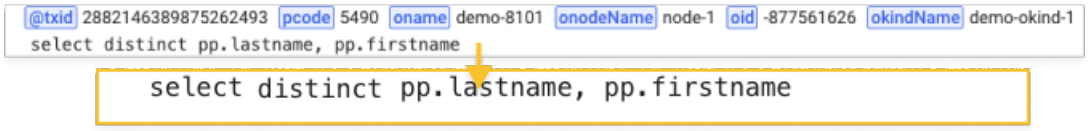
컬럼 순서 변경

컬럼을 추가하면 **표시 항목**에 해당 컬럼이 추가됩니다. 원하는 컬럼을 드래그하여 컬럼의 순서를 변경하세요.

로그 표시 상세 설정

1. 화면의 오른쪽에서 ⚙ **로그 표시 상세 설정** 버튼을 클릭하세요.
2. **로그 표시 상세 설정** 창에서 **content**와 **태그** 중 하나는 반드시 선택하세요.
    - 기본으로 **content**, **태그** 모두 선택되어있으며 두 가지 항목 모두 표시합니다.

체크가 해제된 항목은 테이블에 표시되지 않습니다. 다음과 같이 **태그**를 해제할 경우 테이블에서 로그의 **태그**는 표시되지 않습니다.



태그 관리 목록에 태그를 추가하면 추가한 순서대로 로그의 태그가 나열됩니다. 태그의 순서는 드래그하여 변경할 수 있습니다. 추가한 태그를 비활성화하면 비활성화한 태그는 로그의 태그에 노출되지 않습니다.

> ⓘ **컬럼 및 로그 표시 설정 적용 범위**
>
> - **컬럼 설정**과 **로그 표시 상세 설정**은 **라이브 테일**, **로그 검색**, **로그 트렌드**에서 사용할 수 있습니다.
> - 동일한 프로젝트 내 **라이브 테일**, **로그 검색**, **로그 트렌드**는 **컬럼 설정**과 **로그 표시 상세 설정**을 공유합니다.

로그 파일 다운로드

검색한 로그의 결과값을 CSV, TXT 파일 형식으로 수집된 로그를 다운로드 받을 수 있습니다.

1. 화면 중앙의 오른쪽의 **다운로드** 버튼을 클릭합니다.
2. **CSV 다운로드** 또는 **Log 다운로드**를 클릭해 원하는 형태의 파일을 다운로드 받습니다.

로그 탐색

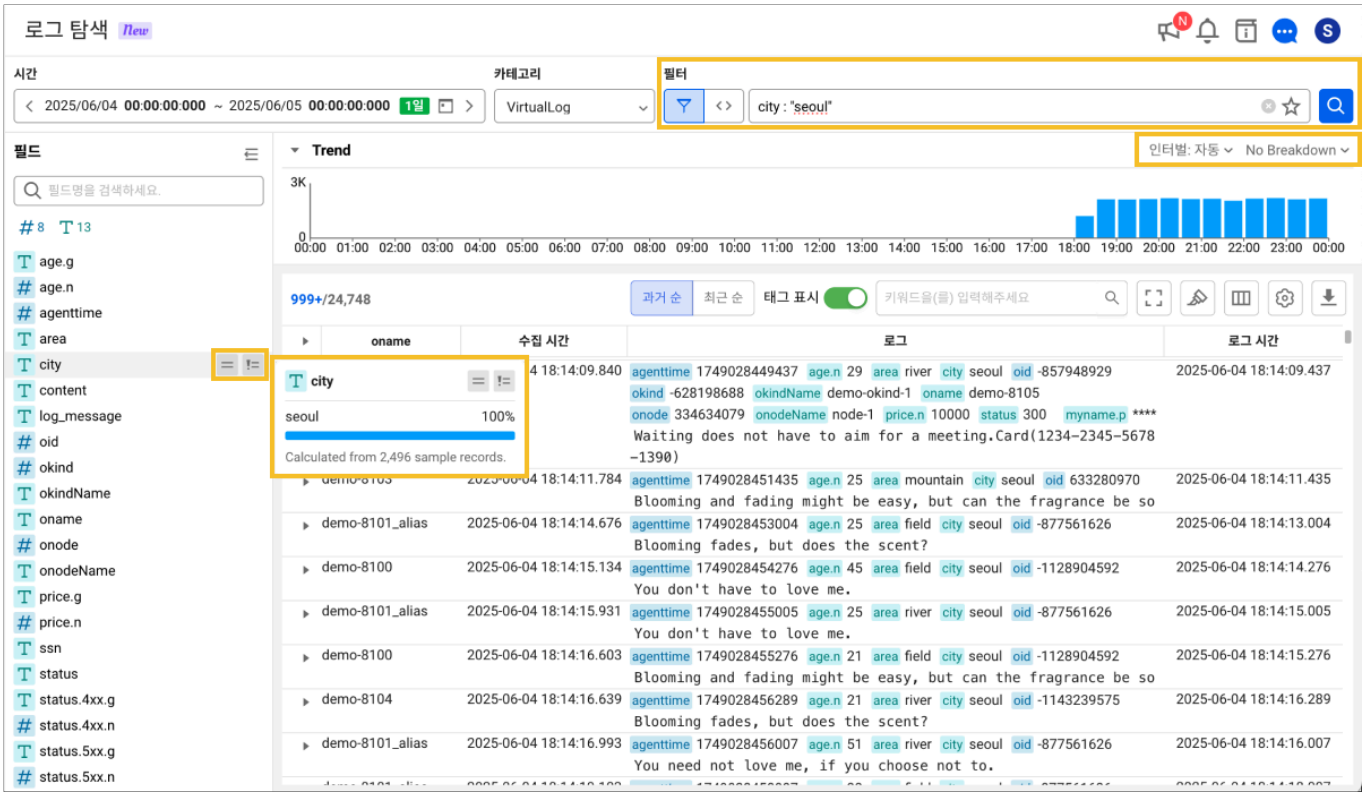
> **로그 조회** 권한이 없을 경우 해당 메뉴에 진입할 수 없습니다.

홈 화면 > 프로젝트 선택 > **로그** > **로그 탐색** *New*

로그 탐색(Log Explorer) 기능은 로그 데이터를 빠르게 검색, 필터링, 분석합니다. 필드 구조에 대한 정보를 파악하고, 쿼리 기반 필터링을 통해 필요한 로그를 효율적으로 검색해, 결과를 데이터 시각화하여 확인할 수 있습니다. 본 기능은 Lucene 인덱스 기반으로 동작하며, 주요 특징은 다음과 같습니다.

1. **로그 트렌드 분석 및 검색**: 로그 트렌드와 검색 기능을 동시에 지원하여, 시간에 따른 로그 변화 추이를 파악할 수 있습니다.
2. **빠른 검색 및 필터링**: 사용자는 간단한 단어 입력만으로도 문제의 로그를 신속하게 검색할 수 있습니다. WhaTap log search query와 Lucene 쿼리 언어를 지원하여, 다양한 검색 조건을 설정할 수 있습니다.
3. **자동완성 기능**: 쿼리 입력 시 자동완성 기능을 통해 사용자가 쉽게 쿼리를 작성하고 학습할 수 있도록 돕습니다.
4. **데이터 시각화**: 스택 그래프를 활용하여 시계열 로그 건수 데이터를 시각적으로 표현하며, 드릴다운 기능을 통해 특정 로그를 심층 분석할 수 있습니다.
5. **즐겨찾기 및 공유**: 자주 사용하는 필터를 즐겨찾기로 저장하고, URL을 통해 다른 사용자와 쉽게 공유할 수 있습니다.

기본 화면 안내



시간 설정

상단 옵션바의 **타임셀렉터**를 통해 로그 검색 시간 범위를 설정할 수 있습니다.

카테고리

상단 옵션바의 **카테고리 선택**에서 탐색하고 싶은 로그 종류 선택합니다. (예: AppLog 등)

필터

상단 옵션 바의 **필터**에서 쿼리 언어를 선택하고 검색 조건을 입력하여 필터를 적용합니다. 검색 쿼리 문법은 WhaTap 로그 검색 쿼리와 Lucene 쿼리를 지원합니다.

- : WhaTap 로그 검색 쿼리
- : Lucene 쿼리

검색 쿼리 문법

WhaTap log search query

WhaTap 로그 검색 쿼리는 자동 완성 기능을 제공하며, 간단한 단어 입력만으로도 문제의 로그를 신속하게 검색할 수 있습니다. and, or 등 다양한 문법을 활용해 다양한 검색 조건을 입력할 수 있습니다.

- 예, 검색 필터에 `error` 만 입력하여 content(로그 본문) 검색할 수 있습니다.

    1. 상단 옵션 바에서 필터 아이콘을 클릭한 후, 쿼리 검색창에서 필드(선택한 시간과 카테고리에 맞는 목록 노출)를 선택하세요.
    2. 연산자를 선택하세요.

        a. 기본: `:` , `:*` , `and` , `or`
        b. 선택한 필드가 `Number` 인 경우(기본 + `<=` , `>=` , `<` , `>`)
        c. 선택한 필드가 `String` 인 경우(기본)

    3. 값(선택한 시간, 카테고리, 필드에 맞는 목록 노출)을 선택하세요.

검색 인덱스 필드 타입

지원 필드 타입은 `Number` , `String` , `Boolean` 세 가지 유형만 존재합니다.

- `Number` : Long, double, float 타입의 숫자 값
- `String` : 이메일 본문과 같은 전체 텍스트
- `Boolean` : True 또는 False 값

> WhaTap log search query 문법에 대한 자세한 내용은 다음 문서를 참고하세요.

Lucene

apache lucene 8.11.1 버전 오픈소스 문법을 그대로 사용하세요. 단, 자동 완성 기능을 제공하지 않습니다.

> Lucene의 QueryParser 클래식 패키지에 대한 자세한 내용은 공식 문서를 참고하세요.

필드 목록

드릴 다운 분석에 활용할 수 있습니다.

필드 정보 필터 반영하기

필드 목록에서 필드명을 확인하고, `=`, `!=` 버튼을 클릭해 조건을 필터에 적용하세요.

- 필드에 마우스 호버 시 `=`, `!=` 버튼이 보이고 클릭 시 필터에 반영됩니다.
- 필드 검색 기능을 통해 원하는 필드를 빠르게 찾을 수 있습니다.

Trend 차트

Trend 차트에서 **인터벌**, **Breakdown**을 적용해 그래프로 로그를 확인할 수 있습니다.

- **그래프 인터랙션**: 마우스 드래그 및 🔍 버튼으로 시간 줌인, 바 클릭으로 해당 영역의 로그 확인 가능
- **인터벌**: 자동, 1초, 5초, 30초, 1분, 5분, 10분, 30분, 1시간 중 선택 가능
- **Breakdown**: 특정 필드를 기준으로 Top 3 + Others 값을 분포 시각화하고, 드릴다운 필터 적용 가능

breakdown

breakdown은 로그 건수 추이를 특정 필드로 분석해주는 시각화 기능입니다.

- 특정 필드의 Top 3 값과 Others(나머지 값)를 구분하여 스택 그래프로 분포를 표시합니다.
- 드릴다운 분석에 활용할 수 있으며, 범례와 스택 그래프에서 특정 영역(부분 그래프)를 클릭해 특정 값만 강조하거나 필터 조건을 추가할 수 있습니다.

로그 확인하기

검색 조건에 맞는 로그 목록을 확인할 수 있으며, **필드 목록**에서 필드 정보와 로그 태그 색이 동일하게 표시됩니다.

콘텐츠 하이라이트

로그의 콘텐츠 중 원하는 키워드를 손쉽게 식별하기 위해 하이라이트 기능을 제공합니다. 단일 또는 복수 키워드로 필터링할 수 있습니다.

1. 키워드 입력창에 하이라이트를 원하는 키워드를 입력하세요.
    - 예시, `select`
2. 로그 목록의 Content 내 검색한 키워드가 하이라이팅 됩니다.
    - ⛶ **로그 전체 화면** 아이콘을 선택하면 **로그**와 **타임스탬프**를 전체 화면에서 확인할 수 있습니다.
    - 하이라이트된 키워드 간 이동은 단축키로 조작할 수 있습니다.
        - `Enter` : 다음 row로 이동
        - `Shift+Enter` : 이전 row로 이동
        - ∧ / ∨ 아이콘: row 이동
        - `Cmd` + `F` (Mac) / `Ctrl` + `F` (Windows): 키워드 검색

복수 키워드 조건

복수 키워드로 하이라이팅을 할 경우 다음과 같이 작성합니다.

입력 문자열	설명	결과
`a b c`	띄어쓰기로 각 키워드를 구분합니다.	a, b, c
`"Whatap is good."`	띄어쓰기를 키워드에 포함하고 싶은 경우 `''` 또는 `""`로 감쌉니다.	Whatap is good.
`"Whatap\\ is good."`	""로 감싸진 키워드에서 `\`를 포함할 경우, `\\`로 입력해야합니다.	Whatap\ is good.

하이라이트 색상 설정

하이라이팅할 키워드 및 색상을 설정할 수 있습니다.

- 기본적으로 로그 레벨에 따른 하이라이팅(FATAL, ERROR, WARN, SEVERE, CRITICAL)이 적용되어 있습니다.

- 설정한 내용은 **프로젝트 단위**로 저장됩니다.

1. 아이콘을 클릭하세요.
2. **하이라이트** 창에서 추가적으로 색상 설정을 원하는 키워드를 입력창에 입력하세요.
3. 입력창 왼쪽 색상 클릭 시 색을 선택할 수 있습니다.

컬럼 설정

버튼을 클릭해, 컬럼을 추가하거나 순서를 설정할 수 있습니다.

- **컬럼 추가**

  컬럼 설정에서 태그를 선택하여 테이블에 컬럼을 추가할 수 있습니다.

  > 로그 컬럼 선택을 해제할 경우 **로그 표시 상세 설정**을 확인할 수 없습니다. 반드시 한 개 이상의 컬럼을 선택하세요.

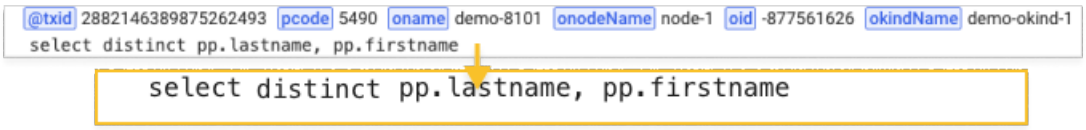
- **컬럼 순서 설정**

  컬럼 설정에서 컬럼을 추가하면 **표시 항목**에 해당 컬럼이 추가됩니다. 원하는 컬럼을 드래그하여 컬럼의 순서를 변경하세요.

로그 표시 상세 설정

버튼을 클릭해, 로그 표시를 상세 설정합니다. 기본으로 **content**, **태그** 모두 체크가 되어있으며 두 가지 항목 모두 표시합니다.

- **content**와 **태그** 중 하나는 반드시 선택하세요.
- **태그 관리** 목록에 태그를 추가하면 추가한 순서대로 로그의 태그가 나열됩니다. 태그의 순서는 드래그하여 변경할 수 있습니다. 추가한 태그를 비활성화하면 비활성화한 태그는 로그의 태그에 노출되지 않습니다.
- 체크가 해제된 항목은 테이블에 표시되지 않습니다. 다음과 같이 **태그**를 해제할 경우 테이블에서 로그의 **태그**는 표시되지 않습니다.



> **컬럼 및 로그 표시 설정 적용 범위**

- **컬럼 설정**과 **로그 표시 상세 설정**은 **라이브 테일**, **로그 검색**, **로그 트렌드**에서 사용할 수 있습니다.
- 동일한 프로젝트 내 **라이브 테일**, **로그 검색**, **로그 트렌드**는 **컬럼 설정**과 **로그 표시 상세 설정**을 공유합니다.

로그 파일 다운로드

검색한 로그의 결과값을 CSV, TXT 파일 형식으로 수집된 로그를 다운로드 받을 수 있습니다.

1. 화면 중앙의 오른쪽의 **다운로드** 버튼을 클릭합니다.
2. **CSV 다운로드** 또는 **Log 다운로드**를 클릭해 원하는 형태의 파일을 다운로드 받습니다.

WhaTap log search query 문법

> WhaTap 로그 탐색에서 사용할 수 있는 WhaTap log search query 문법에 관한 문서입니다.

1. **Equal**
   - `field : value`
2. **Not**
   - `not field : value`
3. **Wildcard** (`String` 타입 필드만 사용 가능, 검색 값 앞뒤에 `"` 사용 금지)
   - `field : val*`
4. **Exist**
   - `field : *`
5. **Range** (`Number` 타입 필드만 사용 가능)
   - `<` , `>` , `<=` , `>=` 사용 가능
   - 예: `field < 30 and field >= 10`
6. **Combining Multiple Queries**
   - AND 예: `field < 30 and field >= 10`
   - OR 예: `field < 30 or field >= 10`
7. **Grouping**
   - 예: `(field < 30 or field >= 10) and other : 70`
8. **Escaping Special Characters**
   - Lucene의 이스케이프 지원 특수 문자 목록: `&&` `||` `!` `(` `)` `{` `}` `^` `"` `~` `*` `?` `:` `\` `/`
   - 문자 앞에 `\` 를 사용해 이스케이프함
   - 예: `(1+1):2` 검색하려면, `\(1\+1\)\:2`

검색 인덱스 필드 타입

지원 필드 타입은 `Number` , `String` 두 가지 유형만 존재합니다.

- `Number` : Long, double, float 타입의 숫자 값
  - `field : "value"` 로 사용
- `String` : 이메일 본문과 같은 전체 텍스트
  - `field : value` 로 사용
- `Boolean` : True 또는 False
  - `field : true` 또는 `field : false` 로 사용

로그 트렌드

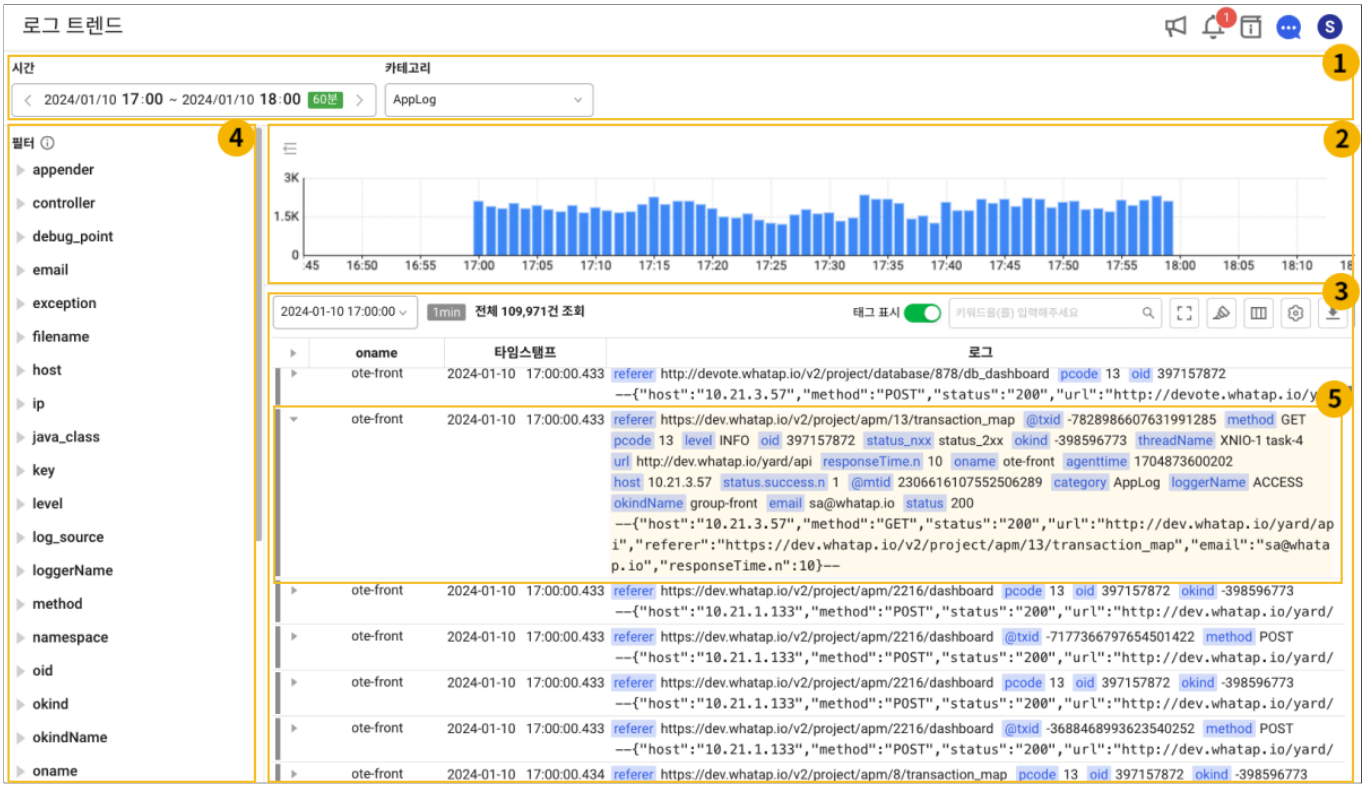
> ⓘ **로그 조회** 권한이 없을 경우 해당 메뉴에 진입할 수 없습니다.

홈 화면 > 프로젝트 선택 > **로그** > **로그 트렌드**

로그 트렌드에서 유형별로 분류된 로그의 발생 건수 추이를 통해 특정 에러 유형의 발생 패턴을 확인하고 시간별 상세 로그 데이터를 확인할 수 있습니다. 하이라이트 기능을 통해 원하는 로그를 빠르게 식별할 수 있습니다. 카테고리별로 수집된 로그의 추이를 조회할 수 있습니다.

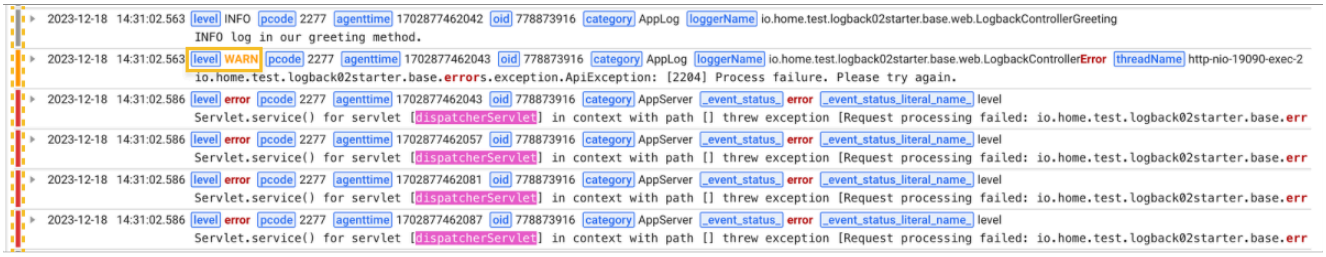
주요 용어는 다음과 같습니다.

- **Category**: 로그의 수집 및 조회 단위
- **Content**: 로그 메시지
- **Search Key**: 로그 파서 설정을 통해 생성되는 검색 키
- **Tag**: 수집된 로그를 검색할 수 있는 태그



데이터 조회하기

- 스크롤이 바닥에 닿으면 다음 데이터를 조회합니다.
- 1에서 탐색할 로그 데이터의 **시간**과 수집 단위인 **카테고리**를 지정합니다.
- **카테고리**를 변경하면 선택된 카테고리에 해당하는 로그를 조회합니다. 2 바 차트와 3 로그 테이블에서 확인할 수 있습니다.
- 2 바 차트의 막대를 클릭하면 막대의 시간 범위에 해당하는 로그를 조회합니다.
- 3 로그 테이블 상단 왼쪽에서 조회한 총 로그 개수를 확인할 수 있습니다.
- 3 로그 테이블 상단 오른쪽 **전체 화면** 아이콘을 선택하면 **로그**와 **수집 시간**을 전체 화면에서 확인할 수 있습니다.
- 4 사이드 메뉴에서 태그로 필터를 걸어서 로그를 확인할 수 있습니다. 검색 키는 2개까지 선택할 수 있고, 검색값은 복수 개 선택이 가능합니다.
- **에이전트 옵션이 설정된 경우** 로그 레벨을 수집해 로그 레벨 기준 색상이 다음과 같이 표시됩니다.



에이전트 옵션 설정

```
# whatap.conf
weaving=log4j-2.17
weaving=logback-1.2.8
```

- Java 에이전트 2.2.22 버전 이후부터 위빙 설정에 log4j-2.17 또는 logback-1.2.8 설정 시 사용할 수 있습니다. 에이전트 재시작이 필요합니다.
- 로그 레벨은 파싱된 키워드 중 `level`, `type` 기준으로 판별합니다. `level`, `type` 으로 파싱된 키가 존재하고 파싱 값이 FATAL, CRITICAL, ERROR, WARN, WARNING, INFO를 포함할 경우 로그 레벨 색상을 표시합니다.

> ⓘ 로그 테이블 컬럼 가장자리를 드래그해 컬럼 너비를 수정할 수 있습니다.

개인정보 조회

로그 개인정보 조회 권한이 있을 경우, **개인정보 표시** 토글 버튼 활성화를 통해 비식별화 처리한 개인정보를 표시할 수 있습니다.

> ⓘ
> - **로그 개인정보 조회** 권한에 대한 자세한 내용은 다음 문서를 참고하세요.
> - **로그 개인정보 비식별화** 기능에 대한 자세한 내용은 다음 문서를 참고하세요.

로그 Content 확인하기

Content는 로그 메시지입니다. 로그 컬럼의 첫 번째 줄은 로그의 파싱(parsing)된 키와 값이고 두 번째 줄은 Content입니다.

- 3 로그 테이블의 행(로그)마다 ▸ **더보기** 버튼이 있습니다. ▸ **더보기** 버튼을 선택하면 5처럼 해당 로그의 전체 content를 확인할 수 있습니다.

하이라이트 설정하기

원하는 키워드를 손쉽게 식별할 수 있도록 하이라이트 기능을 제공합니다. 아이콘을 클릭해 키워드를 입력 후 엔터를 눌러 추가하세요.

콘텐츠 하이라이트

로그의 콘텐츠 중 원하는 키워드를 손쉽게 식별하기 위해 하이라이트 기능을 제공합니다. 단일 또는 복수 키워드로 필터링할 수 있습니다.

1. 키워드 입력창에 하이라이트를 원하는 키워드를 입력하세요.
   - 예시, `select`
2. 로그 목록의 Content 내 검색한 키워드가 하이라이팅 됩니다.
   - **로그 전체 화면** 아이콘을 선택하면 **로그**와 **타임스탬프**를 전체 화면에서 확인할 수 있습니다.
   - 하이라이트된 키워드 간 이동은 단축키로 조작할 수 있습니다.
     - `Enter` : 다음 row로 이동
     - `Shift+Enter` : 이전 row로 이동
     - ∧ / ∨ 아이콘: row 이동
     - `Cmd` + `F` (Mac) / `Ctrl` + `F` (Windows): 키워드 검색

복수 키워드 조건

복수 키워드로 하이라이팅을 할 경우 다음과 같이 작성합니다.

입력 문자열	설명	결과
`a b c`	띄어쓰기로 각 키워드를 구분합니다.	a, b, c
`"Whatap is good."`	띄어쓰기를 키워드에 포함하고 싶은 경우 `'` 또는 `"` 로 감쌉니다.	Whatap is good.
`"Whatap\\ is good."`	`"`로 감싸진 키워드에서 `\`를 포함할 경우, `\\`로 입력해야합니다.	Whatap\ is good.

하이라이트 색상 설정

하이라이팅할 키워드 및 색상을 설정할 수 있습니다.

- 기본적으로 로그 레벨에 따른 하이라이팅(FATAL, ERROR, WARN, SEVERE, CRITICAL)이 적용되어 있습니다.
- 설정한 내용은 **프로젝트 단위**로 저장됩니다.

1. 아이콘을 클릭하세요.
2. **하이라이트** 창에서 추가적으로 색상 설정을 원하는 키워드를 입력창에 입력하세요.
3. 입력창 왼쪽 색상 클릭 시 색을 선택할 수 있습니다.

차트로 로그 조회하기

- 바 차트에서 a 원하는 시간 영역 바를 클릭하거나 드래그하여 해당 시간 범위의 로그를 확인할 수 있습니다.
- 바 차트 아래 로그 테이블 상단 왼쪽의 b 시간 선택 옵션을 이용해 다음과 같이 선택한 시간대에서 더 세분화해 로그를 검색할 수 있습니다.
  - 1min: interval (차트의 막대 사이 간격)
  - 시간 선택 옵션: 선택된 시간 범위를 6개의 구간으로 나눈 시간대

테이블 설정하기

컬럼 설정

컬럼을 추가하거나 순서를 설정할 수 있습니다. 3 영역 오른쪽에서 **컬럼 설정** 버튼을 클릭하세요.

컬럼 추가

태그를 선택하여 테이블에 컬럼을 추가할 수 있습니다. **log** 컬럼 선택을 해제할 경우 **로그 표시 상세 설정**을 확인할 수 없습니다. 반드시 한 개 이상의 컬럼을 선택하세요.

- **제공되는 컬럼**: 로그, 수집 시간, content(로그 메시지)

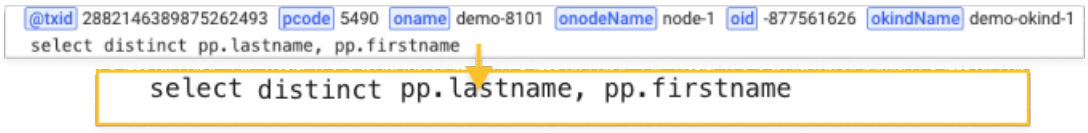
컬럼 순서 변경

컬럼을 추가하면 **표시 항목**에 해당 컬럼이 추가됩니다. 원하는 컬럼을 드래그하여 컬럼의 순서를 변경하세요.

로그 표시 상세 설정

3 영역 오른쪽에서 ⚙ **로그 표시 상세 설정** 버튼을 선택하세요. 기본으로 **content**, **태그** 모두 체크가 되어있으며 두 가지 항목 모두 표시합니다. **content**와 **태그** 중 하나는 반드시 선택하세요.

체크가 해제된 항목은 테이블에 표시되지 않습니다. 다음과 같이 **태그**를 해제할 경우 테이블에서 로그의 **태그**는 표시되지 않습니다.



태그 관리 목록에 태그를 추가하면 추가한 순서대로 로그의 태그가 나열됩니다. 태그의 순서는 드래그하여 변경할 수 있습니다. 추가한 태그를 비활성화하면 비활성화한 태그는 로그의 태그에 노출되지 않습니다.

> ⓘ **컬럼 및 로그 표시 설정 적용 범위**
>
> - **컬럼 설정**과 **로그 표시 상세 설정**은 **라이브 테일**, **로그 검색**, **로그 트렌드**에서 사용할 수 있습니다.
> - 동일한 프로젝트 내 **라이브 테일**, **로그 검색**, **로그 트렌드**는 **컬럼 설정**과 **로그 표시 상세 설정**을 공유합니다.

로그 파일 다운로드

검색한 로그의 결과값을 CSV, TXT 파일 형식으로 수집된 로그를 다운로드 받을 수 있습니다.

1. 화면 중앙의 오른쪽의 **다운로드** 버튼을 클릭합니다.
2. **CSV 다운로드** 또는 **Log 다운로드**를 클릭해 원하는 형태의 파일을 다운로드 받습니다.

로그 검색

> ⓘ **로그 조회** 권한이 없을 경우 해당 메뉴에 진입할 수 없습니다.

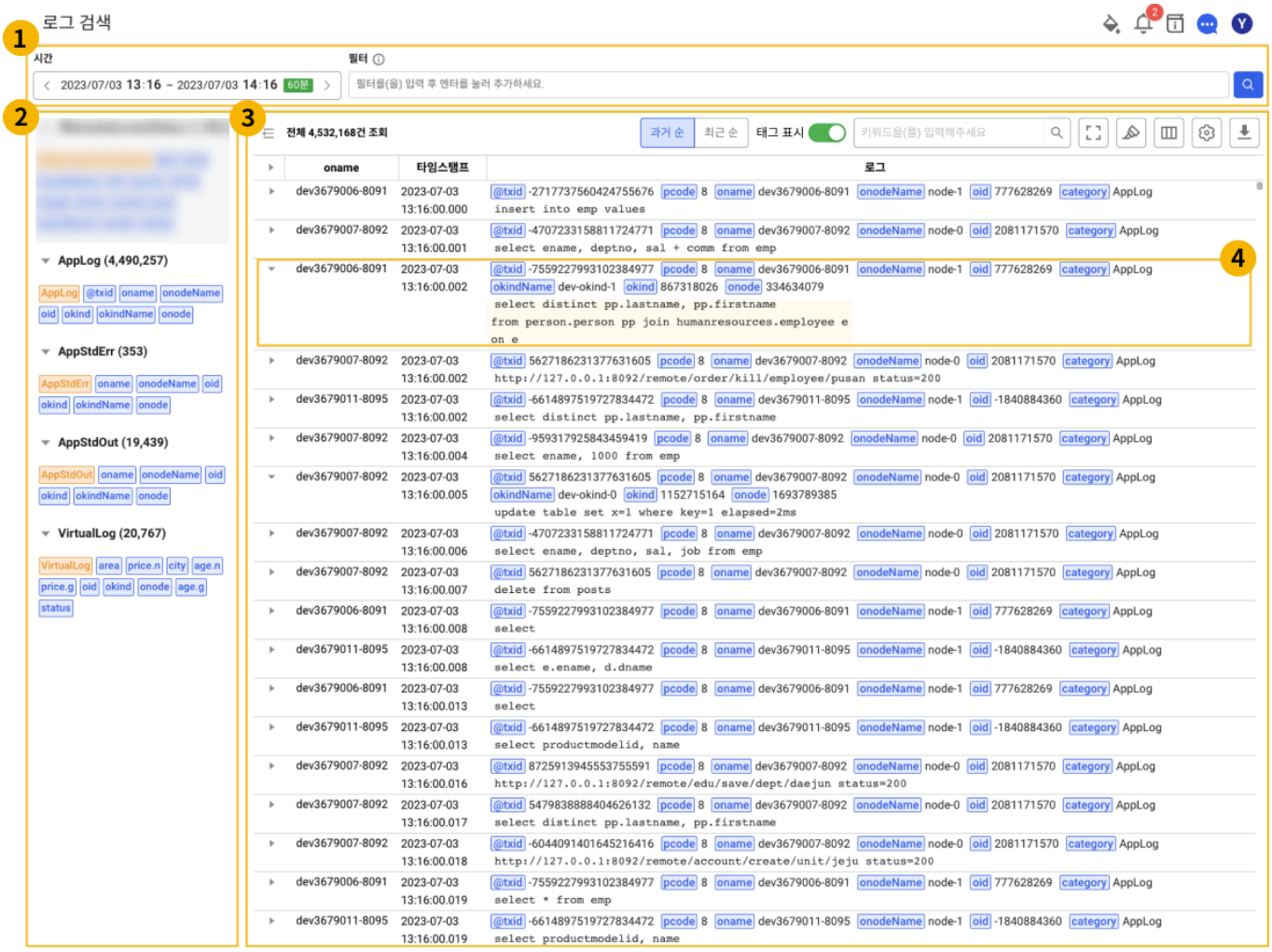
홈 화면 > 프로젝트 선택 > **로그** > **로그 검색**

로그 검색 메뉴에서 통합 수집된 대량의 로그를 다양한 조건으로 검색하고 사용자가 원하는 로그를 특정할 수 있습니다. 복수의 검색 조건을 파싱된 키와 밸류로 지정할 수 있어 원하는 조건에 일치하는 로그 데이터만 추출합니다.

동적 페이지로 검색된 로그 데이터를 정해진 라인 단위로 가져오며, 스크롤 등에 의해 하단에 닿으면 자동으로 다음 데이터를 가져와 표시합니다.

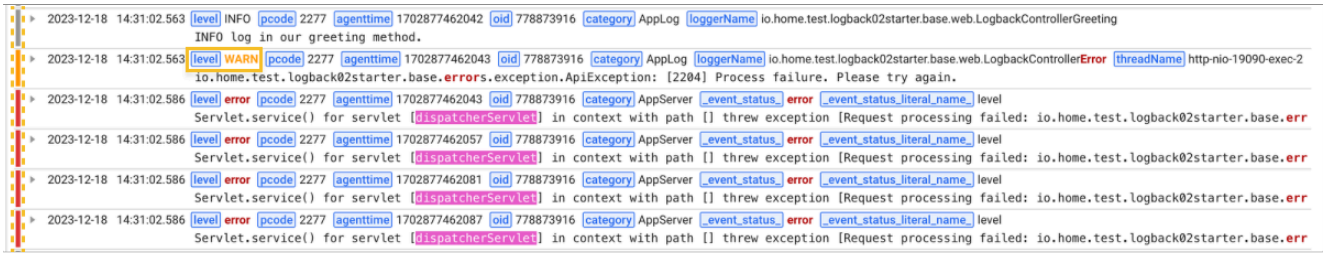
주요 용어는 다음과 같습니다.

- **Category**: 로그의 수집 및 조회 단위
- **Content**: 로그 메시지
- **Search Key**: 로그 파서 설정을 통해 생성되는 검색 키
- **Tag**: 수집된 로그를 검색할 수 있는 태그



데이터 조회하기

- 스크롤이 바닥에 닿으면 다음 데이터를 조회합니다. 한 번에 10,000개의 로그를 조회합니다.
- 3 로그 테이블 상단 왼쪽에서 조회한 총 로그 개수를 확인할 수 있습니다.
- 로그 데이터를 시간 순과 역순으로 조회할 수 있습니다. 3 로그 테이블 상단 오른쪽에서 **과거 순**과 **최근 순** 중 원하는 조회 방식을 선택하세요.
- 시간 범위 지정 후 **적용** 버튼을 선택 해 조회 시간을 설정하고 **검색** 버튼을 선택해 데이터를 조회합니다.
- 3 로그 테이블 상단 오른쪽 **로그 전체 화면** 아이콘을 선택하면 **로그**와 **타임스탬프**를 전체 화면에서 확인할 수 있습니다.
- **에이전트 옵션이 설정된 경우** 로그 레벨을 수집해 로그 레벨 기준 색상이 다음과 같이 표시됩니다.



에이전트 옵션 설정

```
# whatap.conf
weaving=log4j-2.17
weaving=logback-1.2.8
```

- Java 에이전트 2.2.22 버전 이후부터 위빙 설정에 log4j-2.17 또는 logback-1.2.8 설정 시 사용할 수 있습니다. 에이전트 재시작이 필요합니다.
- 로그 레벨은 파싱된 키워드 중 `level`, `type` 기준으로 판별합니다. `level`, `type` 으로 파싱된 키가 존재하고 파싱 값이 FATAL, CRITICAL, ERROR, WARN, WARNING, INFO를 포함할 경우 로그 레벨 색상을 표시합니다.

> ⓘ 로그 테이블 컬럼 가장자리를 드래그해 컬럼 너비를 수정할 수 있습니다.

로그 Content 확인하기

Content는 로그 메시지입니다. 로그 컬럼의 첫 번째 줄은 로그의 파싱(parsing)된 키와 값이고 두 번째 줄은 Content입니다.

- 3 로그 테이블의 행(로그)마다 ▸ **더보기** 버튼이 있습니다. ▸ **더보기** 버튼을 선택하면 4처럼 해당 로그의 전체 Content를 확인할 수 있습니다.
- 로그의 태그를 선택하면 **복사**, **검색**, **제외 검색**, **인접 로그**를 검색 할 수 있는 드롭다운 메뉴가 나타납니다.

필터

필터 적용

1 왼쪽 **시간 선택창**에서 시간 범위를 지정할 수 있습니다. 오른쪽에서 필터를 적용하면 입력한 조건에 맞는 로그를 필터링합니다. 복수의 필터를 입력할 수 있습니다. 필터의 태그가 같은 경우 OR(`||`)로, 그렇지 않은 경우는 AND(`&&`)로 적용됩니다.

입력 창에 값을 직접 입력하거나 **필터** 입력 창을 클릭해 필터를 지정할 수 있습니다. 필터 태그는 `검색 키` , `연산자` , `검색 값` 의 순서로 입력합니다. **검색** 버튼을 선택하면 필터가 적용된 데이터를 로그 목록에서 조회할 수 있습니다.

가이드 UI

다음과 같이 입력 창 아래 **가이드 UI**를 제공합니다. 입력 창 또는 필터 태그에 마우스 커서가 있을 때 ESC 키를 통해 **가이드 UI**를 닫을 수 있습니다.

검색 키, 연산자, 검색 값 입력

- `검색 키` 입력 시 일반 인덱스, 예약어 인덱스, 숫자만 입력할 수 있는 인덱스를 구분해 추천 값을 제공합니다
- `연산자` 입력 시 일반 인덱스 검색 키의 경우 `==` , `!=` 옵션을 하단에 안내합니다. 숫자만 입력할 수 있는 인덱스의 경우 `>` , `<` , `<=` , `>=` , `==` , `!=` 옵션을 제공합니다.
- `검색 값` 입력 시 일치 검색(`>` , `<` , `<=` , `>=` , `==`)일 때 파란색으로, 제외 검색(`!=`)일 때 붉은색으로 하이라이팅합니다. 대소문자 구분 옵션을 활용해 검색할 수 있습니다.

> ⓘ
> - 필터 태그가 2줄 이상 길어지는 경우 ∧ **접기** 아이콘을 선택해 접어둘 수 있습니다.
> - 필터 태그 입력 후 입력 창에서 Shift와 Tab 키를 동시에 사용하여 이전 필터 태그로 이동할 수 있습니다.
> - 필터 태그 입력 후 입력 창에서 Tab 키를 통해 다음 필터 태그로 이동할 수 있습니다.

필터 태그 추가

입력 창에 텍스트 입력 후 Enter 또는 Tab 키를 누르거나, 입력창 하단의 가이드 UI에서 추천 값을 클릭하거나 방향키로 선택해 추가할 수 있습니다.

필터 태그 제거

Backspace 키 또는 태그의 **X** 아이콘을 눌러 태그를 삭제할 수 있습니다. 입력창 오른쪽의 **X** 아이콘을 클릭해 모든 태그를 한 번에 삭제할 수 있습니다.

필터 즐겨찾기

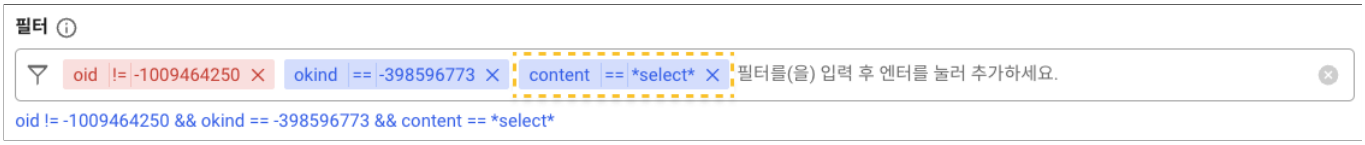
필터 **즐겨찾기** 기능을 제공합니다. 입력 창 오른쪽의 ☆ **즐겨찾기** 아이콘을 선택하여 원하는 필터 검색 조건을 즐겨찾기에 **추가**, **제거** 및 기존 즐겨찾기를 선택할 수 있습니다. **즐겨찾기**는 최대 50개까지 저장됩니다.

필터 적용 예외 상황

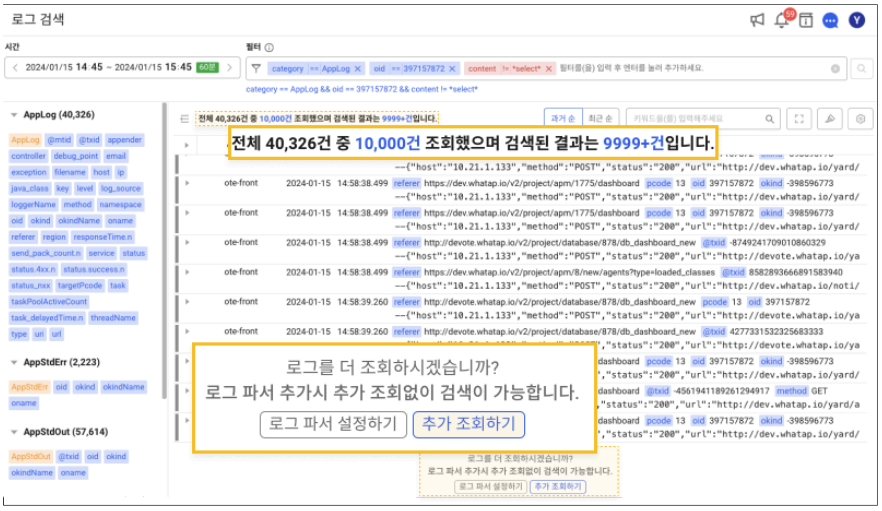
- 숫자만 입력할 수 있는 인덱스(.n 으로 끝나는 검색 키)를 입력한 태그에서 검색 값 은 숫자만 입력할 수 있습니다.
- 중복된 검색 키 , 검색 값 은 입력할 수 없습니다.
- 검색 키 , 검색 값 중 하나라도 없는 태그가 존재할 때 검색할 수 없습니다. 유효하지 않는 태그의 경우 회색으로 표시합니다.

미파싱 키워드 필터 적용

로그에서 파싱되지 않은 즉 인덱스가 생성되지 않은 키워드를 포함한 로그를 조회할 수 있습니다. 이 경우 지정 범위 내 모든 로그를 Full Scan합니다. 그렇기 때문에 인덱스가 생성된 키와 비교해 검색 속도가 다소 떨어질 수 있습니다. 정형화된 로그 데이터의 경우 **로그 파서 설정**을 통해 인덱스 키 값을 활용해 검색하는 것을 권장합니다.



1. **카테고리**를 선택하세요. 카테고리 지정이 필수적입니다.
2. **필터** 입력창에 content 기준 띄어쓰기 후 검색을 원하는 키워드를 입력하세요.
   예시, content *select*
3. **검색** 버튼을 클릭해 로그를 조회하세요. 전체 로그 중 일부 먼저 조회합니다. 1회당 검색 결과는 최대 1만 건입니다.
4. 스크롤을 내려 하단의 **추가 조회하기** 버튼 선택 시 추가 조회할 수 있습니다.



ⓘ 전체 로그 중 서버 조회 범위 당 1만 건씩 조회합니다. 서버 조회 범위의 경우 기본 20만 건이지만 전체 로그 양에 따라 비율이 달라질 수 있습니다.
파서 설정에 대한 자세한 내용은 다음 문서를 참조하세요.

필터 수정

필터에 값을 입력한 뒤 입력한 값을 클릭하면 해당 값을 수정할 수 있습니다.

- 입력 창에 텍스트 재입력해 수정할 수 있습니다.
- 입력 창 아래 **가이드 UI**를 통해 추천 값을 선택해 수정할 수 있습니다.

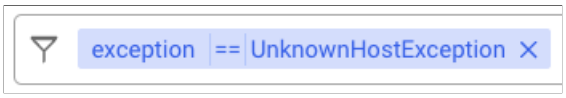
검색 키(Search Key)

검색 키는 로그 데이터 내에서 원하는 특정 값에 접근하기 위한 식별자를 의미합니다. 검색 키에 해당하는 실제 데이터가 검색 값입니다. 왼쪽 2 영역에 있는 태그는 카테고리별로 파싱(parsing) 된 검색 키입니다. 태그를 선택하여 필터를 입력할 수 있습니다. 주황색 태그는 카테고리, 파란색 태그는 검색 키입니다.

예를 들어 2 영역의 AppLog와 AppStdOut은 카테고리, 그 아래 oid와 같은 태그는 파싱(parsing) 된 검색 키입니다. 검색 키는 **로그** > **로그 설정**의 **로그 파서 설정** 탭에서 파싱 로직을 등록해 설정할 수 있습니다.

필터 입력 문법

태그는 검색 키와 검색 값으로 구성되어있습니다. 다음의 예시에서 검색 키는 `exception` , 검색 값은 `UnknownHostException` 입니다. 해당 예시는 수집한 로그 데이터 중 IP 주소와 도메인 주소가 매칭되지 않아 서버를 호스트에 연결할 수 없을 경우 발생하는 예외(`UnknownHostException`)가 포함된 로그 데이터를 조회합니다.



검색 키 종류

검색 키 종류	검색 키 포멧	의미	검색 키와 검색 값 예시	검색 예시
문자열 키워드	keyword	파일 이름	- 키: fileName - 값: /data/whatap/logs/yard.log	fileName:/data/whatap/logs/yard.log
숫자 키워드	keyword.n	응답시간	- 키: response_time.n - 값: 2945	response_time.n>=2945
예약어 키워드 (사전 정의 키워드)	@keyword	트랜잭션 ID	- 키: @txid - 값: 85459614215434144	-

공통 문법

문법 종류	설명	예시
`==searchValue`	검색 값과 일치하는 로그를 검색합니다.	`exception==RuntimeExceptionexception`
`!=searchValue`	검색 값을 제외한 로그를 검색합니다.	`exception!=RuntimeException`
`*searchValue`	검색 값으로 끝나는 로그를 검색합니다.	`word==*hello`
`searchValue*`	검색 값으로 시작하는 로그를 검색합니다.	`word==hello*`
`*searchValue*`	검색 값으로 중간에 포함된 로그를 검색합니다.	`word==*hello*`
`*search*Value*`	검색 값으로 포함된 로그를 검색합니다.	`word==*he*llo*`
`re:{regex}`	정규표현식에 매칭되는 로그를 검색합니다.	`caller==re:^i\.w\.a\.w\.s\.v\.r\.`

문법 종류	설명	예시
**	검색 키에 해당하는 모든 로그를 검색합니다.	

검색 키가 숫자 키워드(keyword.n)인 경우 문법

다음의 문법은 검색 키가 keyword.n 형식인 경우에만 지원합니다.

- 검색 값으로는 숫자만 올 수 있습니다.
- .n 키워드의 값에는 prefix를 붙이지 않습니다. .n 이 아닌 키워드는 모두 prefix를 붙여야합니다.

  예, +>searchValue 는 유효하지 않습니다.

문법 종류	설명	예시
>searchValue	검색 값보다 큰 값이 포함된 로그를 조회합니다.	response_time.n>3000
>=searchValue	검색 값보다 크거나 같은 값이 포함된 로그를 조회합니다.	response_time.n>=3000
==searchValue	검색 값보다 같은 값이 포함된 로그를 조회합니다.	response_time.n==3000
!=searchValue	검색 값보다 다른 값이 포함된 로그를 조회합니다.	response_time.n!=3000
<searchValue	검색 값보다 작은 값이 포함된 로그를 조회합니다.	response_time.n<3000
<=searchValue	검색 값보다 작거나 같은 값이 포함된 로그를 조회합니다.	response_time.n<=3000

개인정보 조회

로그 개인정보 조회 권한이 있을 경우, **개인정보 표시** 토글 버튼 활성화를 통해 비식별화 처리한 개인정보를 표시할 수 있습니다.

- **로그 개인정보 조회** 권한에 대한 자세한 내용은 다음 문서를 참고하세요.
- **로그 개인정보 비식별화** 기능에 대한 자세한 내용은 다음 문서를 참고하세요.

로그 태그 옵션

로그 태그 선택 시 드롭다운 메뉴가 나타납니다. **검색, 제외 검색, 인접 로그, 트랜잭션 정보** 옵션을 확인할 수 있습니다.

로그 태그 옵션 메뉴	설명
복사	태그에 해당하는 로그를 복사합니다.
검색	**필터**에 해당 태그가 포함('==') 조건으로 입력됩니다.
제외 검색	**필터**에 해당 태그가 제외('!=') 조건으로 입력됩니다.
인접 로그	**인접 로그** 상세 창에서 선택한 로그의 서버를 대상으로 선택한 로그와 인접한 시간대의 로그를 조회합니다. 시간을 선택해 인접한 시간대의 로그를 조회할 수 있습니다. 기준 로그는 파란색 바탕으로 표시됩니다.
트랜잭션 정보	로그에 `@txid` 및 `@mtid` 태그가 포함된 경우, **트랜잭션 정보**를 클릭하면 **트랜잭션 정보** 상세창에서 정보를 확인할 수 있습니다.

콘텐츠 하이라이트

로그의 콘텐츠 중 원하는 키워드를 손쉽게 식별하기 위해 하이라이트 기능을 제공합니다. 단일 또는 복수 키워드로 필터링할 수 있습니다.

1. 키워드 입력창에 하이라이트를 원하는 키워드를 입력하세요.

- 예시, `select`

2. 로그 목록의 Content 내 검색한 키워드가 하이라이팅 됩니다.
   - **로그 전체 화면** 아이콘을 선택하면 **로그**와 **타임스탬프**를 전체 화면에서 확인할 수 있습니다.
   - 하이라이트된 키워드 간 이동은 단축키로 조작할 수 있습니다.
     - `Enter` : 다음 row로 이동
     - `Shift+Enter` : 이전 row로 이동
     - ∧ / ∨ 아이콘: row 이동
     - `Cmd` + `F` (Mac) / `Ctrl` + `F` (Windows): 키워드 검색

복수 키워드 조건

복수 키워드로 하이라이팅을 할 경우 다음과 같이 작성합니다.

입력 문자열	설명	결과
`a b c`	띄어쓰기로 각 키워드를 구분합니다.	a, b, c
`"Whatap is good."`	띄어쓰기를 키워드에 포함하고 싶은 경우 `''` 또는 `""`로 감쌉니다.	Whatap is good.
`"Whatap\\ is good."`	""로 감싸진 키워드에서 `\`를 포함할 경우, `\\`로 입력해야합니다.	Whatap\ is good.

하이라이트 색상 설정

하이라이팅할 키워드 및 색상을 설정할 수 있습니다.

- 기본적으로 로그 레벨에 따른 하이라이팅(FATAL, ERROR, WARN, SEVERE, CRITICAL)이 적용되어 있습니다.
- 설정한 내용은 **프로젝트 단위**로 저장됩니다.

1. 아이콘을 클릭하세요.
2. **하이라이트** 창에서 추가적으로 색상 설정을 원하는 키워드를 입력창에 입력하세요.
3. 입력창 왼쪽 색상 클릭 시 색을 선택할 수 있습니다.

테이블 설정하기

컬럼 설정

컬럼을 추가하거나 순서를 설정할 수 있습니다. 3 영역 오른쪽에서 **컬럼 설정** 버튼을 클릭하세요.

컬럼 추가

태그를 선택하여 테이블에 컬럼을 추가할 수 있습니다. **log** 컬럼 선택을 해제할 경우 **로그 표시 상세 설정**을 확인할 수 없습니다. 반드시 한 개 이상의 컬럼을 선택하세요.

- **제공되는 컬럼**: 로그, 수집 시간, content(로그 메시지)

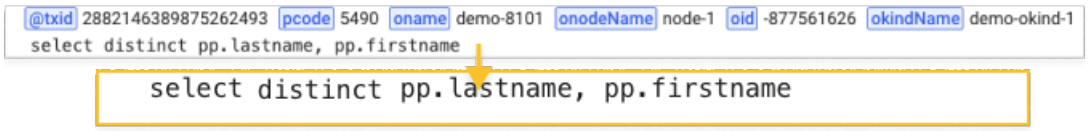
컬럼 순서 변경

컬럼을 추가하면 **표시 항목**에 해당 컬럼이 추가됩니다. 원하는 컬럼을 드래그하여 컬럼의 순서를 변경하세요.

로그 표시 상세 설정

3 영역 오른쪽에서 **로그 표시 상세 설정** 버튼을 선택하세요. 기본으로 **content**, **태그** 모두 체크가 되어있으며 두 가지 항목 모두 표시합니다. **content**와 **태그** 중 하나는 반드시 선택하세요.

체크가 해제된 항목은 테이블에 표시되지 않습니다. 다음과 같이 **태그**를 해제할 경우 테이블에서 로그의 **태그**는 표시되지 않습니다.



태그 관리 목록에 태그를 추가하면 추가한 순서대로 로그의 태그가 나열됩니다. 태그의 순서는 드래그하여 변경할 수 있습니다. 추가한 태그를 비활성화하면 비활성화한 태그는 로그의 태그에 노출되지 않습니다.

> **컬럼 및 로그 표시 설정 적용 범위**
>
> - **컬럼 설정**과 **로그 표시 상세 설정**은 **라이브 테일**, **로그 검색**, **로그 트렌드**에서 사용할 수 있습니다.

- 동일한 프로젝트 내 **라이브 테일**, **로그 검색**, **로그 트렌드**는 **컬럼 설정**과 **로그 표시 상세 설정**을 공유합니다.

로그 파일 다운로드

검색한 로그의 결과값을 CSV, TXT 파일 형식으로 수집된 로그를 다운로드 받을 수 있습니다.

1. 화면 중앙의 오른쪽의 **다운로드** 버튼을 클릭합니다.
2. **CSV 다운로드** 또는 **Log 다운로드**를 클릭해 원하는 형태의 파일을 다운로드 받습니다.

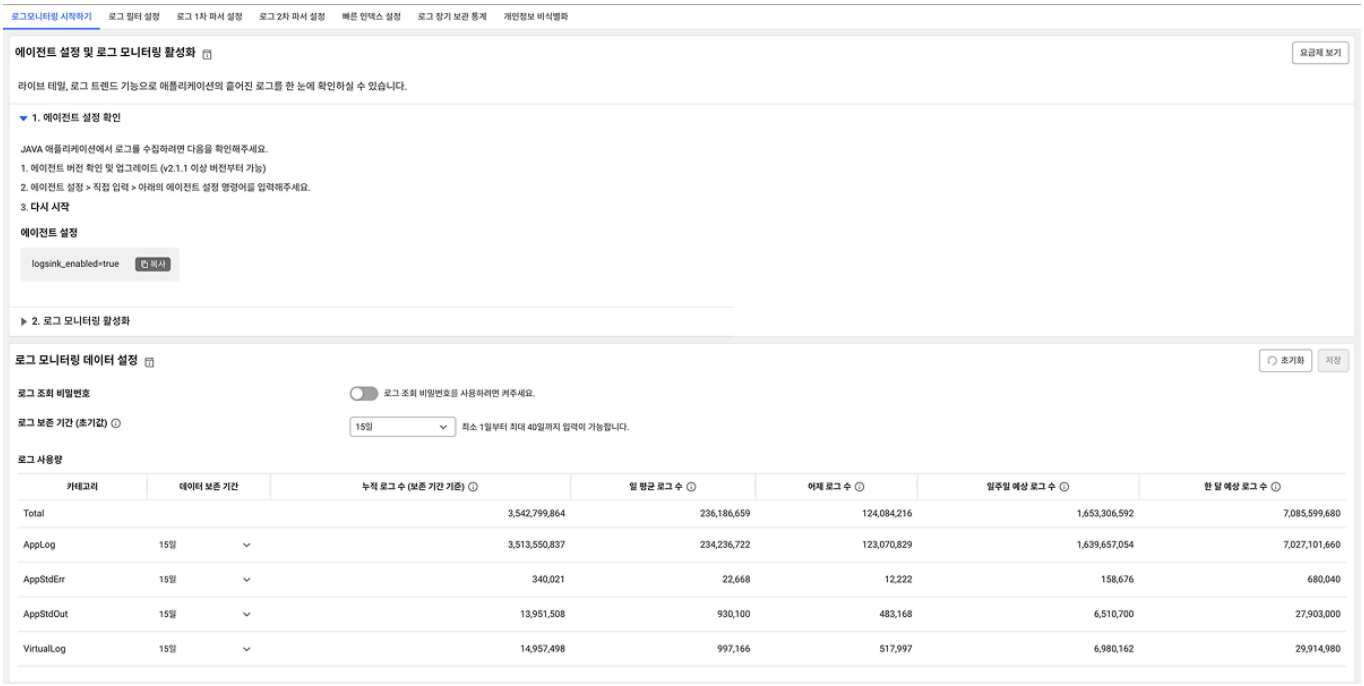
로그 설정

홈 화면 > 프로젝트 선택 > 로그 > 로그 설정

로그 설정의 상단의 탭에서 로그 모니터링을 설정할 수 있습니다. 에이전트 설정 확인, 로그 모니터링의 활성화 여부 결정, 로그 데이터의 유지 기간 및 조회 비밀번호 설정, 로그 파서 등록, 빠른 인덱스 설정, 개인정보 비식별화 관리 등을 할 수 있습니다.

> **권한**
>
> - **로그 모니터링 활성화**을 사용하려면 **프로젝트 수정** 권한이 필요합니다.
> - **로그 편집** 권한이 있으면 **로그 모니터링 활성화** 외 **로그 설정** 메뉴를 수정할 수 있습니다.

로그 모니터링 시작하기



1. **로그 > 로그 설정**의 **로그모니터링 시작하기** 탭을 클릭하세요.
2. **가이드 보기** 아이콘과 **요금제 보기** 버튼을 클릭하면 안내 화면으로 이동합니다.

에이전트 설정 및 로그 모니터링 활성화

1 영역에서 에이전트 설정을 확인하고 로그 모니터링 활성화 및 비활성화 여부를 설정할 수 있습니다.

에이전트 설정 확인

로그 모니터링을 시작하기 위해 에이전트 버전과 설정 정보를 확인하세요. **1. 에이전트 설정 확인**에서 안내에 따라 진행하세요.

- **애플리케이션 모니터링**

  애플리케이션별 적용 방법은 Java, PHP, Node.js, Python, Go 문서를 참고하세요.

- **서버 모니터링**

  서버 모니터링 적용 방법은 Server 문서를 참고하세요. 안내에 따라 whatap.conf 에 로그 감시 대상 파일 설정을 추가하세요.

- **쿠버네티스 모니터링**

  쿠버네티스 모니터링 적용 방법은 Kubernetes 문서를 참고하세요.

로그 모니터링 활성화

로그 모니터링을 활성화 또는 비활성화할 수 있습니다. **2. 로그 모니터링 활성화**에서 로그 모니터링 여부를 설정하세요.

- 로그 모니터링 활성화: 토글 버튼을 켜면 로그 모니터링이 활성화되고, 활성화한 날부터 15일 동안 무료로 체험하실 수 있습니다.
- 로그 모니터링 비활성화: 토글 버튼을 끄면 로그 모니터링이 비활성화되고, 로그를 더 이상 저장하지 않습니다.

로그 모니터링 데이터 설정

로그 사용량을 확인할 수 있습니다. 또한 **로그 보존 기간** 및 **로그 조회 비밀번호** 설정을 변경할 수 있습니다.

로그 조회 비밀번호

보안을 강화하기 위해 **로그 조회 비밀번호**를 설정하세요. 로그 조회 비밀번호 지정은 선택 사항입니다. 로그 조회 비밀번호를 사용 중이라면 로그 화면 진입 시 반드시 비밀번호를 입력해야 합니다.

> **비밀번호 분실**

로그 편집 권한이 있는 사용자는 **로그 설정**에서 새 비밀번호로 변경할 수 있습니다.

로그 보존 기간

공통으로 적용할 기본(default) 데이터 보존 기간입니다.

- 최소 1일부터 최대 40일까지 입력할 수 있습니다. 미지정 시 기본값은 1일입니다. 로그 보존 기간 선택 옵션 외에 사용자가 원하는 기간을 입력할 수 있습니다.
- **로그 사용량** 목록에서 별도로 설정하지 않으면 이 로그 데이터 보존 기간이 기본적으로 적용됩니다. **로그 사용량** 목록에서 카테고리별 데이터 보존 기간을 설정하고 **초기화** 버튼을 클릭하면 기본 데이터 보존 기간으로 초기화됩니다.

로그 사용량

로그 사용량 목록에서 카테고리별 로그 데이터 보존 기간을 지정할 수 있습니다. 로그 수는 해당 기간 동안 쌓인 로그 건수를 의미합니다. 예를 들어 **금일 로그수**는 하루 동안 쌓인 로그 건수, **예상 로그수**는 데이터 보관일에 금일 로그 수를 곱한 로그 건수를 의미합니다.

로그 데이터 보존 기간을 다음과 같이 지정할 수 있습니다. 기간 지정에 따라 오래된 데이터를 삭제해 공간을 확보할 수 있습니다.

- **트라이얼 프로젝트**

  데이터 보존 기간으로 1일, 2일, 3일을 선택할 수 있습니다.

- **유료 프로젝트**

  데이터 보존 기간으로 1일, 2일, 3일, 4일, 5일, 6일, 7일, 10일, 30일, 40일을 선택할 수 있습니다.

- **저장량 기준 과금**

  데이터 보존 기간에 따라 비용이 달라집니다.

  예, 일 평균 200만 로그 건수가 쌓이고 데이터 보존 기간을 3일로 지정한 경우라면 평균 600만 로그 건수가 수집 서버에 유지되고 과금 대상이 됩니다.

로그 필터 설정

로그 필터 설정은 사용자가 지정한 필터 조건에 해당하는 로그만 저장하는 기능입니다. 사용자가 필요한 로그만 저장하여 비용을 절감할 수 있습니다.

> **권한**
>
> 로그 편집 권한이 있는 사용자만 설정할 수 있습니다.

로그 필터 설정 추가하기

새로운 필터 설정을 추가하는 방법입니다. 로그 필터 설정 시 **필터 이름, 라이브 테일 표시 여부, 카테고리, 필터 조건**은 필수 입력 항목입니다.

로그 필터 설정에서 저장하지 않는 로그도 **라이브 테일 표시 여부** 토글을 활성화하면 **라이브 테일**에서 확인할 수 있습니다.

> 카테고리별로 하나의 필터만 설정할 수 있습니다.

1. **로그 > 로그 설정 > 로그 필터 설정** 탭으로 이동하세요.
2. 로그 필터 설정을 위해 **+ 추가하기** 버튼을 클릭하세요.
3. **로그 필터 설정 추가**에서 **필터 이름, 라이브 테일 표시 여부, 카테고리, 필터 조건**을 입력하세요.
    - **필터 이름**: 필터를 식별할 수 있는 이름
    - **라이브 테일 표시 여부**: 저장 여부와 관계없이 라이브 테일에 표시 여부 선택
    - **카테고리**: 필터가 적용될 로그 카테고리 **(복수 선택 가능)**
    - **필터 조건**: 필터 조건에 저장할 로그의 조건

    > 필터 조건 설정 시, 자세한 내용은 WhaTap log search query 문법를 참고하세요.

    - **예시**

- **포함**: `oid : 123`
  - oid가 123인 로그만 수집합니다.
- **불포함**: `not oid : 123`
  - oid가 123이 **아닌** 로그만 수집합니다.
- **전체 포함**: 필터 조건 미입력
- **전체 불포함**: `not content : *`

4. **추가** 버튼을 클릭하면 **등록 결과 미리보기**에서 각 필터 설정을 확인하고 수정할 수 있습니다.
5. 확인이 완료되었다면 **적용** 버튼을 클릭해 로그 필터 설정을 적용합니다.

시뮬레이션

사용자가 입력한 카테고리와 필터 조건으로 로그를 미리 검색하여, 로그 필터 설정을 적용하기 전에 결과를 확인할 수 있습니다.

시뮬레이션 버튼을 클릭하면 로그 필터 설정이 적용된 **로그 검색** 화면이 나타납니다.

로그 필터 설정 수정하기

등록한 로그 필터는 필요에 따라 수정 아이콘을 클릭해 수정할 수 있습니다. **로그 필터 설정 수정**에서 기존에 입력한 값이 표시되며, 원하는 항목을 변경한 뒤 적용하면 됩니다.

> ⚠ 로그 필터 설정을 수정할 때, 하나의 카테고리만 선택할 수 있습니다.

로그 필터 설정 삭제하기

등록한 필터는 삭제 아이콘을 클릭해 삭제할 수 있습니다. 삭제 후 반드시 **저장** 버튼을 클릭하세요.

> ⚠ 변경 사항을 저장하지 않으면 필터가 삭제되지 않습니다.

로그 필터 설정 JSON 가져오기/내보내기

로그 필터 설정은 JSON 형식으로 Import/Export 할 수 있습니다. 다른 프로젝트로 동일한 로그 필터 설정을 쉽게 적용할 수 있습니다. 다른 프로젝트로 로그 필터 설정 내용을 그대로 옮기고 싶은 경우 유용하게 사용할 수 있습니다.

JSON 내보내기

1. 화면 오른쪽 위의 JSON 버튼을 클릭하세요.
2. **JSON 내보내기** 창이 열리면, **내보내기** 버튼을 클릭하세요.
3. 로그 필터 설정 내용이 JSON 형식의 파일로 다운로드됩니다.

JSON 가져오기

1. 화면 오른쪽 위의 버튼을 클릭하세요.
2. 가져올 JSON 형식의 파일을 선택하세요.
3. **JSON 가져오기** 창에서 **목록에 추가하기** 또는 **덮어쓰기** 버튼을 클릭해 로그 필터 설정을 가져옵니다.
4. **저장** 버튼을 클릭해 변경 사항을 반영합니다.

유효성 검사 조건

설정 내용 저장 시 유효성 검사 내용은 다음과 같습니다.

- **type**: `GENERAL` 만 지원
- **category**: 필수값으로 `string` 타입 지원
- **condition**
  - **language**: `jsonDsl` , `lucene` 만 지원
  - **description**: 유효하지 않은 필터 조건이면 저장 불가
- **showLiveTail**: `boolean` 타입만 지원
- **name**: 필수값으로 `string` 타입 지원

로그 파서 설정

로그 1차 파서 설정

로그 설정 상단의 **로그 1차 파서 설정** 탭을 선택해 수집된 로그에 대한 파서를 등록할 수 있습니다.

로그 1차 파서는 **GROK**과 **JSON** 파서를 제공하며 유형별 로그 발생 수 집계나 특정 로그 탐색을 위한 필수 파서입니다. 패턴 조건에 맞는 검색 키와 검색 값을 추출하며, 파싱된 키는 로그 유형 분류 및 검색 인덱싱에 활용합니다.

- **GROK**: 기본은 정규 표현식 기반 파싱에 해당합니다. 예약 키워드 기반의 파싱을 제공합니다.
- **JSON**: 로그 중 JSON으로 출력된 부분에 대해서 일괄 파싱을 제공합니다.

> **파싱 로직 미등록 시 검색 가능한 key**
>
> `category` , `oid` , `oname` , `okind` , `okindName` , `@txid` , `@login` , `httphost`
>
> **파서 등록이 불가한 예약어**
>
> `timestamp` , `message` , `pcode` , `category` , `content` , `logContent`
>
> - 다음 예약어의 경우 파서를 등록하더라도 인덱스가 생성되지 않습니다.

> 로그 파서에 대한 자세한 내용은 다음 문서를 참고하세요.

설정 항목

설정 값	설명	기타
카테고리	패턴을 적용할 카테고리	required
로그 검출 조건	필터로 적용할 검색 키, 검색 값을 입력 - 로그 검출 조건에 맞는 로그 데이터에만 패턴 적용, 로그 검출 조건을 입력하지 않으면 모든 로그를 대상으로 패턴 적용	optional

설정 값	설명	기타
패턴	로그를 파싱(parsing)할 패턴 - 작성한 패턴에 맞추어서 파싱을 하고 인덱스 생성 GROK, 정규 표현식 문법 지원	required

파서 목록

등록한 파서를 조회하고, 추가 및 편집할 수 있습니다. 파서는 등록 순서대로 적용되며, 최초로 일치하는 파서만 적용됩니다.

- 상단 오른쪽 **+ 추가하기** 버튼을 클릭하면 **파서 추가** 창이 나타납니다.
- **적용 순서** 컬럼의 ⦀ 아이콘을 드래그해 파서 설정 순서를 변경할 수 있습니다.
- **활성화** 토글로 파서 활성화 여부를 지정할 수 있습니다.
- **수정** 및 **삭제** 아이콘을 통해 등록한 파서를 수정 및 삭제할 수 있습니다.

파서 등록

다음은 파서를 등록하는 공통 순서입니다.

1. **로그 1차 파서 설정**에서 **+ 추가하기** 버튼을 클릭하세요.
2. **파서 추가** 창에서 **파서**를 선택하세요.

   > **파서 및 패턴 등록 방법**
   >
   > - **GROK** 파서 및 패턴 등록
   > - **JSON** 파서 및 패턴 등록

3. 파서를 선택하면 **패턴 등록** 버튼이 활성화됩니다. **패턴 등록** 버튼을 클릭한 후, **파서 시뮬레이션** 창에서 패턴과 로그를 입력합니다.

   a. **패턴**을 입력하세요.

   b. **로그**의 **카테고리**와 **시뮬레이션 파싱 로그**를 입력하세요. **로그 검색** 버튼을 클릭한 후, 검색된 목록에서 원하는 로그 오른쪽의 **선택** 버튼을 클릭하면 로그가 자동으로 입력됩니다.

> ⓘ **로그 검색**을 통해 수집된 로그 중 예시 로그를 선택할 수 있습니다. 선택한 로그는 시뮬레이션 입력창에 자동 반영되어 복사/붙여넣기 없이 바로 테스트할 수 있습니다.

c. 등록하려는 패턴이 정상인지 **시뮬레이션** 버튼을 클릭해 시뮬레이션 및 패턴의 퍼포먼스를 측정하세요.

> ⓘ 시뮬레이션과 퍼포먼스 측정에 관한 자세한 내용은 다음 문서를 참고하세요.

4. 성공적으로 파싱되는지 시뮬레이션 결과를 확인한 후, **패턴 적용** 버튼을 클릭하세요.
5. **추가** 버튼을 클릭하면 파서가 추가됩니다.

> ⓘ **개인정보 비식별화**
>
> 파서 등록 시 검색 키 이름을 `검색 키.p` 로 지정하면 비식별화 대상이 됩니다(예: `myname.p` `**`). 와탭은 **개인정보 비식별화** 탭에서 비식별화를 관리하는 방식을 권장합니다. 자세한 내용은 개인정보 비식별화를 참고하세요.

GROK 파서 패턴 등록

기본 문법은 `%{SYNTAX:SEMANTIC}` 입니다. GROK 파서에 대한 자세한 내용은 다음 문서를 참조하세요. 패턴 등록 및 시뮬레이션이 필수적입니다.

- **SYNTAX**

  GROK 정의 패턴입니다.

- **SEMANTIC**

  파싱된 데이터에 할당할 키입니다.

> ⓘ SEMANTIC에는 예약어 등이 사용되지 않도록 조합어 사용을 권장합니다.

JSON 포맷 파서 패턴 등록

로그 전체 또는 일부가 JSON 형태로 출력되는 경우 해당 부분을 파싱할 수 있습니다. **Prefix**, **Postfix** 옵션으로 JSON 인식 범위를 지정합니다. JSON 파서에 대한 자세한 내용은 다음 문서를 참조하세요. 시뮬레이션이 필수적입니다.

옵션	설명
Prefix	JSON 문자열의 시작 부분 앞의 문자열을 지정합니다. 미지정 시 로그 출력문의 맨 앞부터 JSON 문자열로 식별합니다.
Postfix	JSON 문자열의 종료 부분 뒤의 문자열을 지정합니다. 미지정 시 로그 출력문의 맨 뒤까지를 JSON 문자열로 식별합니다.
Ignore	JSON 출력부 중 키 추출을 제외할 필드를 지정합니다.

- **등록 예시**

  Log

  ```
  [2022-10-25 10:15:34:145]...(개행)
  Request : {"key1":"value1","key2":"value2",...}(개행)
  Response : {"key3":"value3","key4":"value4",...}
  ```

  예시처럼 유입되는 로그가 Request JSON, Response JSON을 모두 파싱하고자 하는 경우 다음의 2가지 패턴을 등록합니다.

  - Request 파싱용 패턴

    "Request : " 와 "Response" 사이의 문자열 `{"key1":"value1","key2":"value2",...}` 대상

  - Response 파싱용 패턴

    "Response : " 부터 로그의 마지막 까지의 문자열 `{"key3":"value3","key4":"value4",...}` 대상

- **JSON 커스텀 패턴 등록**

  로그 중 일부가 JSON 형태로 출력되는 경우 JSON으로 출력된 부분을 전용 커스텀 파서를 통해 파싱할 수 있습니다. 패턴을 다음과 같이 입력하세요.

  ```
  io.whatap.logsink.parser.JsonFormatParser{}
  ```

  로그 중 JSON 형태로 출력된 부분을 검출하기 위해 **Prefix**, **Postfix** 옵션을 조합해 로그의 어느 부분을 JSON으로 인식하여 파싱할지 지정하세요.

  `JsonFormatParser{}` 의 `{}` 에 옵션을 지정합니다.

  - **등록 예시**

    Log

```
[2022-10-25 10:15:34:145]...(개행)
Request : {"key1":"value1","key2":"value2",...}(개행)
Response : {"key3":"value3","key4":"value4",...}
```

예시처럼 유입되는 로그가 Request JSON, Response JSON을 모두 파싱하고자 하는 경우 다음의 2가지 패턴을 등록합니다.

- Request 파싱용 패턴

  "Request : " 와 "Response" 사이의 문자열 `{"key1":"value1","key2":"value2",...}` 대상

```
io.whatap.logsink.parser.JsonFormatParser {prefix:"Request : ",postfix:"Response"}
```

- Response 파싱용 패턴

  "Response : " 부터 로그의 마지막 까지의 문자열 `{"key3":"value3","key4":"value4",...}` 대상

```
io.whatap.logsink.parser.JsonFormatParser {prefix: "Response : "}
```

파서 시뮬레이션 및 퍼포먼스 측정

파서 시뮬레이션 후 패턴을 등록할 수 있습니다. **퍼포먼스 측정**은 시뮬레이션 수행 대상 문자열에 대하여 파서의 반복 파싱 소요 시간을 측정합니다.

1. **패턴**과 **로그**를 입력하세요. **로그 검색** 버튼을 클릭한 후, 검색된 목록에서 원하는 로그 오른쪽의 **선택** 버튼을 클릭하면 로그가 자동으로 입력됩니다.
2. **시뮬레이션** 버튼을 클릭하여 등록하려는 패턴으로 파싱에 성공하는지 확인하세요.
3. 시뮬레이션 성공 시 **시뮬레이션 결과** 및 **퍼포먼스 측정 결과**를 조회할 수 있습니다.
4. 시뮬레이션 후 **패턴 적용** 버튼 클릭 시 선택한 파서에 입력한 패턴이 적용됩니다.

파싱 성공

파싱 로직을 등록해 **키**(key)가 생성되면 로그 조회 시 해당 키로 파싱된 값이 추가됩니다. 다음 **라이브 테일** 메뉴 예시와 같이 파싱된 키와 값이 추가됩니다.

파싱된 키는 **라이브 테일**, **로그 검색**, **로그 트렌드**에서 확인할 수 있습니다.

로그 2차 파서 설정

로그 설정 메뉴 상단에서 **로그 2차 파서 설정** 탭을 선택해 로그 파서를 등록할 수 있습니다.

로그 2차 파서는 1차 파서(GROK 또는 JSON)로 추출한 값을 가공해 통계 데이터나 새로운 필드를 생성합니다. HTTP 상태 코드 기반 통계 파서와 함께 키워드 필터, 필드 가공, 응답 시간 분류, GeoIP 변환 등 다양한 프로세서를 제공합니다.

> ⓘ 로그 2차 파서는 1차 파싱된 결과에 대하여 특수 목적의 2차 파싱 기능을 제공합니다. 2차 파서를 사용하기 위해서는 **1차 파서**가 **등록**되어 있어야 합니다.

파서 목록

등록한 파서를 조회하고, 추가 및 편집할 수 있습니다.

파서 추가, 순서 변경, 활성화 토글, 수정 및 삭제 등의 조작 방법은 1차 파서 목록과 동일합니다.

파서 등록 순서

다음은 파서를 등록하는 공통 순서입니다.

1. **+ 추가하기** 버튼을 선택하면 **파서 추가** 창이 나타납니다.
2. **파서** 선택 창에서 파서를 선택하세요. 각 파서별 설정 항목은 다음 문서를 참조하세요.
   - **4xx, 5xx 상태 코드 파서** 설정 항목
   - **상태 코드 성공률 파서** 설정 항목
3. **제외할 상태 코드**를 입력하세요.
4. **카테고리** 선택 창에서 카테고리를 선택하거나 직접 입력하세요.
5. **로그 검출 조건**을 선택하거나 직접 입력하세요.
6. **추가** 버튼을 선택해 파서를 등록하세요.

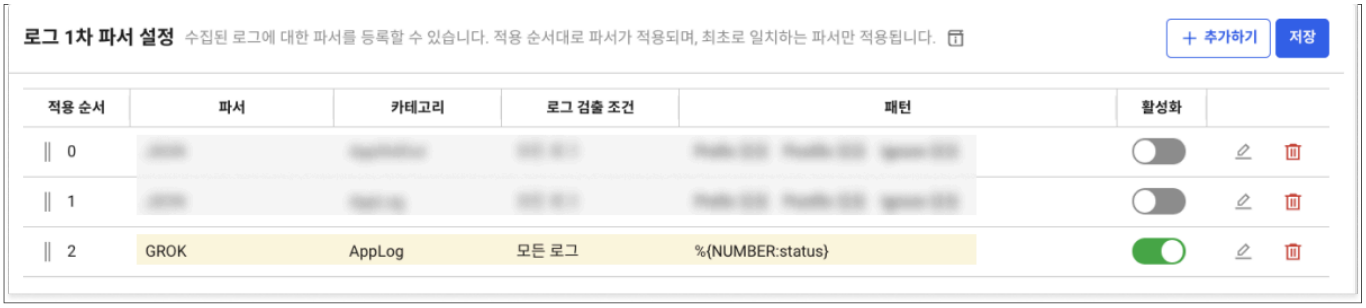
4xx, 5xx 상태 코드 파서 설정 항목

4xx, 5xx 상태 코드 파서는 1차 파서로 파싱된 status 값을 기반으로 4xx, 5xx 건수 데이터를 생성합니다. 특정 상태 코드를 제외하려면 제외할 상태 코드를 입력하세요.

설정 항목

설정 값	설명	기타
카테고리	4xx, 5xx건수 데이터를 생성할 카테고리입니다.	required
로그 검출 조건	필터로 적용할 검색 키, 검색 값을 입력합니다. 로그 검출 조건에 맞는 로그 데이터에 대해서만 4xx, 5xx건수 데이터를 생성합니다. 로그 검출 조건을 입력하지 않으면 모든 로그를 대상으로 데이터를 생성합니다.	optional
제외할 상태 코드	통계 데이터 생성 시 제외할 상태 코드입니다. 입력하지 않으면 4xx~5xx에 해당하는 전체 오류 상태 코드를 대상으로 4xx, 5xx건수 데이터를 생성합니다.	optional

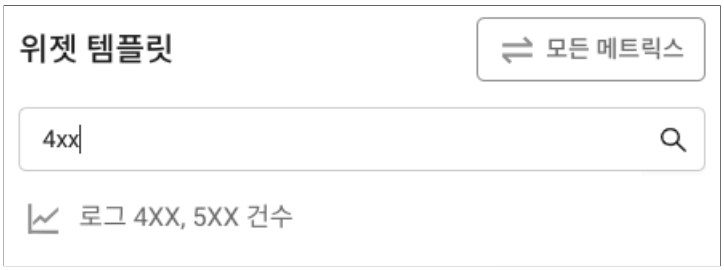
status 파서 등록 예시



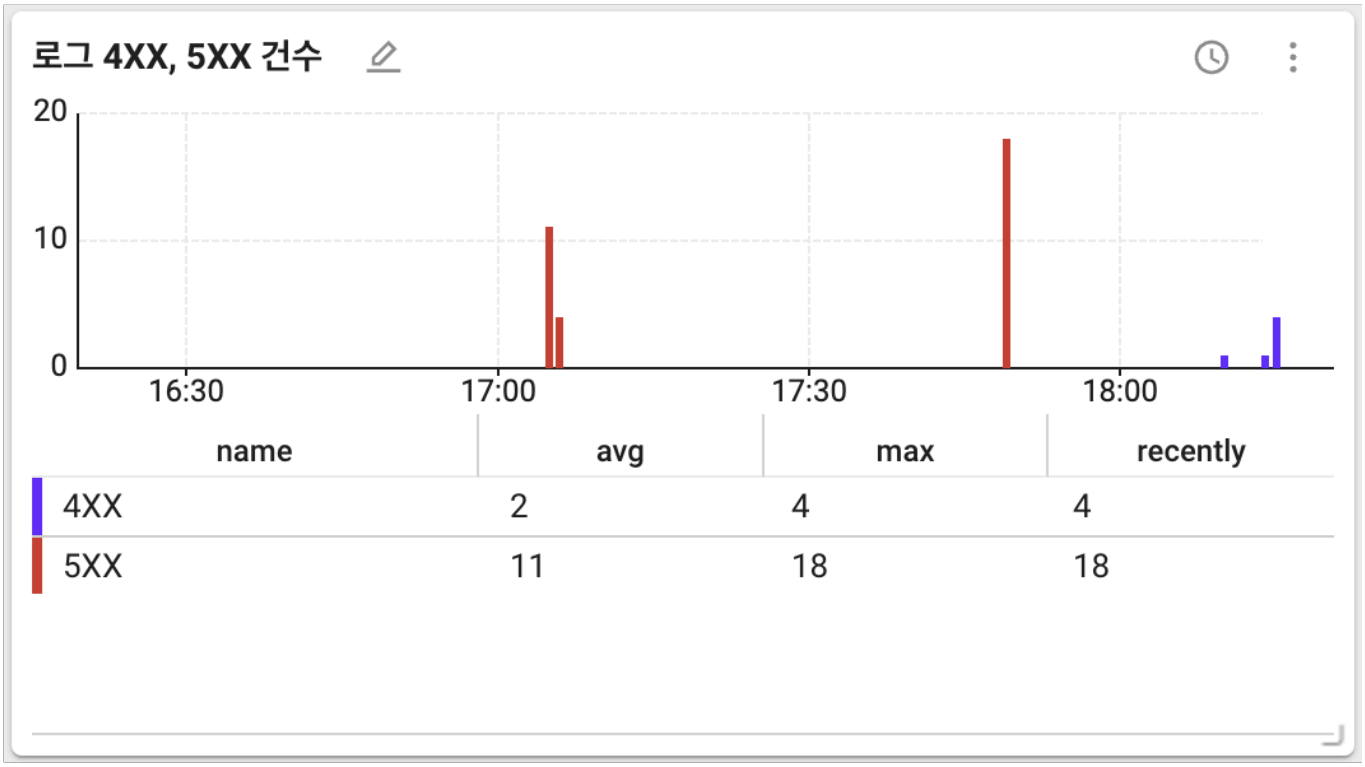
유입되는 로그가 `{"msg":"message","status":404}` 이고 예시처럼 GROK 파서로 status를 파싱한다면, `status: 404` 와 같이 파싱됩니다. status가 정상적으로 파싱되는 것을 확인했다면 4XX,5XX 상태 코드 파서에서 제외할 상태 코드를 등록하세요.

데이터 조회

파서를 모두 등록하면 **통합 Flex 보드**로 이동해 **로그 4XX, 5XX 건수** 위젯을 생성하세요.



위젯을 생성하면 다음과 같이 데이터를 확인할 수 있습니다.



- **avg**: 조회 기간 데이터 평균값입니다.
- **max**: 조회 기간 데이터 중 최댓값입니다.
- **recently**: 조회 기간 데이터 중 마지막 값입니다.

상태 코드 성공률 파서 설정 항목

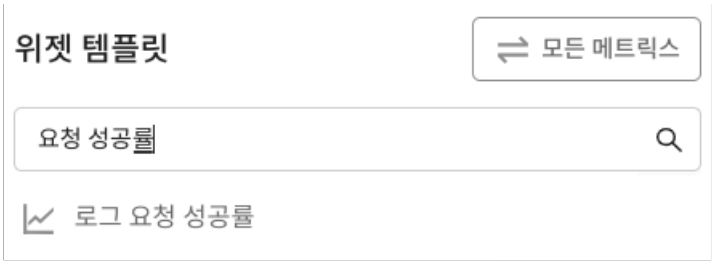
상태 코드 성공률 파서는 1차 파서로 파싱된 status 값을 기반으로 HTTP 요청 성공률 데이터를 생성합니다. 특정 상태 코드를 제외하려면 제외할 상태 코드를 입력하세요. status 파싱에 관한 내용은 4xx, 5xx 상태 코드 파서 설정 항목을 참조하세요.

설정 항목

설정 값	설명	기타
카테고리	요청 성공률 데이터를 생성할 카테고리입니다.	required
로그 검출 조건	필터로 적용할 검색 키, 검색 값을 입력합니다. 로그 검출 조건에 맞는 로그 데이터에 대해서만 요청 성공률 데이터를 생성합니다. 로그 검출 조건을 입력하지 않으면 모든 로그를 대상으로 데이터를 생성합니다.	optional
제외할 상태 코드	요청 성공률 데이터 생성 시 제외할 상태 코드입니다. 입력하지 않으면 2xx~3xx에 해당하는 전체 성공 상태 코드를 대상으로 요청 성공률 데이터를 생성합니다.	optional

데이터 조회

파서를 모두 등록하면 **통합 Flex 보드**로 이동해 로그 요청 성공률 위젯을 생성하세요.



위젯을 생성하면 다음과 같이 데이터를 확인할 수 있습니다.

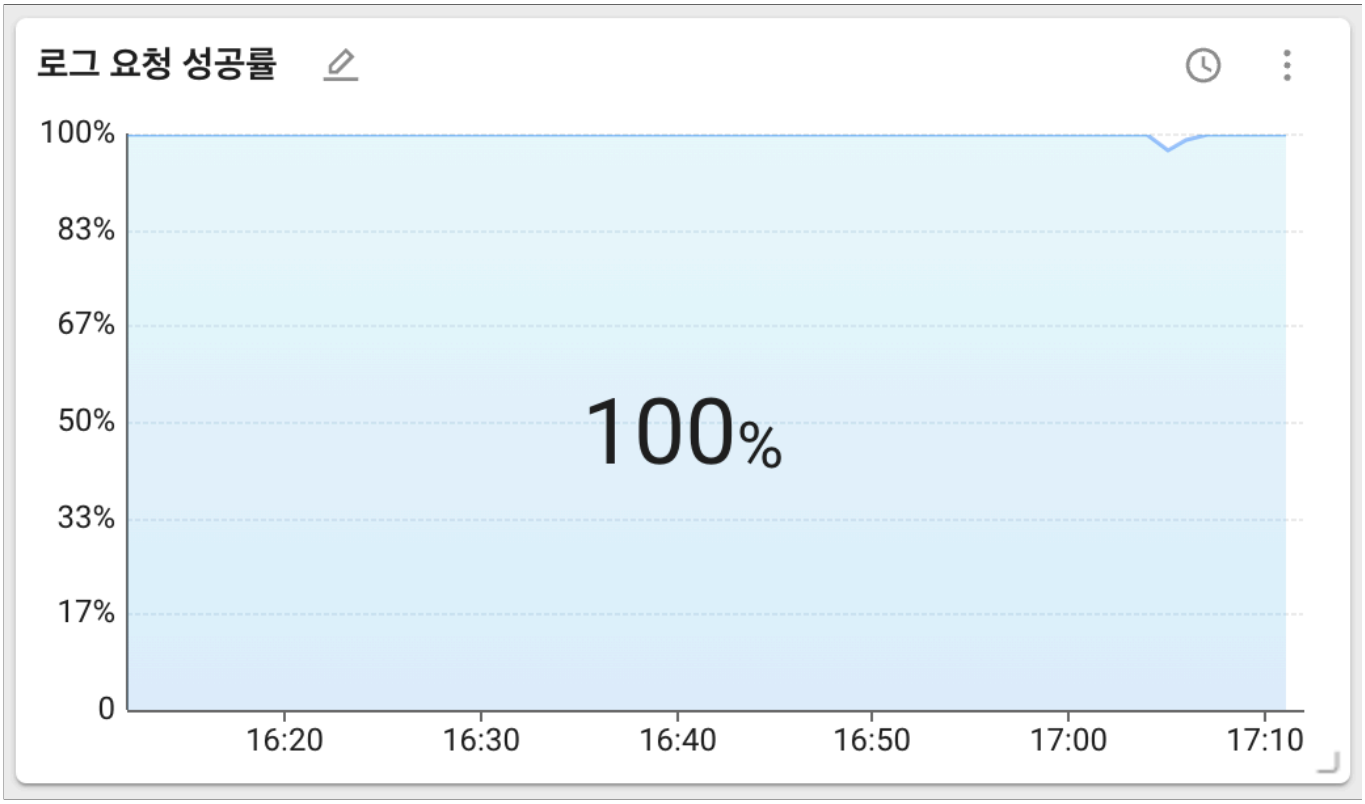


차트 위 데이터는 조회 기간에 대한 통계를 나타냅니다. 통계 방법을 최근값, 최댓값, 평균값 등으로 선택할 수 있습니다. 최근값이 기본으로 선택되어 있습니다.

2차 파서 프로세서

로그 2차 파서는 **4xx, 5xx 상태 코드 파서**와 **상태 코드 성공률 파서** 외에도 다음과 같은 필드 가공 프로세서를 제공합니다.

로그 2차 파서 프로세서 목록

프로세서	설명
모든 키워드 포함 필터	지정한 모든 키워드가 포함된 로그에 플래그 필드를 추가합니다.
일부 키워드 포함 필터	지정한 키워드 중 하나라도 포함된 로그에 플래그 필드를 추가합니다.
필드명 변경	기존 필드명을 다른 이름으로 변경합니다.

프로세서	설명
상태 코드 그룹 분류	HTTP 상태 코드를 2xx/3xx/4xx/5xx 그룹으로 자동 분류합니다.
URL 정적/동적 분류	URL에 확장자(`.`)가 있으면 정적 콘텐츠, 없으면 동적 서블릿으로 분류합니다.
URL 파라미터 제거	URL에서 `?` 이후의 쿼리 파라미터를 제거한 값을 저장합니다.
응답 시간 분류	응답 시간(ms)을 fast/normal/slow/critical 구간으로 자동 분류합니다.
JSON 필드 추출	JSON 문자열 필드에서 특정 키를 추출하여 새 필드로 저장합니다.
GeoIP 변환	IP 주소를 국가/도시 정보로 변환합니다.
직접 입력	FQN 클래스명으로 프로세서를 직접 등록합니다.

모든 키워드 포함 필터

지정한 **모든** 키워드가 포함된 로그에 플래그 필드를 추가합니다.

설정 값	설명	기본값
입력 필드	키워드를 검색할 필드명	`content`
저장 필드	결과를 저장할 필드명	`has_str_all`
keyword	검색할 키워드 목록 (쉼표로 구분해 입력)	required

활용 예시: 입력 필드를 `content`, 키워드를 `error, timeout, database`로 설정하면 세 키워드가 모두 포함된 로그에만 `has_str_all` 필드가 추가됩니다.

일부 키워드 포함 필터

지정한 키워드 중 **하나라도** 포함된 로그에 플래그 필드를 추가합니다.

설정 값	설명	기본값
입력 필드	키워드를 검색할 필드명	`content`
저장 필드	결과를 저장할 필드명	`has_str_any`
keyword	검색할 키워드 목록 (쉼표로 구분해 입력)	required

활용 예시: 입력 필드를 `content`, 키워드를 `reboot, shutdown, restart`로 설정하면 키워드 중 하나라도 포함된 로그에 `has_str_any` 필드가 추가됩니다.

필드명 변경

기존 필드명을 다른 이름으로 변경합니다. 대상 필드명이 이미 존재하면 덮어쓰지 않습니다.

설정 값	설명	기본값
원본 필드	변경할 기존 필드명	required
새 필드명	변경 후 필드명	required

상태 코드 그룹 분류

HTTP 상태 코드를 `status_2xx`, `status_3xx`, `status_4xx`, `status_5xx` 그룹으로 자동 분류합니다.

설정 값	설명	기본값
입력 필드	상태 코드가 담긴 필드명	`status`
저장 필드	그룹 분류 결과를 저장할 필드명	`status_nxx`

활용 예시: `status` 값이 `404`이면 `status_nxx` 필드에 `status_4xx`가 저장됩니다.

URL 정적/동적 분류

URL에 확장자(`.`)가 포함되어 있으면 **정적 콘텐츠**, 포함되지 않으면 **동적 서블릿**으로 분류합니다.

설정 값	설명	기본값
입력 필드	URL이 담긴 필드명	`url`
저장 필드	분류 결과를 저장할 필드명	`url_type`

URL 파라미터 제거

URL에서 `?` 이후의 쿼리 파라미터를 제거한 경로만 저장합니다.

설정 값	설명	기본값
입력 필드	URL이 담긴 필드명	`url`
저장 필드	파라미터 제거 결과를 저장할 필드명	`url_without_params`

활용 예시: `/api/v1/users?page=1&size=20` → `url_without_params` 필드에 `/api/v1/users`가 저장됩니다.

응답 시간 분류

응답 시간(ms)을 구간별로 자동 분류합니다. 구간 임계값을 직접 설정할 수 있습니다.

설정 값	설명	기본값
입력 필드	응답 시간(ms)이 담긴 필드명	`elapsed`
저장 필드	분류 결과를 저장할 필드명	`response_class`
Fast 상한(ms)	fast 구간 상한값	`100`
Normal 상한(ms)	normal 구간 상한값	`500`
Slow 상한(ms)	slow 구간 상한값	`2000`

기본값 기준 분류 결과는 다음과 같습니다.

응답 시간	분류
0 ~ 100ms	`fast`
101 ~ 500ms	`normal`
501 ~ 2000ms	`slow`
2001ms 이상	`critical`

활용 예시: 로그 탐색에서 `response_class: critical`로 필터링하면 응답 시간이 2초를 초과하는 로그만 조회할 수 있습니다.

JSON 필드 추출

1차 파서로 추출한 필드에 JSON 문자열이 포함된 경우 해당 JSON에서 특정 키를 추출하여 새 필드로 저장합니다.

설정 값	설명	기본값
입력 필드	JSON 문자열이 담긴 필드명	`payload`
추출할 키	추출할 JSON 키 목록 (쉼표로 구분해 입력)	required
저장 접두사	추출된 키 이름 앞에 붙일 접두사 (예: `json_`)	optional

- 로그 전체가 JSON인 경우 1차 JSON 파서가 자동으로 파싱합니다. 이 프로세서는 **GROK 1차 파서** 사용 시 텍스트 로그 안에 포함된 JSON 부분을 추가로 파싱할 때 유용합니다.
- 성능 보호를 위해 JSON 필드 크기가 10KB를 초과하면 파싱을 건너뜁니다.

활용 예시

1차 GROK 파싱 결과

```
payload = {"user":"alice","action":"purchase","product":"laptop","amount":45000}
```

입력 필드 `payload`, 추출할 키 `user, action`, 저장 접두사 `pay_` 설정 시 다음과 같이 추출됩니다.

JSON 필드 추출 결과

```
pay_user = alice
pay_action = purchase
```

GeoIP 변환

IP 주소를 국가 및 도시 정보로 변환합니다. IP2Location 데이터베이스를 활용해 접근 로그의 지역별 분석이 가능합니다.

설정 값	설명	기본값
입력 필드	IP 주소가 담긴 필드명	`client_ip`
저장 접두사	결과 필드 이름의 접두사	`geo`

결과로 `{접두사}_country`, `{접두사}_city` 두 개의 필드가 생성됩니다.

활용 예시: 입력 필드 `client_ip` 값이 `223.130.195.200`이고 저장 접두사가 `geo`인 경우:

GeoIP 변환 결과

```
geo_country = KR
geo_city = Seongnam
```

직접 입력

목록에 없는 프로세서를 FQN(Fully Qualified Name) 클래스명으로 직접 등록할 수 있습니다.

직접 입력 형식

```
io.whatap.logsink.processor.ClassName{"param1":"value1","param2":"value2"}
```

ⓘ 클래스명과 파라미터 형식이 정확해야 합니다. 잘못된 입력 시 프로세서가 동작하지 않을 수 있습니다.

2차 파서 체이닝

동일 카테고리에 여러 2차 파서를 등록해 체이닝할 수 있습니다. 각 프로세서는 등록 순서대로 실행됩니다.

원본 로그

```
2026-04-23T10:30:00.123+0900|INFO|api-01|192.168.1.10|/api/v1/orders?detail=true|200|1234|{"user":"alice","action":"purchase"}|Request processed
```

순서	파서	입력 → 출력
1차	GROK	timestamp, level, host, client_ip, url, status, elapsed, payload, content
2차-1	JSON 필드 추출	payload → user, action
2차-2	응답 시간 분류	elapsed → response_class (`slow`)
2차-3	상태 코드 그룹 분류	status → status_nxx (`status_2xx`)
2차-4	URL 파라미터 제거	url → url_without_params (`/api/v1/orders`)
2차-5	GeoIP 변환	client_ip → geo_country (`KR`), geo_city (`Seoul`)

ⓘ 2차 파서의 **카테고리**는 1차 파서와 동일하게 설정해야 합니다. 다른 카테고리로 설정하면 해당 로그에 2차 파서가 적용되지 않습니다.

로그 파서를 사용하면 불규칙한 형태의 로그를 쿼리가 가능한 구조화된 형태로 변경할 수 있습니다. 와탭 로그 모니터링은 다음과 같이 두 가지 유형의 파서를 제공합니다.

- **GROK 파서**: 임의의 형태로 수집되는 로그를 정규 표현식과 GROK 문법을 활용해 파싱합니다.
- **JSON 파서**: JSON 형태로 수집되는 로그를 파싱합니다.

ⓘ **공통 주의사항**

- 같은 카테고리에 여러 개의 파서가 등록되어 있는 경우 첫 번째로 매칭되는 파서만 적용됩니다.

- **와탭은 와탭 서비스의 안정성에 영향을 줄 수 있는 파서를 비활성화할 수 있는 권한을 가집니다.**

GROK 파서

로그가 불규칙한 형태로 수집되는 경우 GROK 파서를 사용해 로그를 파싱할 수 있습니다. GROK 문법은 named regular expressions를 제공해 정규 표현식을 보다 쉽고 편리하게 사용할 수 있습니다.

GROK 파서 패턴 등록에 관해 다음 동영상 가이드를 참조하세요.

GROK 시작하기

GROK은 두 가지 형태의 문법을 제공합니다.

1. `%{SYNTAX:SEMANTIC}` : GROK 라이브러리에서 제공하는 문법입니다. **named regular expressions**를 활용해 태그를 추출할 수 있습니다. 활용 예시는 다음을 참조하세요.
   - **SYNTAX**: GROK이 제공하는 named regular expressions를 지정합니다.
   - **SEMANTIC**: 매칭되는 값에 부여할 이름을 지정합니다.

   named regular expressions

   GROK에서 제공하는 문법입니다. 복잡한 정규 표현식에 이름을 부여해 사용할 수 있도록 GROK에서 제공하는 기능입니다.

name	regular expression
WORD	`\b\w+\b`
SPACE	`\s*`
NOTSPACE	`\S+`
UUID	`[A-Fa-f0-9]{8}-(?:[A-Fa-f0-9]{4}-){3}[A-Fa-f0-9]{12}`

> ⓘ 와탭에서 제공하는 모든 named regular expressions 확인을 원한다면 다음 링크를 참조하세요.

2. `(?<SEMANTIC>REGX)` : 정규 표현식의 **named capturing group** 문법입니다. 정규 표현식을 활용해서 사용자의 의도에 맞게 태그를 추출할 수 있습니다. 활용 예시는 다음을 참조하세요.
   - **SEMANTIC**: 매칭되는 값에 부여할 이름을 지정합니다.
   - **REGX**: 매칭에 사용할 정규 표현식을 입력합니다.

> ⓘ **named capturing group**
>
> 정규 표현식에서 제공하는 문법입니다.
> - capturing group: 여러 개의 토큰을 하나로 묶어 하나의 매칭 단위로 사용하는 기능을 의미합니다.
> - named capturing group: capturing group에 이름을 부여한 것입니다.
> - 문자열 매칭 예시를 살펴보겠습니다. dev@whatap.io
>   - 예시 1 `(\w+)@(\w+\.\w+)`
>   - 예시 2 이메일 전체 매칭 및 username과 domain 추가 매칭 시 `(?<username>\w+)@(?<domain>\w+\.\w+)`

`%{SYNTAX:SEMANTIC}` 활용 예시

다음은 `%{SYNTAX:SEMANTIC}` 문법을 활용하는 예시입니다.

Sample log

```
[2023-08-08 02:02:30,101 GMT][INFO ][i.w.y.l.c.LogSinkDexScheduleThread.realProcess(159)] 8 VirtualLog 20230808 02:01:00.000 {area=4, city=5} 56ms
```

샘플 로그를 보고 각 단어가 의미하는 내용을 유추할 수 있습니다. 각 부분을 semantic한 단어로 치환 시 다음과 같이 표현할 수 있습니다.

semantic replace

```
[date][logLevel][caller] projectCode logCategory dexBuildStartTime {area=areaEnum, city=cityEnum} dexBuildElapsed
```

semantic한 단어 모두 정규 표현식으로 대체할 수 있습니다. GROK 파서를 사용하면 사전 정의된 named regular expressions를 활용할 수 있습니다. 여기서 사용된 `TIMESTAMP_ISO8601`, `LOGLEVEL`, `DATA` 는 GROK에서 제공하는 named regular expressions입니다. 이 값들은 각각 다음의 정규 표현식으로 대체되어 매칭됩니다.

- name: `TIMESTAMP_ISO8601`
  - regular expression: `%{YEAR}-%{MONTHNUM}-%{MONTHDAY}[T ]%{HOUR}:?%{MINUTE}(?::?%{SECOND})?%{ISO8601_TIMEZONE}?`
- name: `LOGLEVEL`
  - regular expression: `LOGLEVEL ([Aa]lert|ALERT|[Tt]race|TRACE|[Dd]ebug|DEBUG|[Nn]otice|NOTICE|[Ii]nfo|INFO|[Ww]arn?(?:ing)?|WARN?(?:ING)?`
- name: `DATA`
  - regular expression: `.*?`

GROK parsing pattern

```
\[%{TIMESTAMP_ISO8601:date}\sGMT\]\[%{LOGLEVEL:level}\s\]\[%{DATA:caller}\]
```

위와 같은 문법으로 파싱을 하면 다음과 같이 태그를 추출할 수 있습니다. 이렇게 GROK의 `%{SYNTAX:SEMANTIC}` 문법은 복잡하고 긴 정규 표현식을 쉽고 간결하게 적용할 수 있도록 도와주는 역할을 합니다.

Tag extraction

```
- date : 2023-08-08 02:02:30,101
- caller : i.w.y.l.c.LogSinkDexScheduleThread.realProcess(159)
- level : LEVEL
```

`(?<SEMANTIC>REGX)` 활용 예시

named regular expressions로 매칭되지 않는 부분은 `(?<SEMANTIC>REGX)` 패턴을 사용해서 파싱할 수 있습니다. 위의 샘플 로그에서 `%{SYNTAX:SEMANTIC}` 문법만으로 파싱되지 않는 영역은 다음과 같습니다.

Unparsed area

```
8 VirtualLog 20230808 02:01:00.000 {area=4, city=5} 56ms
```

해당 로그의 각 부분을 semantic한 단어로 치환 시 다음과 같이 표현할 수 있습니다.

semantic replace

```
projectCode logCategory dexBuildStartTime {area=areaEnum, city=cityEnum} dexBuildElapsed
```

이렇게 불규칙한 형태의 문자열은 다음과 같은 `(?<SEMANTIC>REGX)` 문법을 사용해 파싱할 수 있습니다.

샘플 로그 파싱 키워드별 매칭되는 정규 표현식

파싱 키워드	(?<SEMANTIC>REGX)
8	(?<projectCode>\d)
VirtualLog	(?<logCategory>\w*)
20230808 02:01:00.000	(?<dexBuildStartTime>\d{8}\s\d{2}:\d{2}:\d{2}\.\d{3})
area=4	area=(?<areaEnum>\d)
city=5	city=(?<cityEnum>\d)
56ms	(?<dexBuildElapsed>\d{2})ms

기본 정규 표현식 문법

문법	의미	별칭
?	0 or 1	-
+	1 or more	-
*	0 or more	-
a{5}	exactly 5	-
\w	word character	[a-zA-Z_0-9]

문법	의미	별칭
\s	white space	-
.	any character except newline	-
[abc]	any of	-
[^abc]	not a,b, or c	-
[a-z]	character between a and z	-
[1-3[7-9]]	union (combining two or more character classes)	-
[1-6&&[3-9]]	intersection (교집합)	-
[0-9&&[^2468]]	subtraction (차집합)	-
a{2,}	2 or more	-
a{1,3}	between 1 and 3	-
a+?	match as few as possible	-
{2,3}?	match as few as possible	-
(abc)	capturing group (여러 개의 문자열을 single unit으로 처리함)	-
\d	digit	[0-9]
\D	non-digit	[^0-9]
\W	non-word character	-
\S	non-white space	-

이렇게 파싱된 키워드를 띄어쓰기(`\s`)와 특수 문자 escape(`\{` , `\,` , `\}`)로 연결하면 다음과 같이 패턴을 적용할 수 있습니다.

GROK parsing pattern

```
(?<projectCode>\d)\s(?<logCategory>\w*)\s(?<dexBuildStartTime>\d{8}\s\
d{2}:\d{2}:\d{2}\.\d{3})\s\{area=(?<areaEnum>\d),\scity=(?<cityEnum>\d)\}\s(?<dexBuildElapsed>\d{2})ms
```

위와 같은 문법으로 파싱을 하면 다음과 같이 태그를 추출할 수 있습니다.

Tag extraction

```
- projectCode : 8
- logCategory : VirtualLog
- dexBuildStartTime : 20230808 02:01:00.000
- areaEnum : 4
- cityEnum : 5
- dexBuildElapsed : 56
```

GROK 적용하기

로그 설정 > 로그 1차 파서 설정

1. GROK 패턴 파서를 적용하려면 **로그 설정** 메뉴의 **로그 1차 파서 설정** 탭으로 이동하세요.
2. **+ 추가하기**를 선택 후 **파서** 입력란에서 **GROK** 파서를 선택하세요.
3. **패턴 등록** 버튼 선택 시 오른쪽에 패턴 등록 및 시뮬레이션 창이 나타납니다.
4. **패턴**과 **로그** 입력 후 **시뮬레이션** 버튼을 클릭하여 적용하려는 패턴으로 파싱에 성공하는지 확인하세요.

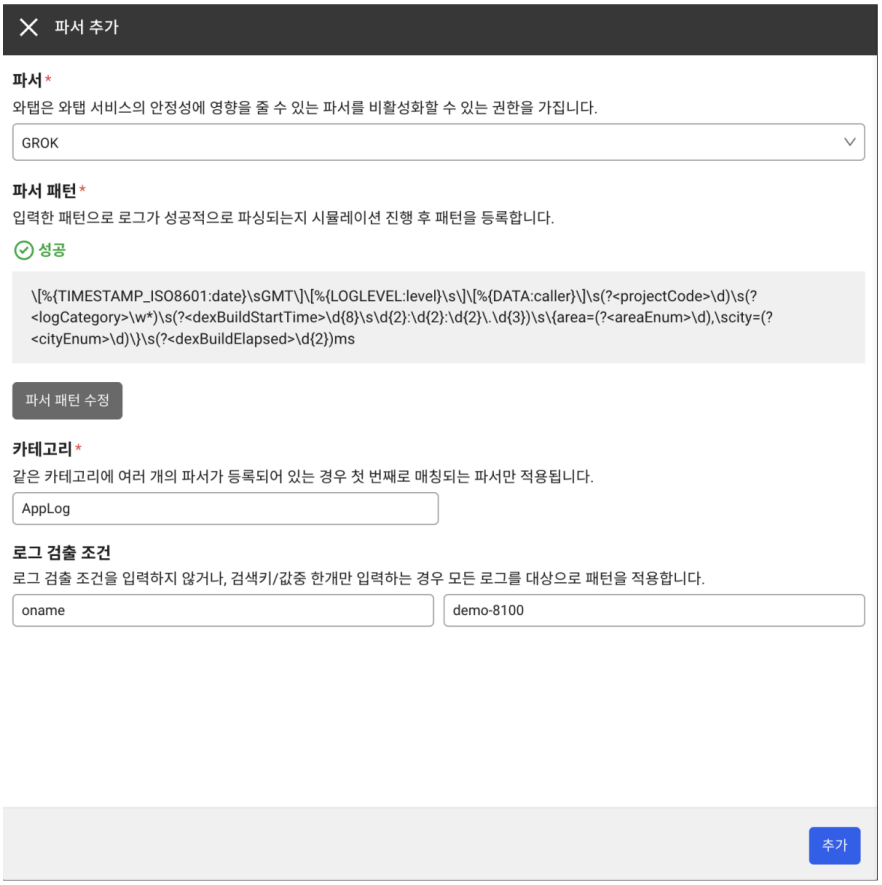
   패턴 예시:

   ```
   [%{TIMESTAMP_ISO8601:date}\sGMT\][%{LOGLEVEL:level}\s\][%{DATA:caller}\]\s(?<projectCode>\d)\s(?<logCateg
   d{2}:\d{2}:\d{2}\.\d{3})\s\{area=(?<areaEnum>\d),\scity=(?<cityEnum>\d)\}\s(?<dexBuildElapsed>\d{2})ms
   ```

   로그 예시:

   ```
   [2023-08-08 02:02:30,101 GMT][INFO ][i.w.y.l.c.LogSinkDexScheduleThread.realProcess(159)] 8
   VirtualLog 20230808 02:01:00.000 {area=4, city=5} 56ms
   ```

5. 시뮬레이션 성공 시 **시뮬레이션 결과** 및 **퍼포먼스 측정 결과**를 조회할 수 있습니다.
6. 시뮬레이션 후 **패턴 적용** 버튼 클릭 시 선택한 파서에 입력한 패턴이 적용됩니다.
7. 패턴 적용 후 **카테고리** 및 **로그 검출 조건**, **패턴**을 입력하세요.



- **카테고리**

  로그 카테고리를 선택하세요. **카테고리**는 필수로 입력해야 합니다.

- **로그 검출 조건**
  - 조건에 만족하는 로그만 파서가 적용됩니다.
  - **검색 키**와 **검색 값**을 선택하거나 직접 입력하세요.
  - **로그 검출 조건**은 모든 파서가 수행되기 전에 적용됩니다. 즉 파서의 결과로 추가되는 **태그**를 사용할 수 없습니다.

8. **추가** 버튼을 선택해 파서를 등록하세요.

> - 로그 파서 목록에서 해당 파서의 **적용 순서**를 변경하거나 **활성화** 및 **수정**, **삭제**할 수 있습니다.
> - **파서 시뮬레이션** 후 패턴을 등록할 수 있습니다.

GROK 파서 주의사항

- GROK 파서는 `%{SYNTAX:SEMANTIC}` , `%{SYNTAX:SEMANTIC}` 두 가지 패턴을 지원합니다.
- `%{SYNTAX:SEMANTIC}` 패턴 사용 시 `SEMANTIC` 을 반드시 입력해야합니다.
- `%{SYNTAX:SEMANTIC}` 패턴 사용 시 `SEMANTIC` 은 하나의 파서 안에서 unique 해야합니다.
- `(?<SEMANTIC>REGX)` 패턴 사용 시 `SEMANTIC` 은 문자(a-z, A-Z)와 숫자(0-9) 그리고 지정된 특수문자(`.` , `_` , `-`)만 올 수 있습니다.
- `SEMANTIC` 은 문자(a-z, A-Z)로 시작해야 합니다.
- `SEMANTIC` 은 문자(a-z, A-Z) 또는 숫자(0-9)로 끝나야 합니다.

GROK 추가 패턴 선택하기

GROK 파서 시뮬레이션 시 기본 패턴(`WORD` , `NUMBER` , `IP` 등) 외에 특정 로그 형식에 맞는 **추가 패턴**을 선택할 수 있습니다. Java, Nginx, MySQL 등 32개 패턴 타입이 11개 카테고리로 그룹핑되어 제공됩니다.

추가 패턴 카테고리 드롭다운

1. 파서 시뮬레이션 창에서 **추가 패턴 (선택)** 영역의 드롭다운을 클릭하세요.
2. 카테고리별로 그룹핑된 패턴 타입 목록에서 필요한 패턴 타입을 선택하세요. 검색 입력란에 패턴 타입 이름 또는 설명을 입력해 검색할 수도 있습니다.
3. 선택한 패턴 타입은 시뮬레이션 시 기본 패턴과 함께 로드됩니다.

기본 패턴(Core)은 항상 자동으로 포함됩니다. 추가 패턴 선택은 선택 사항입니다.

카테고리별 패턴 타입 목록

카테고리	패턴 타입
Web Servers	haproxy, httpd, iis, nginx, squid
Application Servers & Frameworks	java, log4j, rails, ruby, springboot

카테고리	패턴 타입
Databases	mongodb, mysql, oracle, postgresql, redis
Cloud	aws
System & Infrastructure	docker, kubernetes, linux-syslog
Messaging & Search	elasticsearch, kafka
Network & Security	bind, bro, firewalls, junos, zeek
Mail	exim, postfix
Monitoring & Backup	bacula, mcollective, nagios
Build Tools	maven

활용 예시

Tomcat Catalina 8 로그를 파싱하려면 **java** 패턴 타입을 추가로 선택한 뒤 `%{CATALINA8_LOG}` 패턴을 사용하세요.

Sample log

```
31-Jul-2020 16:40:38 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Server version name: Apache Tomcat/8.5.57
```

GROK parsing pattern

```
%{CATALINA8_LOG}
```

Tag extraction

```
- log_timestamp : 31-Jul-2020 16:40:38
- log.level : INFO
- java.log.origin.thread.name : main
- java.log.origin.class.name : org.apache.catalina.startup.VersionLoggerListener
- log.origin.function : log
```

```
- log_message : Server version name: Apache Tomcat/8.5.57
```

패턴 상세 보기

선택한 추가 패턴 타입에 어떤 패턴이 포함되어 있는지 상세 창에서 확인할 수 있습니다. 패턴 정의(정규 표현식)와 샘플 로그를 조회하고 시뮬레이션에 바로 적용할 수 있습니다.

패턴 상세 보기

추가 패턴 타입을 선택한 뒤 선택된 패턴 타입 태그를 클릭하면 오른쪽에 상세 창이 열립니다. 상세 창은 시뮬레이션 폼을 가리지 않고 나란히 표시되므로, 패턴 정의를 확인하면서 시뮬레이션을 진행할 수 있습니다.

각 패턴 항목에서 다음 기능을 사용할 수 있습니다.

버튼	기능	설명
샘플	샘플 로그 입력	해당 패턴에 매칭되는 예시 로그를 **시뮬레이션 파싱 로그** 입력란에 자동으로 입력합니다.
선택	패턴 삽입	`%{패턴이름}` 형식으로 **패턴** 입력란에 삽입합니다. 해당 패턴 타입도 자동으로 선택됩니다.
복사	클립보드 복사	`%{패턴이름}` 형식을 클립보드에 복사합니다.

- 패턴 항목을 클릭하면 패턴 정의(정규 표현식)가 펼쳐집니다. 여러 패턴을 동시에 펼쳐 정의를 비교할 수 있습니다.
- 패턴 정의에 포함된 `%{PATTERN:field}` 참조는 패턴 이름, 필드명, 리터럴 문자가 서로 다른 색상으로 구분되어 표시됩니다.
- **샘플** 버튼으로 입력된 로그로 바로 **시뮬레이션**을 실행해 파싱 결과를 확인할 수 있습니다.

시뮬레이션 샘플 로그 저장

시뮬레이션에 사용한 로그는 파서 설정과 함께 자동으로 저장됩니다. 파서를 수정할 때 이전에 사용한 샘플 로그가 시뮬레이션 창에 자동으로 복원되어 매번 로그를 다시 입력할 필요가 없습니다.

JSON 파서

로그가 JSON 포맷으로 수집될 경우 JSON 파서를 사용해 쉽고 편리하게 파싱할 수 있습니다.

JSON 적용하기

로그 설정 > 로그 1차 파서 설정

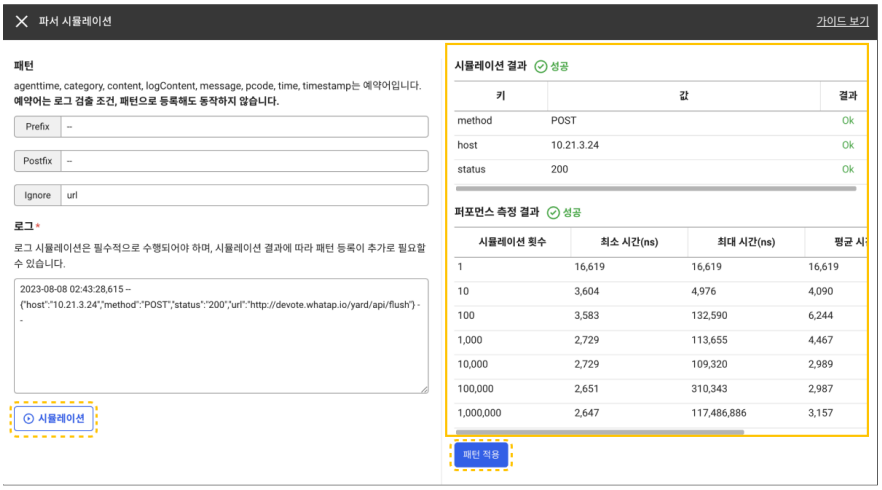
1. JSON 패턴 파서를 적용하려면 **로그 설정** 메뉴의 **로그 1차 파서 설정** 탭으로 이동하세요.
2. **+ 추가하기**를 선택 후 **파서** 입력란에서 **JSON** 파서를 선택하세요.
3. **패턴 등록** 버튼 선택 시 오른쪽에 패턴 등록 및 시뮬레이션 창이 나타납니다.
4. **패턴**과 **로그** 입력 후 **시뮬레이션** 버튼을 클릭하여 적용하려는 패턴으로 파싱에 성공하는지 확인하세요.

   Example

   ```
   2023-08-08 02:43:28,615 -- {"host":"10.21.3.24","method":"POST","status":"200","url":"http://dev.whatap.io/yard/api/flush"} --
   ```

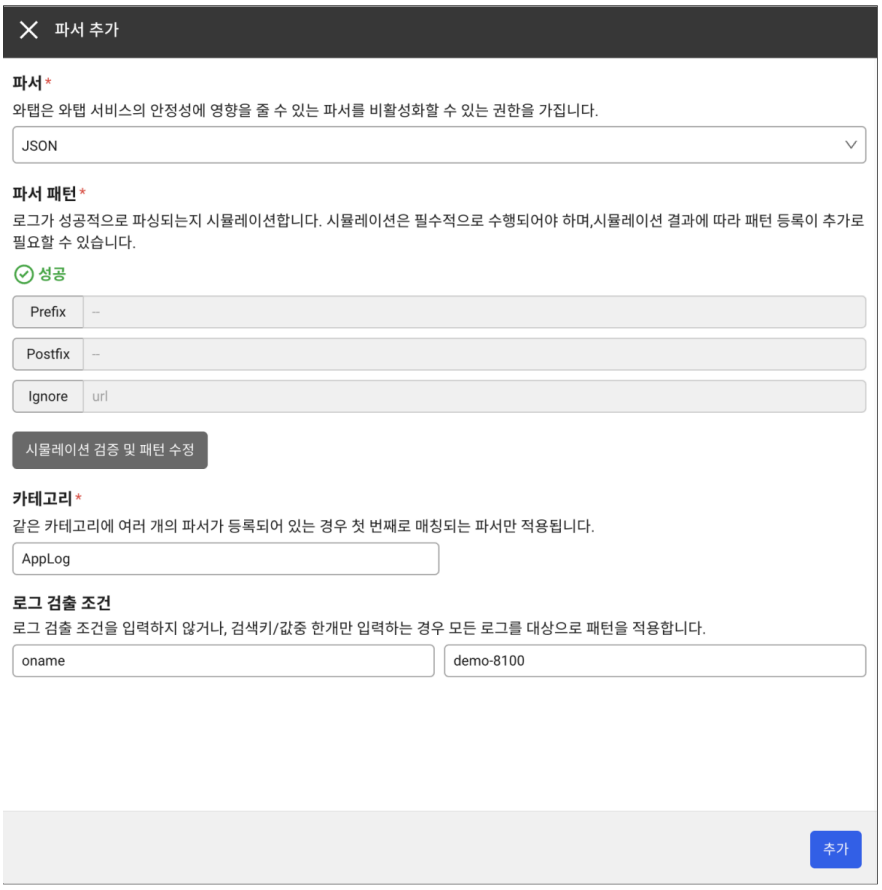
   예시 로그에서 **Prefix**와 **Postfix**를 `--` 로 지정하고 **Ignore**에 `url` 을 지정 시 `host` , `method` , `status` 3개의 **태그**만 생성됩니다.

   - **Prefix**
     로그에서 JSON 포맷이 시작하는 위치를 지정합니다. 로그 전체가 JSON 포맷인 경우 빈 값으로 설정합니다.
   - **Postfix**
     로그에서 JSON 포맷이 끝나는 위치를 지정합니다. 로그 전체가 JSON 포맷인 경우 빈 값으로 설정합니다.
   - **Ignore**
     JSON 포맷 중 태그를 생성하지 않을 키를 지정합니다.
5. 시뮬레이션 성공 시 **시뮬레이션 결과** 및 **퍼포먼스 측정 결과**를 조회할 수 있습니다.



6. 시뮬레이션 후 **패턴 적용** 버튼 클릭 시 선택한 파서에 입력한 패턴이 적용됩니다.

7. 패턴 적용 후 **카테고리** 및 **로그 검출 조건**, **패턴**을 입력하세요.



- **카테고리**

  로그 카테고리를 선택하세요. **카테고리**는 필수로 입력해야 합니다.

- **로그 검출 조건**
  - 조건에 만족하는 로그만 파서가 적용됩니다.
  - **검색 키**와 **검색 값**을 선택하거나 직접 입력하세요.
  - **로그 검출 조건**은 모든 파서가 수행되기 전에 적용됩니다. 즉 파서의 결과로 추가되는 **태그**를 사용할 수 없습니다.

8. **추가** 버튼을 선택해 파서를 등록하세요.

> - 로그 파서 목록에서 해당 파서의 **적용 순서**를 변경하거나 **활성화** 및 **수정**, **삭제**할 수 있습니다.
> - 파서를 등록하기 전에 **시뮬레이션**을 통해 등록하려는 패턴이 정상적인지 확인할 수 있습니다. GROK 파서 등록 시뮬레이션 과정과 동일합니다. 다음 문서를 참조하세요.

JSON 포맷 활용 예시

Sample log

```
{"host":"10.21.3.24","method":"POST","status":"200","url":"http://dev.whatap.io/yard/api/flush"}
```

위와 같은 샘플 로그가 수집된 경우 **파서 추가** 창에서 **JSON** 파서를 선택하세요. 복잡한 파싱 로직을 작성할 필요없이 로그 분석에 필요한 **태그**를 다음과 같이 추출할 수 있습니다.

Tag extraction

```
- host : 10.21.3.24
- method : POST
- status : 200
- url : http://dev.whatap.io/yard/api/flush
```

JSON 포맷 일부 구성 시 활용 예시

Some JSON format sample log

```
2023-08-08 02:43:28,615 -- {"host":"10.21.3.24","method":"POST","status":"200","url":"http://dev.whatap.io/yard/api/flush"} --
```

만약 예시와 같이 로그의 일부만 JSON 포맷으로 구성되어있다면 **Prefix**와 **Postfix**를 지정해 주세요. 와탭 로그 모니터링은 **Prefix**와 **Postfix** 사이의 영역을 JSON 포맷으로 인식 후 파싱합니다.

Tag extraction

```
- host : 10.21.3.24
- method : POST
- status : 200
- url : http://dev.whatap.io/yard/api/flush
```

로그 빠른 인덱스 설정

빠른 인덱스 설정

로그 설정 메뉴 상단에서 **빠른 인덱스 설정** 탭을 선택하세요. 대량의 로그를 수집할 경우 로그 검색 성능이 현저하게 저하될 수 있습니다. 자주 사용하는 검색 조건을 **인덱스**(index)로 생성하면 로그 검색 성능을 개선해 빠른 탐색이 가능합니다. 설정 항목은 다음과 같습니다.

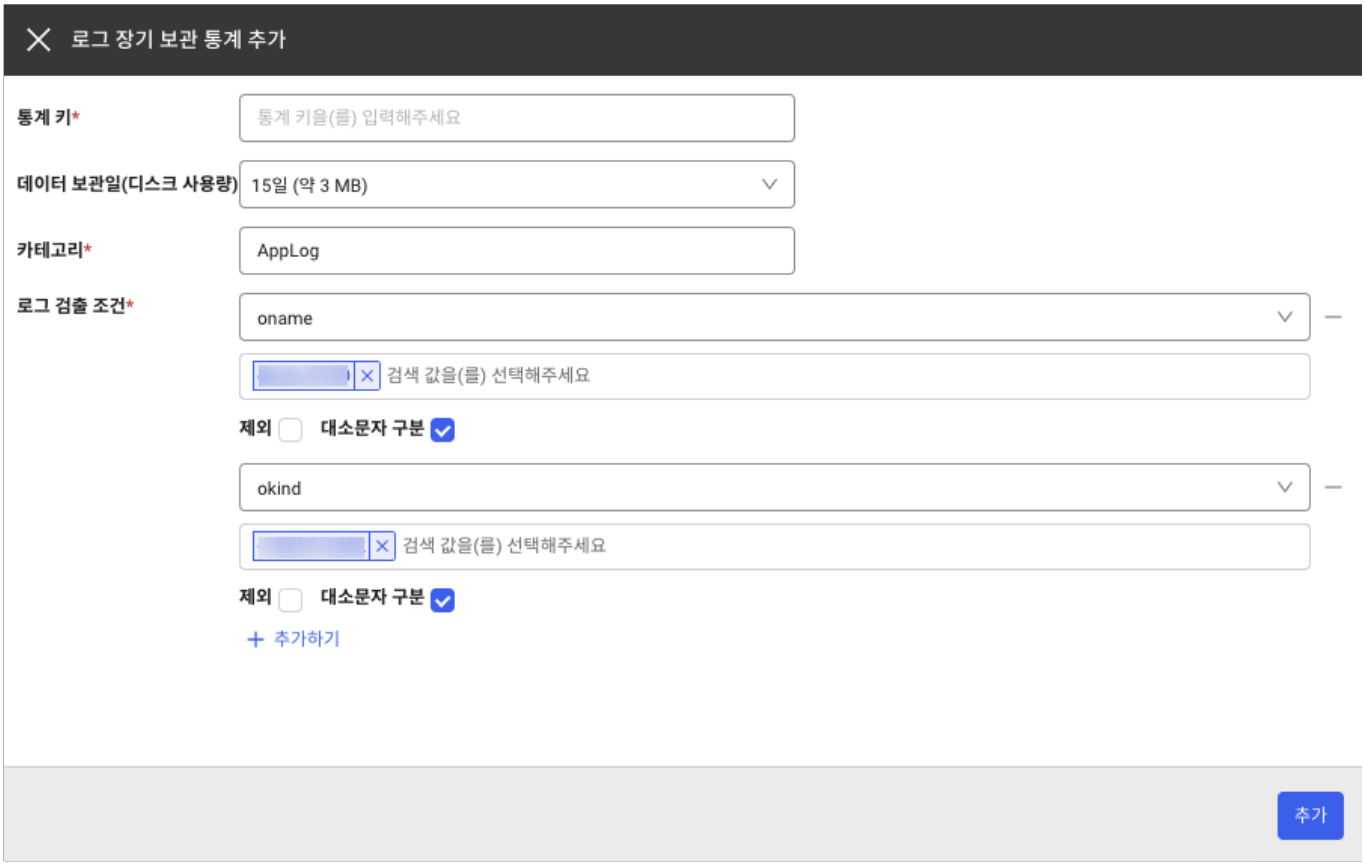
설정 값	필수 여부	설명
카테고리	필수	빠른 인덱스를 설정할 카테고리
검색 키	필수	빠른 인덱스를 설정할 검색 키
대소문자 구분 안 함	옵션	대소문자 구분 여부
규칙	필수	`*` 한 개 이상 포함 필수
활성	필수	활성 또는 비활성 여부(기본값 `true`)

로그 장기 보관 통계 설정

로그 장기 보관 통계 설정

로그 설정 메뉴 상단에서 **로그 장기 보관 통계** 탭을 선택하세요. 로그 데이터는 용량이 매우 커서 장기간 보관하기가 어렵습니다. 따라서 로그 통계 데이터 설정 기능을 사용하여 **특정 조건을 만족하는 로그 데이터가 5분마다 몇 건씩 수집되었는지에 대한 정보를 저장**할 수 있습니다. 장기간 시 실제 로그 데이터는 삭제되어도 해당 조건을 만족하는 로그가 얼마나 수집되었는지 추이를 확인할 수 있습니다.

로그 장기 보관 통계 추가



로그 장기 보관 통계 탭에서 **+ 추가하기** 버튼을 선택하면 **로그 장기 보관 통계 추가** 창이 나타납니다. **+ 추가하기** 버튼을 통해

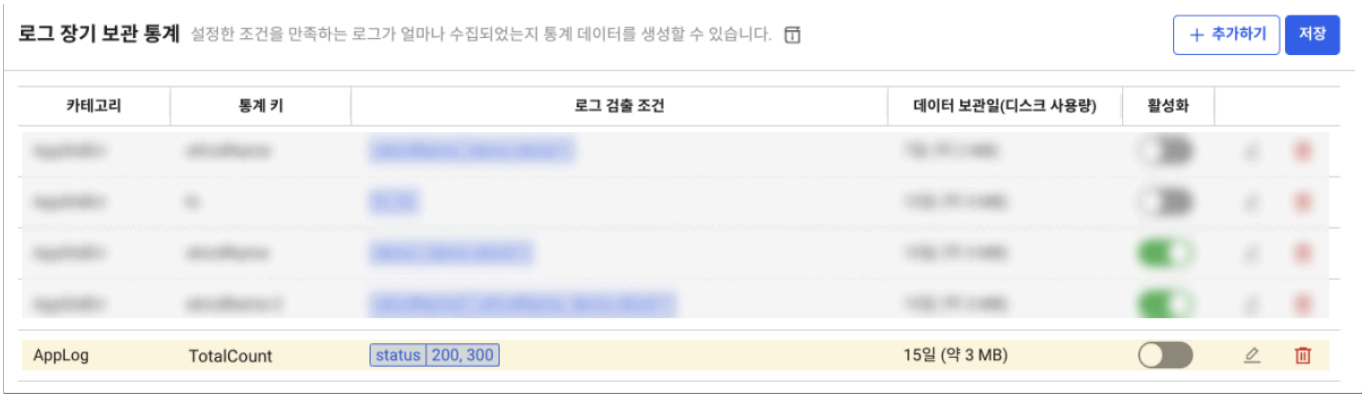
규칙을 추가하거나 생성한 규칙을 - 아이콘을 통해 삭제할 수 있습니다.

설정 항목

필드	설명
카테고리	규칙을 적용할 카테고리입니다.
통계 키(key)	규칙을 만족하는 로그 발생 시 저장할 키 값으로 동일한 키를 중복으로 설정할 수 없습니다.
로그 검출 조건	로그 통계 데이터를 생성할 조건입니다. 이 조건을 만족하는 로그가 얼마나 수집되었는지를 기반으로 통계 데이터를 생성합니다.
제외	제외를 체크하면 입력한 조건에 해당하지 않는 값으로 통계 데이터를 생성합니다.
대소문자 구분	입력한 로그 검출 조건의 값에 대해서 대소문자 구분 여부를 지정합니다.
활성	활성 또는 비활성 여부(기본값 `true`)

예시

다음과 같이 설정을 추가한 경우 수집된 로그 중 status가 `200` , `300` 인 로그를 대상으로 **TotalCount**라는 키값으로 통계 데이터를 생성합니다.

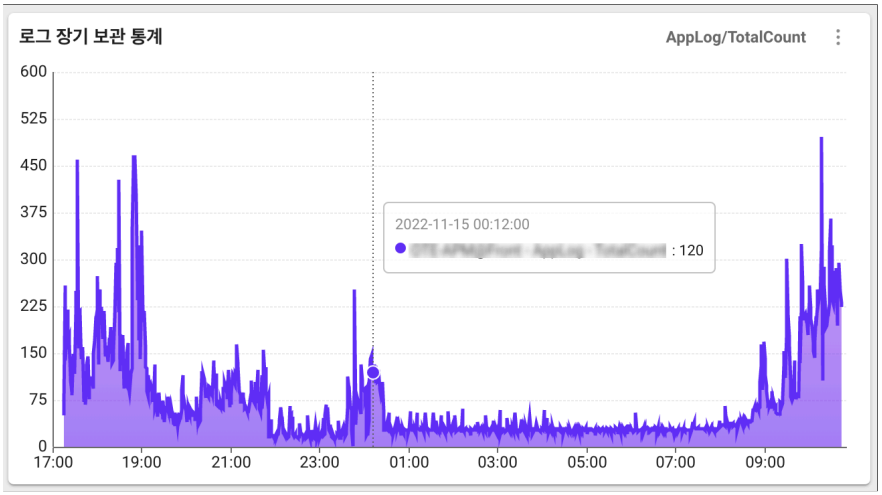


데이터 조회

1. **통합 Flex 보드**의 **위젯 템플릿**에서 **로그 장기 보관 통계**를 검색해 위젯을 생성하세요.

2. 데이터를 조회할 **카테고리**와 **통계 키**를 지정한 뒤 **적용** 버튼을 선택합니다.

3. 추가한 설정값으로 **로그 장기 보관 통계** 데이터를 다음처럼 확인할 수 있습니다.


로그 장기 보관 통계
AppLog/TotalCount
600
525
450
375
300
225
150
75
0
17:00
19:00
21:00
23:00
01:00
03:00
05:00
07:00
09:00
2022-11-15 00:12:00
: 120

로그 개인정보 비식별화 설정

개인정보 비식별화

개인정보 비식별화에서 로그 데이터 내 개인정보를 마스킹 처리하거나 안전한 값으로 치환할 수 있습니다.

- 암호화 저장: 지정된 민감 정보는 암호화되어 데이터베이스에 저장됩니다.
- 마스킹 처리: 로그 화면에 표시될 때 민감 정보는 기본적으로 마스킹(`**`) 처리되어 표시됩니다.
- 치환 처리: 민감한 정보를 사용자가 지정한 안전한 값으로 변경합니다.

개인정보 분류

일반적으로 개인정보로 분류하는 항목은 다음과 같습니다. 일반 개인정보 외에도 민감정보로 처리해야 하는 항목이 있다면 **개인정보 비식별화** 기능을 통해 로그 내에서 원하는 항목을 마스킹 처리 또는 치환 처리하세요.

- 주민등록번호
- 여권번호
- 운전면허번호
- 외국인 등록번호
- 신용카드 번호
- 계좌번호
- 생체인식 정보

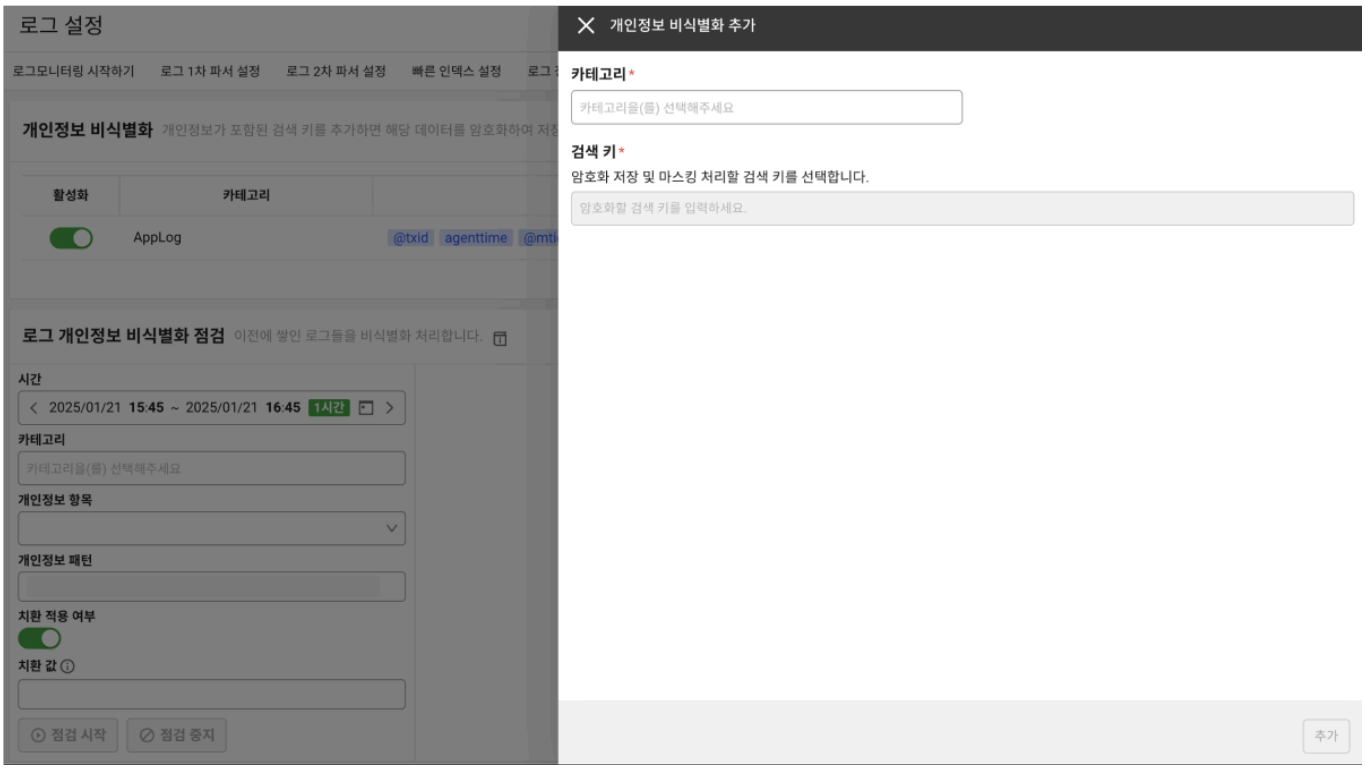
요구 권한

비식별화된 개인정보는 마스킹 처리되어 표시됩니다. **로그 개인정보 조회** 권한이 있는 사용자만 **개인정보 표시** 토글을 통해 원본 데이터를 확인할 수 있습니다. 비식별화 대상 검색 키의 추가 및 수정은 **로그 편집** 권한이 필요합니다.

> ⓘ **로그 개인정보 조회** 권한 및 **로그 편집** 권한에 대한 자세한 내용은 다음 문서를 확인하세요.

개인정보 비식별화 설정

신규 로그 데이터에 마스킹 처리를 적용하여 민감 정보를 보호할 수 있습니다.



1. **개인정보 비식별화** 탭에서 **추가하기** 버튼을 선택하세요.
2. 오른쪽 **개인정보 비식별화 추가** 창에서 **카테고리**를 선택하고, 마스킹 처리할 **검색 키**를 입력하세요.
3. **추가** 버튼을 클릭하여 설정을 추가하세요.
4. **개인정보 비식별화** 목록에서 비식별화 항목 확인 후 오른쪽 상단 **저장** 버튼을 눌러 현재 설정을 저장하세요.

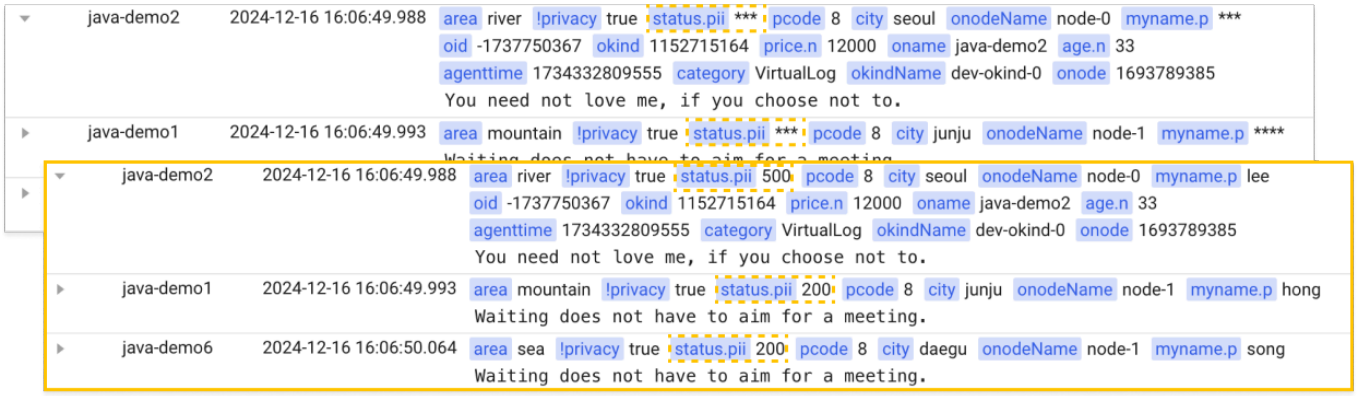
- 비식별화 처리된 검색 키의 경우 로그 콘텐츠 내 `검색 키.pii` 형식으로 출력됩니다.
- **개인정보 비식별화** 목록 내 **활성화** 컬럼에서 비식별화 항목을 **활성화**할 수 있습니다.

- **개인정보 비식별화** 목록 내 **편집** 컬럼에서 **수정** 및 **제거**할 수 있습니다.

> **로그 1차 파서 설정** 탭에서 검색 키 이름을 `검색 키.p` 로 지정하면 비식별화 대상으로 자동 등록됩니다. 자동 등록된 항목은 **개인정보 비식별화** 탭에서 **카테고리** 선택 후 **지정된 검색 키**에서 확인하거나 해제할 수 있습니다.
>
> 파서에서 `.p` 접미사로 지정하는 방식보다 **개인정보 비식별화** 탭에서 직접 설정하는 것을 권장합니다.

개인정보 표시

비식별화된 데이터는 **라이브 테일**, **로그 트렌드**, **로그 검색** 메뉴에서 마스킹 상태로 표시됩니다. **로그 개인정보 조회** 권한이 있는 사용자는 로그 목록의 **개인정보 표시** 버튼을 클릭해 원본 데이터를 확인할 수 있습니다.



> **개인정보 비식별화** 탭에서 지정한 검색 키는 로그 메뉴 내 로그 콘텐츠에서 `검색 키.pii` 형식으로 나타납니다.
>
> 예시, `status.pii` `**`

로그 개인정보 비식별화 점검

> **로그 개인정보 비식별화 점검** 기능을 통해 기존 데이터를 치환할 경우 해당 데이터는 원복이 불가능합니다.

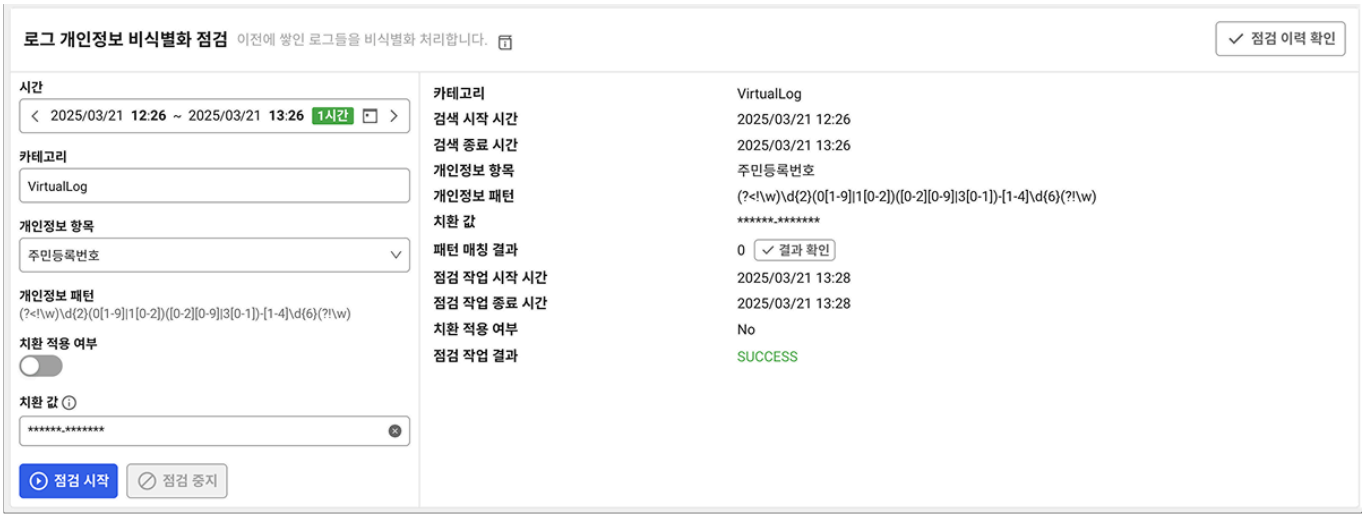
로그 개인정보 비식별화 점검는 기존 로그 데이터 내 개인정보를 식별하여 사용자가 지정한 **치환 값**으로 대체합니다.

대상 키 지정 치환

대상 키를 지정하면 해당 키에 저장된 값만 치환하고 나머지 로그 내용은 그대로 유지합니다. 대상 키는 로그의 `field` 값과 `tag` 값 모두에 적용됩니다. 대상 키를 지정하지 않으면 기존과 동일하게 로그 콘텐츠 전체에서 개인정보를 식별해 치환합니다.

> ⓘ 치환은 저장된 로그 데이터에만 적용됩니다. 검색 인덱스에는 치환 이후에도 원본 값이 남아 있을 수 있습니다.

대상 키는 로그 개인정보 비식별화 점검 폼의 **대상 키** 입력란에 지정하며, 화면에서는 `field` 와 `tag` 를 구분하지 않고 하나의 대상 키로 입력합니다.



1. 시작 시간 및 종료 시간을 설정하세요.
2. **카테고리**를 설정하세요.
3. **개인정보 항목**을 예시 중 선택 또는 **직접 입력**을 선택하세요.

   > ⓘ **직접 입력** 선택 시 개인정보를 식별하기 위한 정규표현식 패턴을 직접 입력합니다.

4. **치환 적용 여부** 토글을 통해 개인정보가 식별된 경우 이를 치환할지 여부를 선택하세요.
5. 기존 로그에서 정규표현식 패턴으로 식별된 문자열을 변경할 **치환 값**을 입력하세요.

> ⓘ
> - **치환 값**은 원본 값보다 길어서는 안됩니다.
> - **치환 적용 여부** 토글 활성화 시 반드시 **치환 값**을 입력해야 합니다.

6. 조건 입력 후 **점검 시작** 버튼을 선택하세요.

> ⓘ 점검 중인 경우 언제든지 **점검 중지** 버튼을 통해 점검 과정을 중지할 수 있습니다.

7. 치환 적용 시 **개인정보 비식별화 점검 시작** 확인 창에서 **치환하기** 입력 후 **확인** 버튼을 클릭하세요.

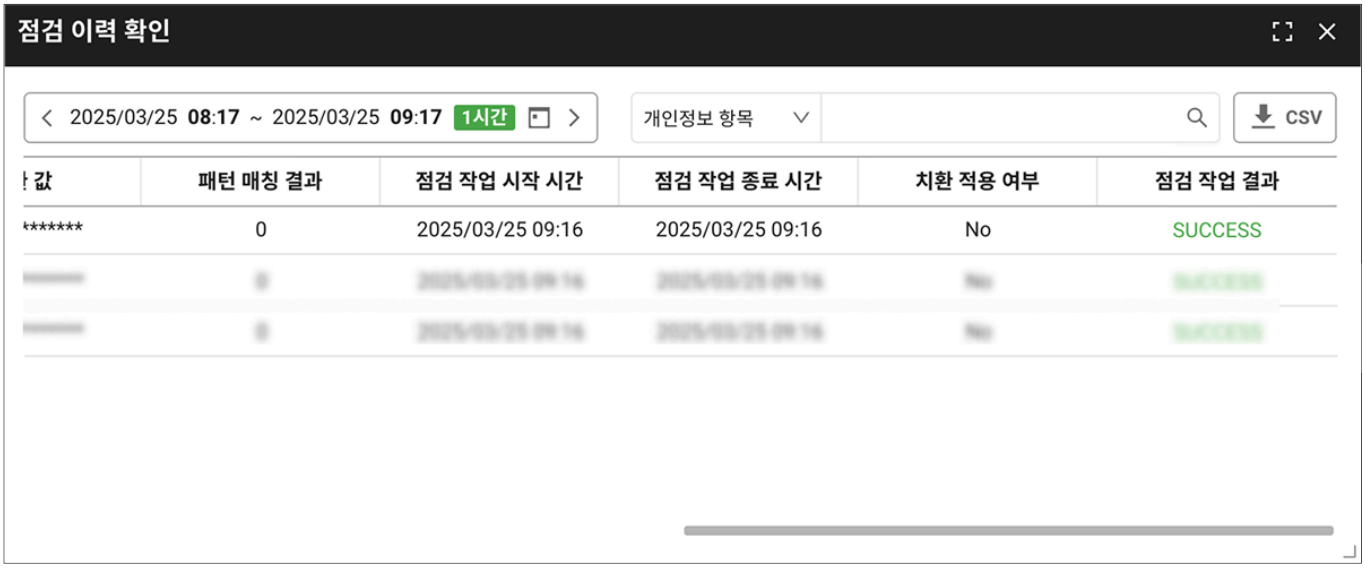
8. 점검 작업 완료 후 예시 이미지와 같이 오른쪽에서 **점검 작업 결과**를 확인할 수 있습니다.
   - **카테고리**: 점검된 카테고리 정보를 표시합니다.
   - **검색 시작 시간** 및 **검색 종료 시간**: 점검이 실행된 시간 범위를 표시합니다.
   - **개인정보 항목**: 점검된 개인정보 항목을 표시합니다.
   - **개인정보 패턴**: 사용된 정규표현식 패턴을 표시합니다.
   - **치환 값**: 적용된 치환 값을 표시합니다.
   - **패턴 매칭 결과**: 식별된 개인정보 개수를 표시합니다. **결과 확인** 클릭 시 샘플 로그를 확인할 수 있습니다.
   - **점검 작업 결과** : 개인정보가 식별된 결과를 표시합니다.

     `SUCCESS` (점검 성공), `FAILED` (점검 실패), `STOP` (점검 중지), `TIMED_OUT` (시간 초과로 점검 종료)

점검 이력 확인

로그 개인정보 비식별화 점검을 진행한 후 과거에 수행된 점검 결과를 조회하려면 상단 오른쪽에서 **점검 이력 확인** 버튼을

클릭하세요. 치환 내역, 점검 시점, 적용된 개인정보 항목 등을 추적할 수 있습니다. 또한 **점검 이력 확인** 창 상단 오른쪽에서 **CSV** 버튼 클릭 시 CSV 파일 형식으로 해당 목록 정보를 다운로드할 수 있습니다.



개인정보 패턴

기존 로그 데이터에서 개인정보를 식별할 때 사용되는 정규표현식 패턴 예시는 다음과 같습니다.

개인정보 패턴	개인정보 예시	정규식 패턴	치환 값
주민등록번호	123456-1234567	(?<!w)d{2}(0[1-9]|1[0-2])([0-2][0-9]|3[0-1])-[1-4]d{6}(?!w)	`****_*****`
여권번호	M1234567	([MSRODTC][0-9]{7}\b)	`******`
운전면허번호	01-23-456789-00	(?<!w)d{2}-d{2}-d{6}-d{2}(?!w)	`**_**_****_**`
외국인등록번호	987654-1234567	(?<!w)d{6}-?[5-8]d{6}(?!w)	`****_*****`
신용카드번호	1234-5678-9012-3456	(?<!w)d{4}-d{4}-d{4}-d{4}(?!w)	`***_***_***_***`

개인정보 패턴	개인정보 예시	정규식 패턴	치환 값
전화번호	010-1234-5678	(?<!w)(+82-?1[016789]-?d{3,4}-?d{4}|01[016789]-?d{3,4}-?d{4})(?!w)	`**-***-***`
이메일 주소	`example@domain.com`	([a-zA-Z0-9][a-zA-Z0-9._%+-]*@[a-zA-Z0-9.-]+.[a-zA-Z]{2,})	`**@**.**`
계좌번호	123456-01-654321	(?<!w)d{6}-d{2}-d{6}(?!w)	`****-**-****`
사업자등록번호	111-11-01111	(?<!w)d{3}-d{2}-d{5}(?!w)	`**-**-****`
법인등록번호	111111-1111111	(?<!w)d{6}-?d{7}(?!w)	`****-*****`

로그 설정 가져오기/내보내기

로그 설정 가져오기/내보내기

파서 설정, 빠른 인덱스 설정, 필터 설정 내용을 JSON 파일로 내보내고, 다른 프로젝트에서 가져와 적용할 수 있습니다.
프로젝트마다 반복해서 설정을 작성하는 번거로움을 줄일 수 있습니다.

1. 하나의 프로젝트에 파서 설정, 빠른 인덱스 설정, 필터 설정을 추가하세요.
2. 각 설정 탭에서 화면 오른쪽 위에 **JSON** 버튼을 선택하세요.
3. **JSON 내보내기** 창이 나타나면 화면 오른쪽 위에 **내보내기** 버튼을 선택하세요.
4. JSON 설정 파일이 사용자 PC에 저장됩니다.
5. 다른 프로젝트로 이동한 다음 **로그** > **로그 설정** 메뉴로 이동하세요.
6. 앞서 JSON 설정 파일을 내보낸 설정 탭을 선택한 다음 버튼을 선택하세요.
7. 파일 선택 창이 나타나면 사용자의 PC에 저장한 JSON 설정 파일을 선택하세요.
8. **JSON 가져오기** 창이 나타나면 설정 내용을 확인한 다음 **목록에 추가하기** 또는 **덮어쓰기** 버튼을 선택하세요.
9. 화면 오른쪽 위에 **저장** 버튼을 선택하세요.

> JSON 설정 파일을 가져온 다음 **저장** 버튼을 선택하지 않으면 가져온 설정 내용을 저장할 수 없습니다.

경고 알림

모니터링 대상 시스템 또는 애플리케이션에서 문제가 발생했을 때, 미리 설정한 조건에 따라 사용자에게 알림을 전달하는 기능입니다. 알림은 이메일, SMS, 메신저, 앱 푸시 등 다양한 채널로 전송됩니다.

예를 들어, CPU 사용률, 메모리 사용량, 디스크 I/O, 네트워크 트래픽, 트랜잭션 수, 오류 발생, 응답 시간 지연 등 주요 성능 지표에 알림 기준을 설정해두면, 해당 기준을 초과하거나 이상 징후가 감지될 때 알림이 전송됩니다.

와탭의 AI 기반 알림 기능을 사용하면 복잡한 조건을 직접 설정하지 않아도 됩니다. AI는 애플리케이션의 정상적인 동작 패턴을 학습해 이상 패턴을 자동으로 감지하여 알려줍니다. 태그 카운트 알림 기능을 활용하면 특정 태그 기반으로 집계된 값에 조건을 설정하여 복잡한 상황도 정밀하게 감지할 수 있습니다.

주요 기능

- **이벤트 설정**: 임계치 기반 알림 조건과 수신 채널 설정
- **이벤트 수신 설정**: 사용자별 수신 채널, 시간대 설정
- **이벤트 기록**: 알림 발생 기록 조회

권한

이벤트를 추가하거나 설정하려면 **알림 설정 권한**이 필요합니다.

- **조회**: 메트릭스 이벤트 설정 내용을 확인하려면 **조회 분석 권한**이 필요합니다.
- **추가/수정/삭제**: 설정을 변경하려면 **알림 설정 권한**이 필요합니다.
- **알림 수신**: 알림을 수신하려면 **알림 수신 권한**이 필요합니다.

> **참고 문서**
>
> 자세한 내용은 멤버 권한을 참고하시기 바랍니다.

이벤트 설정

홈 화면 > 프로젝트 선택 > **경고 알림** > **이벤트 설정**

이벤트 설정에서 모니터링 대상 시스템 또는 애플리케이션에서 발생할 수 있는 이상 징후를 기준으로 이벤트를 설정하고 관리하는 기능을 제공합니다. 설정된 조건이 충족되면 이메일, SMS, 메신저, 앱 푸시 등 원하는 채널로 알림을 받을 수 있습니다. 모니터링 목적에 따라 이벤트를 구성할 수 있습니다.

메트릭스

메트릭스 이벤트는 실시간 데이터를 기반으로 시스템 및 애플리케이션의 성능 지표를 모니터링할 수 있습니다.

실시간 로그

실시간으로 수집한 로그에서 설정한 이벤트 조건이 발견되면 경고 알림을 보냅니다.

복합 로그

로그 내용을 기준으로 경고 알림을 발생시키는 복합 로그 이벤트를 설정합니다.

참고

- 이벤트 설정 시 적정 임계값을 사용하여 불필요한 이벤트 발생을 방지하세요.
- 알림 설정 시 해소 알림(문제 해결 시 알림) 여부도 함께 고려하면 운영 대응에 도움됩니다.

공통 기능

모든 이벤트 설정을 **JSON 또는 Excel 형식으로 다운로드**할 수 있습니다. 같은 유형의 제품 설정을 공유하면 반복되는 이벤트 설정 작업을 줄일 수 있습니다.

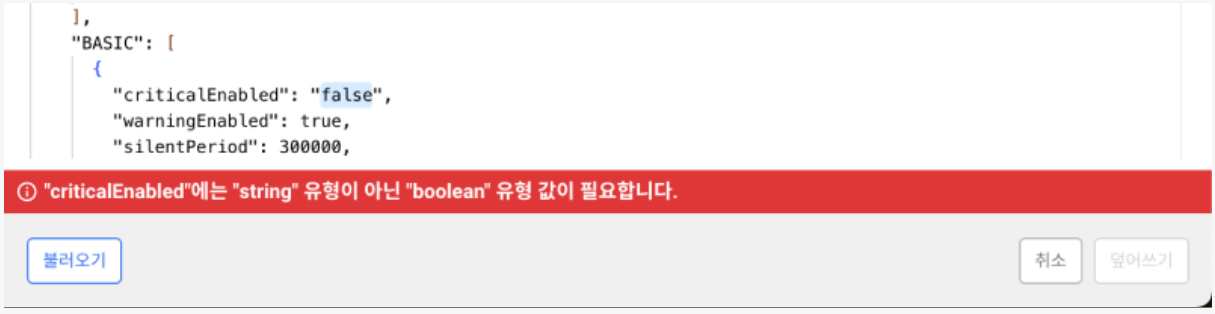
- **JSON 일괄 수정**: 모든 이벤트 조건을 JSON 형식으로 일괄 수정할 수 있습니다.
- **JSON 일괄 다운로드**: 현재 등록된 이벤트 조건을 JSON 파일로 저장할 수 있습니다.
- **Excel 일괄 다운로드**: 이벤트 목록을 Excel 파일로 내려받아 외부 보고나 백업에 활용할 수 있습니다.

JSON 일괄 수정

1. 화면 상단의 [JSON 일괄 수정] 버튼을 클릭하세요.
2. **JSON 일괄 다운로드** 기능을 통해 다운로드 받은 JSON 파일을 가져오거나 직접 수정하세요.
3. [덮어쓰기] 버튼을 클릭해 저장합니다.
4. 확인 메시지가 나타나면 [덮어쓰기] 버튼을 클릭하세요.

덮어쓰기 후 이전의 이벤트 설정으로 복구할 수 없습니다.

- `productType` 또는 `platform` 속성의 값을 임의로 수정하면 다른 프로젝트에서 이벤트가 정상 동작하지 않을 수 있습니다.
- JSON 편집창은 JSON 검증(Validation) 기능이 내장되어 있습니다. 형식이나 값이 올바르지 않은 경우, 편집창 아래에 오류 메시지가 표시되며 [덮어쓰기] 버튼은 비활성화됩니다.



JSON 데이터 구조

```
{
  "productType": "{productType}",
  "platform": "{platform}",
  "events": {
    "HITMAP": {
      ...
    },
    "LOG_REALTIME": [
      {...}
    ],
    "METRICS": [
      {...}
    ],
    "COMPOSITE_LOG": [
      {...}
    ],
    "BASIC": [
      {...}
    ],
    "COMPOSITE_METRICS": [
      {...}
    ],
    "ANOMALY": [
      {...}
    ]
  }
}
```

JSON 필드	설명
productType	제품 타입(예: APM, Infra 등)
platform	플랫폼 정보(예: java, node 등)
events	이벤트 데이터
ㄴ HITMAP	히트맵 패턴

JSON 필드	설명
ㄴ LOG_REALTIME	실시간 로그 이벤트
ㄴ METRICS	메트릭스
ㄴ COMPOSITE_LOG	복합 로그 이벤트
ㄴ BASIC	기본 이벤트
ㄴ COMPOSITE_METRICS	복합 메트릭스
ㄴ ANOMALY	이상치 탐지

JSON 일괄 다운로드

화면 상단의 [JSON 일괄 다운로드] 버튼을 클릭하세요.

- JSON 파일 이름은 integrated-event-`YYYYMMDD`.json 형식입니다.
- JSON 데이터의 구조와 `events` 속성의 하위 속성은 제품(`platform`) 또는 제품 유형(`productType`)에 따라 다를 수 있습니다.

JSON 데이터의 속성 중 `productType` 또는 `platform` 속성의 값을 임의로 수정하면 다른 프로젝트에서 이벤트가 동작하지 않을 수 있습니다.

Excel 일괄 다운로드

화면 상단의 [Excel 일괄 다운로드] 버튼을 클릭하세요.

메트릭스

홈 > 프로젝트 선택 > **경고 알림** > **이벤트 설정** > **메트릭스** 탭

프로젝트에 설정된 모든 메트릭스 기반 이벤트 목록을 확인할 수 있습니다. 사용자가 설정한 메트릭스 이벤트를 확인, 수정, 삭제하거나, 새로운 이벤트를 추가할 수 있습니다.

기본 옵션

- **이벤트 레벨**: 이벤트의 심각도를 의미하며 정상(Info), 경고(Warning), 위험(Critical) 세 단계로 구분
- **에이전트 중복 알림**: 설정한 시간(30초, 1분, 3분, 5분, 10분)에 같은 이름의 에이전트가 동시에 2개 이상 동작하는 경우, 선택한 시간 동안 이벤트 발생 안함
- **검색**: 이벤트 이름을 기준으로 검색
- **JSON 다운로드/업로드**: 이벤트 규칙은 JSON 편집기를 통해 바로 수정하거나, 파일로 다운로드하여 수정 후 다시 업로드 가능
- **+ 이벤트 추가**: 새로운 메트릭스 이벤트 추가
- **컬럼 크기 조정**: 각 컬럼의 경계를 드래그하여 너비 조정 가능

표 | 메트릭스 이벤트 목록 구성

항목	설명
No.	이벤트의 순번
	이벤트의 활성화 여부 – 활성화: 조건을 충족하면 이벤트 발생 – 비활성화: 이벤트가 동작하지 않으며, 조건을 검사하지 않음
	이벤트 설정 편집 및 삭제
이벤트 이름	사용자가 지정한 이벤트명 – 최대 255자까지 입력 가능
규칙	이벤트 발생 조건

항목	설명
대상	이벤트가 적용되는 모니터링 대상
발생 횟수	이벤트 발생 기준이 되는 기간과 횟수 – 선택한 시간 동안 설정한 횟수만큼 조건을 충족하면 이벤트 발생
일시 중지	이벤트 알림 발생 후, 같은 이벤트 발생을 일정 시간 동안 멈춤 – 선택 가능 시간: 사용 안 함, 5분, 10분, 15분, 20분, 30분, 1시간, 2시간, 3시간, 6시간, 12시간, 1일 – 해소된 알림이 활성화된 경우, 정상(RECOVERED) 상태 알림을 받은 후 선택한 시간 동안 같은 이벤트가 발생해도 알림이 발생하지 않음
해소된 알림	Critical과 Warning 레벨의 이벤트가 해소되면 RECOVERED(정상) 상태의 알림 수신 여부 – 활성화: 진행 중 → 정상 상태로 전환되면 RECOVERED 알림을 보내고, 이벤트 기록에서 진행 중/해소 상태를 구분해 볼 수 있음 – 비활성화: 임곗값을 초과할 때마다 단발성 이벤트 발생
이벤트 수신	해당 이벤트 알림을 받을 사용자 또는 그룹(수신 태그)

이벤트 설정

메트릭스 이벤트는 모니터링 대상에서 발생하는 이벤트를 조건에 따라 정의하고, 효율적으로 알림을 받을 수 있도록 설정할 수 있는 기능을 제공합니다.

이벤트 추가

메트릭스 이벤트를 추가하려면 **이벤트 설정** > **메트릭스** 탭으로 이동한 후, 오른쪽 상단의 [+ 이벤트 추가] 버튼을 클릭하세요.

메트릭스 이벤트 추가 방법
Step 1. 템플릿 선택: 이벤트의 기본 구조를 선택합니다.
Step 2. 이벤트 조건 정의: 이벤트를 어떤 기준으로 발생시킬지 조건을 설정합니다.

메트릭스 이벤트 추가 방법
Step 3. 이벤트 대상 선택: 이벤트가 발생하는 모니터링 대상을 선택합니다.
Step 4. 기본 정보 및 수신 설정: 이벤트 이름 및 메시지, 수신 대상을 설정합니다.

템플릿 선택

1. 이벤트를 어떻게 구성할지에 따라 화면 왼쪽의 3가지 템플릿 중 하나를 선택하세요.
    - **새로 만들기**: 카테고리와 지표를 처음부터 자유롭게 구성할 수 있으며, 복잡한 조건이나 세밀한 모니터링 규칙이 필요한 고급 사용자에게 적합
    - **빠른 설정**: 사전 정의된 카테고리, 지표, 기본 임계치가 제공되며, 임곗값만 조정하여 빠르게 적용할 수 있어 초급 사용자에게 적합
    - **고급 설정**: 사전 정의된 카테고리와 지표를 기반으로 조건을 추가하거나 자유롭게 변경

> ⓘ 메트릭스 이벤트에서는 다양한 설정 방식을 지원합니다. 모니터링 플랫폼에 따라 제공하는 템플릿은 다를 수 있습니다.

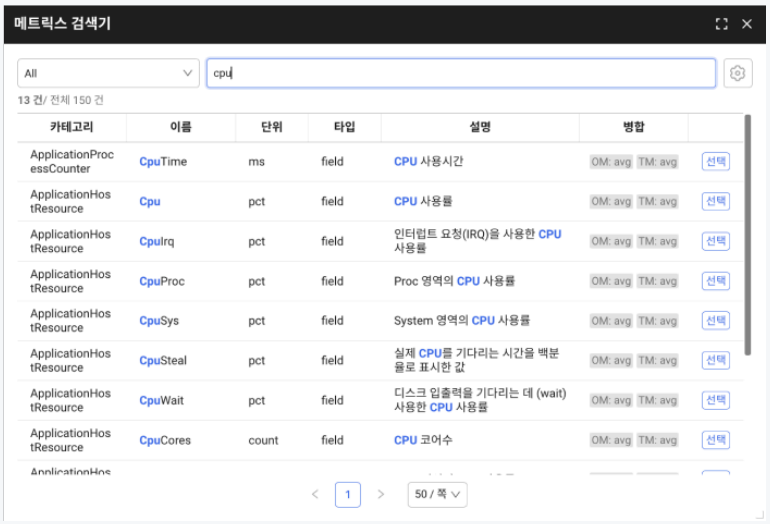
이벤트 조건 정의

이벤트 조건 정의 단계에서 이벤트가 언제 발생할지 결정하는 지표 설정, 발생 횟수, 일시 중지, 해소된 알림, 이벤트 동작 시간을 설정합니다.

지표 설정

1. 검색창에서 카테고리를 입력하거나, 카테고리 목록에서 카테고리(지표)를 선택하세요.
   카테고리를 선택해야 이후 지표 설정을 진행할 수 있습니다.
    - 카테고리 목록은 카테고리 이름, 데이터 수집 주기, 키를 확인할 수 있습니다. 최근 3시간 동안 프로젝트에서 수집된 메트릭스 데이터가 표시됩니다.
2. 목록에 없는 지표를 사용하려면 **직접 입력하기**를 선택하여 카테고리를 직접 입력하세요.

> **카테고리와 지표를 모르는 경우 메트릭스 검색기를 활용하세요.**
>
> 처음 지표를 설정하는 경우 메트릭스 검색기를 활용하면 더 빠르게 지표를 찾을 수 있습니다.
>
> - 메트릭스 검색기에서 원하는 지표를 검색하면 관련 카테고리와 상세 정보를 확인할 수 있습니다.
> - 검색 결과에서 필요한 메트릭스의 [선택] 버튼을 클릭하면, 선택한 값이 이벤트 설정 화면에 자동 반영됩니다.



3. 이벤트 레벨(Critical, Warning, Info)을 선택하세요. 하나의 지표에 여러 레벨을 설정할 수 있습니다.

> **이벤트 레벨 동작**
>
> 여러 조건을 만족하더라도 가장 높은 우선 순위(Critical → Warning → Info)의 레벨만 이벤트가 발생합니다. 레벨 동작 방식은 레벨 동작 가이드를 참고하세요.
>
> - 이벤트 레벨 수준은 위험(Critical)과 경고(Warning), 정상(Info)으로 구분합니다.
> - 같은 지표로 레벨을 설정하고 단계별 임계치를 설정하는 것을 권장합니다.

4. 선택한 이벤트 레벨의 필드, 연산자, 값을 입력하세요.

   a. **+ 추가**를 클릭해 조건을 추가할 수 있습니다.

   - `&&` : **모든 조건을 동시에 만족**해야 이벤트 발생

- `||`: **하나 이상의 조건만 만족**하면 이벤트 발생

b. — 를 클릭하면 해당 조건이 삭제됩니다.

> ⓘ 조건식 작성 방법과 지원되는 연산자는 조건 설정 가이드를 참고하세요.

발생 횟수

조건이 몇 번 충족되었을 때 이벤트를 발생시킬지 설정합니다. 지속적인 문제만 감지하고 싶을 때 유용합니다.

1. 발생 빈도를 선택하세요.
   - **연속**: 설정한 횟수만큼 이벤트가 연속으로 발생하는 경우 이벤트 발생
   - **최근**: 선택한 시간 내에 설정한 횟수만큼 이벤트가 발생하는 경우 이벤트 발생
     - 선택 가능 시간: `5초` , `10초` , `15초` , `30초` , `1분` , `2분` , `3분` , `4분` , `5분` , `10분` , `30분` , `1시간`
2. 발생 횟수를 입력하세요.
   - **Interval** 값은 선택한 카테고리의 데이터 수집 간격입니다. (Interval: 10s)

일시 중지

과도한 이벤트 발생을 방지하기 위해, 이벤트 동작을 일시 중지할 시간을 선택하세요. 이벤트 발생 후 설정한 시간 동안 동일한 이벤트가 발생하지 않습니다.

- 선택 가능 시간: `사용 안 함` , `5분` , `10분` , `15분` , `20분` , `30분` , `1시간` , `2시간` , `3시간` , `6시간` , `12시간` , `1일`

> ⓘ **해소된 알림 활성화 시**
>
> 알림은 정상 상태 알림(**RECOVERED**)을 받은 시점을 기준으로 설정한 시간이 지나기 전까지 동일한 이벤트가 발생해도 알림이 발생하지 않습니다.

해소된 알림

해소된 알림 기능을 활성화하면, 이벤트 발생 이후 상태 변화(진행 중 → 해소됨)를 추적할 수 있습니다. **이벤트 기록** 메뉴에서 진행

중인 이벤트로 표시됩니다.

- 해소된 알림은 Critical과 Warning 레벨에만 적용되며, Info 레벨은 항상 단발성 이벤트로 동작합니다.
- Critical과 Warning 레벨의 이벤트가 해소되면, 정상(RECOVERED) 상태의 알림이 전송됩니다.
- 토글 버튼을 클릭해 **활성화** 또는 **비활성화**하세요.
  - **활성화**: 이벤트는 상태 기반으로 동작하며, 임계치 초과 시 진행 중, 임계치 이하로 복구 시 해소 상태로 전환됨
  - **비활성화**: 임곗값을 초과할 때마다 단발성 이벤트 발생

이벤트 동작 시간

이벤트가 특정 시간대(근무/비근무/점검 시간 등)에 동작하도록 태그를 설정하세요. 태그를 설정하지 않으면 이벤트가 활성화된 상태에서 24시간 상시 동작합니다.

1. **+ 추가**를 클릭하세요.
2. 이벤트 동작 시간에서 **+새로운 태그 생성**을 클릭하세요.
3. **이벤트 동작 시간 태그 생성**에서 **태그 이름, 요일, 시간, 색상**을 선택하고 [적용] 버튼을 클릭하세요.
4. 생성한 태그는 태그 목록에서 확인할 수 있으며, 체크 박스를 선택하면 적용됩니다.
5. 수정 또는 삭제할 태그의 아이콘을 클릭해 **이벤트 동작 시간 태그 수정**에서 태그를 수정하거나 태그 삭제할 수 있습니다.

> 태그 삭제 시 해당 태그가 적용된 모든 사용자의 태그가 삭제됩니다. 단, 이벤트 규칙에서 사용 중인 태그는 삭제할 수 없습니다.

시뮬레이션

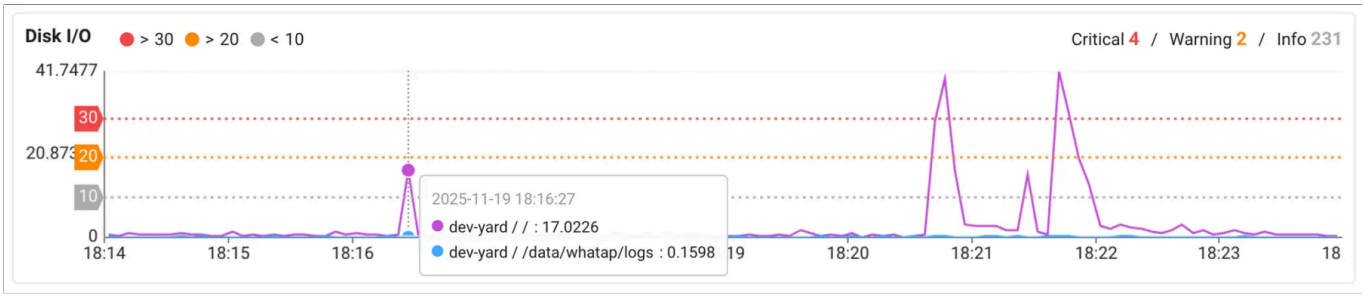
시뮬레이션 기능은 설정한 이벤트 조건을 과거 데이터에 기반으로 사전 테스트 할 수 있습니다. 예상 이벤트 수가 지나치게 많거나 부족하다면 조건을 조정해 최적의 임곗값을 찾을 수 있습니다.

- **규칙 검증**: 새로운 이벤트 규칙을 만들었을 때, 과거 데이터에 적용하면 어떤 결과가 나오는지 확인 가능
- **임곗값 튜닝**: 이벤트가 자주 발생하거나 전혀 발생하지 않을 때 적절한 수준으로 임곗값 조정

시뮬레이션 시작하기

1. 이벤트 조건 설정이 완료되면 [시뮬레이션] 버튼을 클릭하세요.
2. 예상 이벤트 건수와 시각화된 결과를 확인합니다.

시뮬레이션 결과 이해하기



1. **시계열 데이터**

   지표의 시간대별 변화와 각 레벨의 임계선이 그래프로 표시되어 이벤트 발생 시점을 직관적으로 확인할 수 있습니다.

2. **레벨별 발생 현황**

   화면 우측에 Critical, Warning, Info 레벨의 이벤트 발생 횟수가 표시됩니다. 전체 이벤트 분포와 빈도를 파악할 수 있습니다.

3. **대상별 상세 정보**

   차트의 특정 지점에 마우스를 올리면 해당 시점의 정확한 지표 값과 모니터링 대상이 표시됩니다. 여러 대상이 있는 경우 색상으로 구분되며, 대상별 패턴이 다른 경우 필터링 기능을 사용해 개별적으로 분석할 수 있습니다.

4. **피크 구간 식별**

   그래프에서 지표가 급격히 상승하는 시간을 확인하여 문제가 자주 발생하는 시간대나 패턴을 파악할 수 있습니다.

시뮬레이션 활용 팁

- **다양한 시간대 시뮬레이션**

  평일/주말, 피크/비피크 시간대를 각각 시뮬레이션하여 시간대별 최적 임곗값을 설정하세요.

- **실제 장애 기간 포함**

  과거 장애가 발생했던 기간을 포함하여 시뮬레이션을 실행하면 현재 규칙이 장애를 정확히 감지하는지 확인하세요.

- **대상 필터링 추가**

모니터링 대상이 많으면 시뮬레이션 결과가 복잡해집니다. 필터링 기능으로 특정 대상만 선택하여 명확한 결과를 확인할 수 있습니다.

- **임곗값 단계적 조정**

  임곗값을 한 번에 크게 변경하기보다 10% 단위로 점진적으로 조정하면서 최적값을 탐색하는 것이 좋습니다.

이벤트 대상 선택

이벤트 대상을 지정하지 않으면, 프로젝트에 포함된 전체 대상으로 이벤트가 발생되어 많은 이벤트가 발생할 수 있습니다. 따라서 이벤트가 적용될 대상을 명확히 설정하는 것을 권장합니다.

이벤트 대상을 변경한 경우, 이벤트 건수가 달라질 수 있습니다. **시뮬레이션** 버튼을 클릭한 후, 다시 실행해 예상 이벤트 수를 확인하세요.

대상 선택

이벤트가 적용될 대상을 **선택 입력** 또는 **직접 입력** 방식으로 지정할 수 있습니다.

- 선택 입력 **직접 입력**

1. 태그, 연산자, 값을 선택 또는 입력하세요. a. 태그를 찾기 어렵다면, **메트릭스 검색기**에서 태그를 검색할 수 있습니다.
2. **+ 추가**를 클릭해 조건(`&&` , `||`)으로 이벤트 대상을 추가하세요.

모니터링할 대상의 태그, 연산자, 값을 직접 입력하세요.

- 직접 입력을 선택하면 입력란 아래에 사용 가능한 태그 목록(**태그 선택 입력**)이 함께 제공됩니다.

```
ex. endsWith(okindName, 'example_name') && container == 'prod.billing'
ex. ${4xxErrorType} == '401'
```

입력 방법을 변경하면, 작성한 내용이 초기화됩니다.

대상 선택 연산자 목록

선택 입력 방식에서 문자열 태그를 선택하면 연산자 목록에 다음 두 항목이 함께 표시됩니다. 목록에 나타나는 이름은 함수 표기 그대로이며, `startsWith(key, 'match_text')` 와 같은 기존 항목과 형태가 같습니다.

- `equalsIgnoreCase(key, 'match_text')`
- `notEqualsIgnoreCase(key, 'match_text')`

두 연산자는 대소문자를 구분하지 않고 값을 비교합니다. **직접 입력** 방식에서는 같은 함수를 조건식에 직접 작성해 사용할 수 있습니다. 조건식 문법과 지원되는 함수 전체 목록은 조건 설정 가이드를 참고하세요.

> ⓘ 이벤트 발생 조건과 이벤트 대상 선택은 같은 문법을 사용하므로 두 함수를 양쪽 조건식에 모두 작성할 수 있습니다.

기본 정보 및 수신 설정

이벤트의 이름, 메시지, 수신 대상을 설정합니다. 설정한 내용은 이벤트 기록과 실제 알림 메시지에 그대로 표시되므로, 식별하기 쉽고 명확하게 작성하는 것이 중요합니다.

이벤트 활성화

설정한 조건을 충족했을 때 이벤트 발생 여부를 설정합니다.

- **활성화**: 설정한 조건을 충족하면 이벤트가 발생됨
- **비활성화**: 조건을 검사하지 않음

이벤트 이름

이벤트 제목으로 사용될 이벤트 이름을 입력하세요. 설정한 이벤트 이름은 **이벤트 목록** 및 **이벤트 기록** 메뉴에 표시되며, 해당 이름으로 검색할 수 있습니다.

메시지

이벤트 메시지를 입력하세요. 입력한 메시지는 이벤트 기록 및 알림에 사용됩니다. 을 클릭하면 이전에 작성한 메시지 기록을 확인할 수 있습니다. 메시지에 변수를 사용하여 이벤트 발생 시점의 실제 값을 포함할 수 있습니다.

- **변수 사용 규칙**
  - 변수 형식: `${지표명}`

- 설정한 **카테고리와 동일한 카테고리** 내 지표만 변수로 사용할 수 있습니다.
- 여러 변수를 조합해 상세한 메시지를 구성할 수 있습니다.
- 사용 가능한 변수 목록은 **메트릭스 조회** 메뉴에서 확인할 수 있습니다.

- **변수 활용 예시**

	이벤트 설정	이벤트 발생 시
제목	Disk 사용량 증가	Disk 사용량 증가
메시지	`Disk = ${disk}%`	`Disk = 89.9%`

이벤트 수신

이벤트 알림을 받을 멤버를 지정합니다.

- **전체 수신** **태그 선택 수신**

프로젝트에서 이벤트 수신 설정이 활성화된 모든 대상(멤버, 채널)에게 알림을 전송합니다.

해당 태그를 가진 프로젝트 멤버에게 알림을 전송합니다.

1. **태그 선택 수신**을 선택하세요.
2. **수신 태그** 항목에서 [+ 태그 추가] 또는 [+] 버튼을 클릭하세요.
3. **이벤트 수신 태그** 창 아래의 **+ 새 태그 생성**을 클릭하세요.
4. **태그 생성** 창에서 태그 이름을 입력하고, 색상을 선택한 후, [태그 생성] 버튼을 클릭해 태그를 생성하세요.
5. 생성된 태그는 태그 목록의 아이콘을 클릭해 수정 또는 삭제하세요.
6. **이벤트 수신 태그** 창에서 **태그 목록**에서 원하는 태그를 선택하면 적용됩니다.

> **수신 태그**
>
> 수신자 태그는 알림을 받을 사용자 또는 채널을 **그룹 단위로 관리하기 위한 기능**입니다. 이벤트에 수신자 태그를 설정하면, 해당 태그가 포함된 사용자에게만 알림이 전달됩니다. 수신 태그를 설정하지 않으면 프로젝트 내 모든 사용자에게 알림이 전송됩니다.

이벤트 수정/삭제

1. **경고 알림** > **이벤트 설정**의 **메트릭스** 탭으로 이동하세요.
2. 이벤트 목록에서 수정 또는 삭제하려는 이벤트의 아이콘을 클릭하세요.
3. **이벤트 규칙 수정** 창에서 옵션을 수정하고, [저장] 버튼을 클릭하세요.

   a. 선택한 이벤트를 삭제하려면 **이벤트 규칙 수정** 창의 오른쪽 위의 [삭제] 버튼을 클릭하세요.

이벤트 공유

메트릭스 이벤트 설정을 JSON 파일로 저장해 다른 사용자와 설정을 공유하거나 다른 사용자의 설정을 가져올 수 있습니다.

- JSON 파일명: event-rules- `YYYYMMDD` .json

내보내기

1. 화면 오른쪽 위에 [**JSON**] 버튼을 클릭하세요.
2. JSON 편집 창이 나타나면 [내보내기] 버튼을 클릭하세요.
   - 이벤트를 검색 후 내보내기 기능을 이용하면 검색한 목록만 JSON 파일로 다운받을 수 있습니다.
3. JSON 파일이 다운로드되면 공유할 다른 사용자에게 전달하세요.

가져오기

1. 화면 오른쪽 위에 [ ] 버튼을 선택하세요.
2. **내보내기** 기능을 통해 다운로드한 JSON 파일을 선택하세요.
3. JSON 편집 창이 나타나면 [목록에 추가하기] 또는 [덮어쓰기] 버튼을 선택하세요.

> 같은 종류의 제품 간에 이용할 것을 권장합니다. 다른 제품의 프로젝트로부터 이벤트 설정을 가져올 수는 있지만 정상 작동하지 않습니다.

Json 형식으로 수정하기

1. 화면 오른쪽 위에 [**JSON**] 버튼을 클릭하세요.
2. 편집 창이 나타나면 JSON 형식에 맞춰 내용을 수정하세요.
3. 수정을 완료하면 화면 아래 [저장] 버튼을 선택하세요.

> ⓘ 수정한 내용이 JSON 형식에 맞지 않으면 화면 아래에 에러 메시지가 표시되며, 저장할 수 없습니다. 표시되는 에러 메시지는 형식에 따라 다를 수 있습니다.
>
> Expected ',' or '}' after property value in JSON at position 1964 (line 80 column 42)

JSON 데이터 구조


```
{
  "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"
        ]
  }
}
```


표 | JSON 데이터 구조

JSON 필드	타입	설명	비고
version	Integer	이벤트 규칙의 버전	변경 제한
eventId	String	이벤트 규칙의 고유 식별자	
enabled	boolean	이벤트 활성화 여부	
stateful	boolean	상태 기반 이벤트 여부	
title	String	사용자가 설정한 이벤트 규칙 이름	
message	String	사용자가 설정한 이벤트 규칙 메시지	
category	String	데이터 카테고리	
alertLabel	`List<String>`	이벤트의 상태 관리를 위해 카테고리마다 지정되는 기본 식별자(primary key) 값	변경 제한
repeatCount	Integer	반복 횟수	
repeatDuration	Long	반복 기간	

JSON 필드	타입	설명	비고
silent	Integer	일시 중지 시간	
receiver	List<String>	수신자 태그의 키값 목록	
timeTag	List<String>	동작시간 태그의 키값 목록	
selectString	String	대상 선택	
conditions	List<Object>	레벨별 발생 조건 목록	
ㄴ level	byte	레벨 값	Critical: 30 Warning: 20 Info: 10
ㄴ enabled	boolean	레벨 활성화 여부	
ㄴ rule	String	레벨 발생 조건	
createTime	Long	이벤트 규칙 최초 생성 시간	
lastModifiedTime	Long	이벤트 규칙 마지막 수정 시간	
lastModifiedUser	String	이벤트 규칙 마지막 수정 계정	
basic	boolean	빠른 설정으로 생성한 이벤트 여부	변경 제한
metaId	String	빠른 설정 템플릿 고유 식별자	변경 제한: 이 값은 빠른 설정으로 생성한 이벤트에만 존재합니다. (basic=true)
selectCondition	Object	빠른 설정 대상 선택값	이 값은 빠른 설정으로 생성한 이벤트에만 존재합니다. (basic=true)

‘변경 제한’으로 표시된 필드는 이벤트 동작에 영향을 줄 수 있으므로 값을 변경하지 않는 것을 권장합니다.

부록

발생 조건, 대상 선택 가이드

메트릭스 경고 알림의 이벤트 발생 조건과 이벤트 대상 선택은 동일한 문법을 사용합니다. 단, 이벤트 발생 조건은 필드(Field)의 Key를 변수로 사용하고, 이벤트 대상 선택은 태그(Tag)의 Key를 변수로 사용합니다.

레벨 동작 가이드

이벤트 레벨은 위험(Critical)과 경고(Warning), 정상(Info)으로 구분합니다.

1. **우선순위 기반 이벤트 발생**

   여러 레벨의 조건을 동시에 충족할 경우, 가장 높은 우선순위의 이벤트만 발생합니다.

   예시

   ```
   설정:
   - Warning: CPU > 70%
   - Critical: CPU > 90%

   현재 상태: CPU 95%
   → 결과: Critical 이벤트만 발생 (Warning은 억제됨)
   ```

2. **레벨 상승**

   진행중 상태를 갖는 이벤트에서 낮은 레벨 이벤트가 진행 중일 때 높은 레벨 조건을 충족하면, 두 레벨 모두 진행 중 상태가 됩니다.

   예시

   ```
   설정:
   - Warning: CPU > 70%
   - Critical: CPU > 90%

   시나리오:
   1) CPU 80% → Warning 발생 (진행 중)
   ```

```
2) CPU 95% → Warning (진행 중), Critical (진행 중)
```

3. **레벨 하강**

   진행중 상태를 갖는 이벤트에서 높은 레벨에서 낮은 레벨로 전환되면 높은 레벨은 해소되고 낮은 레벨이 유지됩니다.

   예시

```
설정:
- Warning: CPU > 70%
- Critical: CPU > 90%

시나리오:
1) CPU 97% → Critical 발생
2) CPU 85% → Critical 해소, Warning 유지
3) CPU 65% → Warning 해소
```

4. **동작 흐름도**

```
- Warning: CPU > 70%
- Critical: CPU > 90%

메트릭 값: 60%  →  75%  →  92%  →  85%  →  60%
상태:      정상    Warning   Critical  Warning   정상
                   (발생)    (발생)    (유지)    (해소)
                             Warning
                             (유지)
```

> ⓘ 진행 중 상태를 갖는 이벤트는 조건이 충족되는 동안 계속 활성화 상태를 유지하며, 조건이 해제될 때까지 지속됩니다.

조건 설정 가이드

> ⚠ 지표명이 숫자로 시작하거나 특수문자가 포함된 경우 `${지표명}` 형태로 작성해야 합니다.
>
> - 괄호 `()`, `[]`

- 연산자 `+`, `-`, `*`, `/`, `%`
- 구분자 `:`, `@`, `#`, `,`, (공백)
- 기타 특수문자 `!`, `^`, `&`, `|`, `~`, ```, `=`

`${}` 사용 예시:

```
${cpu(xos)} > 50
${mem[0]} >= 100
${cpu-usage} > 80
${namespace:cpu} > 70
${metric name} > 60
${4xx_error} > 10
```

`${}` 없이 사용 가능: 영문자, 언더스코어(`_`), 점(`.`)만 포함된 경우

```
cpu > 80
cpu_usage > 50
CPUUtilization.Average > 0.8
```

1. 비교 연산자

숫자 비교

```
크다:          cpu > 80
크거나 같다:    cpu >= 80
작다:          memory < 1000
작거나 같다:    memory <= 1000
같다:          status == 200
같지 않다:      error != 0
```

문자열 비교

```
status == 'OK'
region == "us-east-1"
```

2. 산술 연산자

- 기본 산술 연산

```
덧셈:   cpu + 10 >= 90
뺄셈:   memory - 100 >= 500
곱셈:   cpu * 2 >= 100
나눗셈:  disk / 1024 >= 100
나머지:  value % 10 == 0
```

- 괄호를 이용한 우선순위 제어

```
(cpu + memory) * 2 >= 200
(disk - used) / total >= 0.2
((cpu + memory) / 2) >= 50
```

- 음수 표현

```
cpu > -100
```

3. 논리 연산자

- AND 연산 (&&)

```
cpu > 80 && memory > 1000
${cpu(xos)} > 50 && ${mem(xos)} > 60
```

- OR 연산 (||)

```
cpu > 90 || memory > 90
disk < 10 || network > 1000
```

- 복합 논리 연산

```
(cpu > 50 && memory > 50) || disk < 20
cpu > 80 && (memory > 1000 || disk > 500)
```

4. 패턴 매칭 연산자

- LIKE 연산자

```
oname like 'prod-*'
url like '*error*'
message like 'WARN%'
```

- NOT LIKE 연산자

```
oname not like 'test-*'
url not like '*debug*'
```

5. 내장 함수

- Null 체크 함수

```
isNull(value)          # null인지 확인
isNotNull(value)       # null이 아닌지 확인
nvl(value, 0)          # null이면 기본값 반환
isEmpty(str)           # 빈 문자열인지 확인
isNotEmpty(str)        # 빈 문자열이 아닌지 확인
```

- 집계 함수

```
sum(cpu, memory, disk)          # 합계
avg(cpu, memory)                # 평균
max(cpu, memory, disk)          # 최대값
min(cpu, memory, disk)          # 최소값
count(value1, value2, value3)   # 개수
```

- 수학 함수

```
round(cpu, 2)          # 반올림 (소수점 2자리)
```

- 문자열 함수

```
length(str)            # 문자열 길이
startsWith(str, 'prefix')  # 접두사 확인
```

```
endsWith(str, 'suffix')    # 접미사 확인
indexOf(str, 'search')     # 문자열 위치
substring(str, 0, 10)      # 부분 문자열
trim(str)                  # 공백 제거
replace(str, 'old', 'new') # 문자열 치환
hasStr(str, 'search')      # 문자열 포함 여부
```

문자열 함수

```
equalsIgnoreCase(str, 'match_text')     # 대소문자를 구분하지 않는 일치 비교
notEqualsIgnoreCase(str, 'match_text')  # 대소문자를 구분하지 않는 불일치 비교
```

`==` 와 `!=` 는 대소문자를 구분하므로 `WebServer` 와 `webserver` 를 다른 값으로 판단합니다. 표기가 일정하지 않은 태그나 필드를 비교할 때 위 두 함수를 사용하면 대소문자 차이를 무시하고 값을 맞출 수 있습니다.

예시

```
equalsIgnoreCase(okindName, 'WebServer')
notEqualsIgnoreCase(container, 'PROD.billing')
```

이벤트 발생 조건과 이벤트 대상 선택은 같은 문법을 사용하므로 두 함수를 양쪽 조건식에 모두 작성할 수 있습니다.

- 조건 함수
  - **if:** 조건식이 참(`true`)이면 `true값` 을 반환하고, 거짓(`false`)이면 `false값` 을 반환하는 조건 함수

    형식

    ```
    if(조건, true값, false값)
    ```

    예시

    ```
    if(value >= 90, 'High', 'Medium') == 'High'
    ```

  - **decode**: 값과 조건을 비교하여 일치하는 조건의 결과를 반환, 없으면 기본값 반환

    형식

```
decode(값, 조건1, 결과1, 조건2, 결과2, ..., 조건N, 결과N, 기본값)
```

예시

```
decode(value, 1, 'Low', 2, 'Medium', 3, 'High', 'Unknown') == 'Unknown'
```

- **in**: 값이 지정한 후보값 중 하나와 일치하면 `true`, 아니면 `false`

형식

```
in(값, 후보값1, 후보값2, ..., 후보값N)
```

예시

```
in(status, 200, 201, 204) == false
```

작성 시 주의사항

- 권장사항
  - 간단한 변수명은 `${}` 없이 사용 (예: cpu, memory)
  - 특수문자 포함 시 반드시 `${}` 사용
  - 복잡한 연산은 괄호로 명확하게 구분
  - 함수명은 정확히 입력 (대소문자 구분)
- 피해야 할 사항
  - 불완전한 표현식 작성 (예: `cpu >`, `memory &&`)
  - 괄호 미스매칭 (예: `(cpu > 80 && cpu > 70)`)
  - 연속된 연산자 사용 (예: `cpu >> 80`, `cpu >< 80`)

실시간 로그 이벤트

홈 > 프로젝트 선택 > 경고 알림 > **이벤트 설정** > **실시간 로그** 탭

수집한 로그 데이터를 조건에 맞춰 필터링하려 경고 알림을 설정할 수 있습니다. 로그 이벤트를 사용하려면 로그 모니터링이 활성화되어야 합니다. **로그 모니터링 활성화**에 대한 자세한 내용은 각 제품 로그 문서를 참고하세요.

- **실시간 로그 이벤트**: 실시간으로 수집한 로그에서 설정한 이벤트 조건이 충족되면 이벤트가 발생합니다.

기본 옵션

- **검색**: 이벤트 이름 또는 이벤트 수신 태그를 선택해 원하는 이벤트 검색
- **JSON 다운로드/업로드**: 이벤트 규칙은 JSON 편집기를 통해 바로 수정하거나, 파일로 다운로드하여 수정 후 다시 업로드 가능
- **알림 메시지 설정**: 수신할 메시지의 상세 항목 선택 및 항목 별 내용 편집
- **+ 이벤트 추가**: 새로운 로그 이벤트 추가

표 | 실시간 로그 이벤트 목록 구성

항목	설명
이벤트 이름	알림 제목으로 사용되는 이벤트 이름 - 최대 255byte까지 입력 가능
카테고리	로그 구분 명칭(로그 폴더명)
검색 키	로그 데이터에서 찾고 싶은 항목 - 예: HTTP 응답 상태 코드를 나타내는 값에 접근하고자 할 경우 검색 키 `status`
검색 값	검색 키에 해당하는 실제 데이터로 로그에서 입력한 단어를 포함할 경우 경고 알림을 보냄 - 예: 검색 키 `status` 검색 값 `200`을 설정한 경우 HTTP 응답 상태 코드 200을 포함하는 로그 데이터 수집 시 경고 알림 발생
이벤트 발생 일시 중지	이벤트 알림 발생 후, 같은 이벤트 발생을 일정 시간 동안 멈춤 - 선택 가능 시간: 사용 안 함, 5분, 10분, 15분, 20분, 30분, 1시간, 2시간, 3시간, 6시간, 12시간, 1일 - 해소된 알림이 활성화된 경우, 정상(RECOVERED) 상태 알림을 받은 후 선택한 시간 동안 같은 이벤트가

항목	설명
	발생해도 알림이 발생하지 않음
이벤트 대상 필터링	이벤트가 적용될 대상의 조건 - 대상 조건을 설정하지 않으면 수집되는 모든 데이터에 대해 알림 여부 판단 - 선택 입력: 태그, 연산자, 값을 선택해 조건 구성 - 직접 입력: 조건을 직접 입력해 대상 지정
이벤트 수신 태그	해당 이벤트 알림을 받을 사용자 또는 그룹(수신 태그)
	이벤트 수정 및 삭제
	이벤트의 활성화 여부 - 활성화: 조건을 충족하면 이벤트 발생 - 비활성화: 이벤트가 동작하지 않으며, 조건을 검사하지 않음

이벤트 설정

로그 이벤트는 실시간 로그 또는 일정 조건을 만족하는 로그가 누적될 때 경고 알림을 설정할 수 있는 기능입니다.

실시간 로그 이벤트 추가

1. **경고 알림** > **이벤트 설정**의 **실시간 로그** 탭을 클릭하세요.
2. 화면 오른쪽의 [+ 이벤트 추가] 버튼을 클릭하세요.
3. **이벤트 규칙 추가** 창에서 로그 이벤트 조건을 설정하세요.
    - **이벤트 이름**: 이벤트 제목으로 사용할 이름을 입력하세요.
    - **이벤트 활성화**: 토글 버튼 활성화 시, 토글 버튼 활성화 시, 조건을 충족하면 이벤트 발생합니다.
    - **레벨**: 이벤트 레벨(Critical, Warning, Info)을 선택하세요.
    - **메시지**: 이벤트 발생 시 전달받을 알림 메시지를 입력하세요. 메시지에 변수를 사용하여 이벤트 발생 시점의 실제 값을 포함할 수 있습니다.

> **변수 사용 규칙**
>
> - 변수 형식: `${지표명}`
> - 설정한 **카테고리와 동일한 카테고리** 내 지표만 변수로 사용할 수 있습니다.
> - 여러 변수를 조합해 상세한 메시지를 구성할 수 있습니다.
> - 사용 가능한 변수 목록은 **메트릭스 조회**에서 확인할 수 있습니다.

- **카테고리**: 로그 구분 명칭(로그 폴더명)을 목록에서 선택하거나 직접 입력하세요. 카테고리를 선택해야 검색 키와 검색 값을 입력할 수 있습니다.
- **검색 키**: 로그 데이터에서 찾고 싶은 항목을 입력하세요.
  - 예: HTTP 응답 상태 코드를 나타내는 값에 접근하고자 할 경우 **검색 키** `status`
- **검색 값**: **검색 키**에 해당하는 실제 데이터를 입력하세요. 로그에서 입력한 단어를 포함할 경우 경고 알림을 보냅니다.
  - 예: **검색 키** `status` **검색 값** `200` 을 설정한 경우 HTTP 응답 상태 코드 200을 포함하는 로그 데이터 수집 시 경고 알림 발생
- **이벤트 대상 필터링**: 이벤트가 적용될 대상의 조건을 입력하세요. 입력값이 없으면, 수집되는 모든 데이터에 대해 알림 여부 판단합니다.
  - **선택 입력**: 태그, 연산자, 값을 선택해 대상 조건을 설정하세요.

    > 조건식 작성 방법과 지원되는 연산자는 조건 설정 가이드를 참고하세요.

  - **직접 입력**: 조건을 직접 입력하세요.
- **이벤트 발생 일시 중지**: 알림을 일시 중지할 시간을 선택하세요. 이벤트 발생 후 설정한 시간 동안 동일한 이벤트가 발생하지 않습니다.
  - 선택 가능 시간: `사용 안 함` , `1분` , `2분` , `3분` , `5분` , `10분` , `20분` , `30분` , `1시간` , `3시간` , `6시간` , `12시간`
- **이벤트 동작 시간**: 이벤트가 특정 시간대(근무/비근무/점검 시간 등)에 동작하도록 태그를 설정하세요. 태그를 설정하지 않으면 이벤트가 활성화된 상태에서 24시간 상시 동작합니다.

1. **+ 추가**를 클릭하세요.
2. **이벤트 동작 시간**에서 **+새로운 태그 생성**을 클릭하세요.
3. **이벤트 동작 시간 태그 생성**에서 **태그 이름, 요일, 시간, 색상**을 선택하고 [적용] 버튼을 클릭하세요.
4. 생성한 태그는 태그 목록에서 확인할 수 있으며, 체크 박스를 선택하면 적용됩니다.
5. 수정 또는 삭제할 태그의 ✎ 아이콘을 클릭해 **이벤트 동작 시간 태그 수정**에서 태그를 수정하거나 태그 삭제할 수 있습니다.

> ⚠ 태그 삭제 시 해당 태그가 적용된 모든 사용자의 태그가 제거됩니다. 단, 이벤트 규칙에서 사용 중인 태그는 삭제할 수 없습니다.

- **이벤트 수신 태그**: 이벤트에 수신 태그를 설정하면, 해당 태그를 가진 프로젝트 멤버에게 알림을 전송합니다. 수신자 태그를 설정하지 않으면 프로젝트 내 모든 사용자에게 알림이 전송됩니다.

  1. **+ 태그 추가** 또는 **+**를 클릭하세요.
  2. **이벤트 수신 태그** 창 아래의 **+ 새 태그 생성**을 클릭하세요.
  3. **태그 생성** 창에서 태그 이름을 입력하고, 색상을 선택한 후, [태그 생성] 버튼을 클릭해 태그를 생성하세요.
  4. 생성된 태그는 태그 목록의 ✎ 아이콘을 클릭해 수정 또는 삭제하세요.
  5. **이벤트 수신 태그** 창의 **태그 목록**에서 원하는 태그를 선택하면 적용됩니다.

  > ✓ 이벤트 설정 시 이벤트 수신 태그를 선택하여 해당 태그를 가진 프로젝트 멤버와 3rd-party 플러그인에 알림을 전송할 수 있습니다. **경고 알림 > 이벤트 수신 설정**에서 프로젝트 멤버와 3rd-party 플러그인에 각각 태그를 지정할 수 있습니다.

4. **이벤트 규칙 추가** 창에서 설정이 완료되면 [저장] 버튼을 클릭하세요.

이벤트 수정/삭제

1. **경고 알림** > **이벤트 설정**의 **실시간 로그** 탭으로 이동하세요.

2. 로그 이벤트 목록에서 수정 또는 삭제하려는 이벤트의 아이콘을 클릭하세요.
3. **이벤트 설정** 화면에서 옵션을 수정하고, [저장] 버튼을 클릭하세요.

   a. 선택한 이벤트를 삭제하려면 **이벤트 설정** 화면의 오른쪽 위의 [삭제] 버튼을 클릭하세요.

이벤트 공유

메트릭스 이벤트 설정을 JSON 파일로 저장해 다른 사용자와 설정을 공유하거나 다른 사용자의 설정을 가져올 수 있습니다.

- JSON 파일명: event-rules-`YYYYMMDD`.json

내보내기

1. 화면 오른쪽 위에 [**JSON**] 버튼을 클릭하세요.
2. JSON 편집 창이 나타나면 [내보내기] 버튼을 클릭하세요.
   - 이벤트를 검색 후 내보내기 기능을 이용하면 검색한 목록만 JSON 파일로 다운받을 수 있습니다.
3. JSON 파일이 다운로드되면 공유할 다른 사용자에게 전달하세요.

가져오기

1. 화면 오른쪽 위에 [ ] 버튼을 선택하세요.
2. **내보내기** 기능을 통해 다운로드한 JSON 파일을 선택하세요.
3. JSON 편집 창이 나타나면 [목록에 추가하기] 또는 [덮어쓰기] 버튼을 선택하세요.

> 같은 종류의 제품 간에 이용할 것을 권장합니다. 다른 제품의 프로젝트로부터 이벤트 설정을 가져올 수는 있지만 정상 작동하지 않습니다.

Json 형식으로 수정하기

1. 화면 오른쪽 위에 [**JSON**] 버튼을 클릭하세요.
2. 편집 창이 나타나면 JSON 형식에 맞춰 내용을 수정하세요.

3. 수정을 완료하면 화면 아래 [저장] 버튼을 선택하세요.

ⓘ 수정한 내용이 JSON 형식에 맞지 않으면 화면 아래에 에러 메시지가 표시되며, 저장할 수 없습니다. 표시되는 에러 메시지는 형식에 따라 다를 수 있습니다.

```
Expected ',' or '}' after property value in JSON at position 1964 (line 80 column 42)
```

부록

발생 조건, 대상 선택 가이드

메트릭스 경고 알림의 이벤트 발생 조건과 이벤트 대상 선택은 동일한 문법을 사용합니다. 단, 이벤트 발생 조건은 필드(Field)의 Key를 변수로 사용하고, 이벤트 대상 선택은 태그(Tag)의 Key를 변수로 사용합니다.

레벨 동작 가이드

이벤트 레벨은 위험(Critical)과 경고(Warning), 정상(Info)으로 구분합니다.

1. **우선순위 기반 이벤트 발생**

   여러 레벨의 조건을 동시에 충족할 경우, 가장 높은 우선순위의 이벤트만 발생합니다.

   예시

   ```
   설정:
   - Warning: CPU > 70%
   - Critical: CPU > 90%

   현재 상태: CPU 95%
   → 결과: Critical 이벤트만 발생 (Warning은 억제됨)
   ```

2. **레벨 상승**

   진행중 상태를 갖는 이벤트에서 낮은 레벨 이벤트가 진행 중일 때 높은 레벨 조건을 충족하면, 두 레벨 모두 진행 중

상태가 됩니다.

예시

```
설정:
- Warning: CPU > 70%
- Critical: CPU > 90%

시나리오:
1) CPU 80% → Warning 발생 (진행 중)
2) CPU 95% → Warning (진행 중), Critical (진행 중)
```

3. 레벨 하강

진행중 상태를 갖는 이벤트에서 높은 레벨에서 낮은 레벨로 전환되면 높은 레벨은 해소되고 낮은 레벨이 유지됩니다.

예시

```
설정:
- Warning: CPU > 70%
- Critical: CPU > 90%

시나리오:
1) CPU 97% → Critical 발생
2) CPU 85% → Critical 해소, Warning 유지
3) CPU 65% → Warning 해소
```

4. 동작 흐름도

```
- Warning: CPU > 70%
- Critical: CPU > 90%

메트릭 값: 60%  →  75%  →  92%  →  85%  →  60%
상태:      정상     Warning   Critical  Warning   정상
                    (발생)    (발생)    (유지)    (해소)
                              Warning
                              (유지)
```

진행 중 상태를 갖는 이벤트는 조건이 충족되는 동안 계속 활성화 상태를 유지하며, 조건이 해제될 때까지 지속됩니다.

조건 설정 가이드

지표명이 숫자로 시작하거나 특수문자가 포함된 경우 `${지표명}` 형태로 작성해야 합니다.

- 괄호 `()`, `[]`
- 연산자 `+`, `-`, `*`, `/`, `%`
- 구분자 `:`, `@`, `#`, `,`, (공백)
- 기타 특수문자 `!`, `^`, `&`, `|`, `~`, ```, `=`

`${}` 사용 예시:

```
${cpu(xos)} > 50
${mem[0]} >= 100
${cpu-usage} > 80
${namespace:cpu} > 70
${metric name} > 60
${4xx_error} > 10
```

`${}` 없이 사용 가능: 영문자, 언더스코어(`_`), 점(`.`)만 포함된 경우

```
cpu > 80
cpu_usage > 50
CPUUtilization.Average > 0.8
```

1. **비교 연산자**

   숫자 비교

```
크다:          cpu > 80
크거나 같다:    cpu >= 80
작다:          memory < 1000
작거나 같다:    memory <= 1000
같다:          status == 200
```

```
같지 않다:      error != 0
```

문자열 비교

```
status == 'OK'
region == "us-east-1"
```

2. 산술 연산자

- 기본 산술 연산

```
덧셈:   cpu + 10 >= 90
뺄셈:   memory - 100 >= 500
곱셈:   cpu * 2 >= 100
나눗셈:  disk / 1024 >= 100
나머지:  value % 10 == 0
```

- 괄호를 이용한 우선순위 제어

```
(cpu + memory) * 2 >= 200
(disk - used) / total >= 0.2
((cpu + memory) / 2) >= 50
```

- 음수 표현

```
cpu > -100
```

3. 논리 연산자

- AND 연산 (&&)

```
cpu > 80 && memory > 1000
${cpu(xos)} > 50 && ${mem(xos)} > 60
```

- OR 연산 (||)

```
cpu > 90 || memory > 90
disk < 10 || network > 1000
```

- 복합 논리 연산

```
(cpu > 50 && memory > 50) || disk < 20
cpu > 80 && (memory > 1000 || disk > 500)
```

4. 패턴 매칭 연산자

- LIKE 연산자

```
oname like 'prod-*'
url like '*error*'
message like 'WARN%'
```

- NOT LIKE 연산자

```
oname not like 'test-*'
url not like '*debug*'
```

5. 내장 함수

- Null 체크 함수

```
isNull(value)          # null인지 확인
isNotNull(value)       # null이 아닌지 확인
nvl(value, 0)          # null이면 기본값 반환
isEmpty(str)           # 빈 문자열인지 확인
isNotEmpty(str)        # 빈 문자열이 아닌지 확인
```

- 집계 함수

```
sum(cpu, memory, disk)          # 합계
avg(cpu, memory)                # 평균
max(cpu, memory, disk)          # 최대값
```

```
min(cpu, memory, disk)             # 최소값
count(value1, value2, value3)      # 개수
```

- 수학 함수

```
round(cpu, 2)          # 반올림 (소수점 2자리)
```

- 문자열 함수

```
length(str)              # 문자열 길이
startsWith(str, 'prefix')  # 접두사 확인
endsWith(str, 'suffix')    # 접미사 확인
indexOf(str, 'search')     # 문자열 위치
substring(str, 0, 10)      # 부분 문자열
trim(str)                # 공백 제거
replace(str, 'old', 'new') # 문자열 치환
hasStr(str, 'search')      # 문자열 포함 여부
```

문자열 함수

```
equalsIgnoreCase(str, 'match_text')     # 대소문자를 구분하지 않는 일치 비교
notEqualsIgnoreCase(str, 'match_text')  # 대소문자를 구분하지 않는 불일치 비교
```

`==` 와 `!=` 는 대소문자를 구분하므로 `WebServer` 와 `webserver` 를 다른 값으로 판단합니다. 표기가 일정하지 않은 태그나 필드를 비교할 때 위 두 함수를 사용하면 대소문자 차이를 무시하고 값을 맞출 수 있습니다.

예시

```
equalsIgnoreCase(okindName, 'WebServer')
notEqualsIgnoreCase(container, 'PROD.billing')
```

이벤트 발생 조건과 이벤트 대상 선택은 같은 문법을 사용하므로 두 함수를 양쪽 조건식에 모두 작성할 수 있습니다.

- 조건 함수
  - **if:** 조건식이 참(`true`)이면 `true값` 을 반환하고, 거짓(`false`)이면 `false값` 을 반환하는 조건 함수

형식

```
if(조건, true값, false값)
```

예시

```
if(value >= 90, 'High', 'Medium') == 'High'
```

- **decode**: 값과 조건을 비교하여 일치하는 조건의 결과를 반환, 없으면 기본값 반환

형식

```
decode(값, 조건1, 결과1, 조건2, 결과2, ..., 조건N, 결과N, 기본값)
```

예시

```
decode(value, 1, 'Low', 2, 'Medium', 3, 'High', 'Unknown') == 'Unknown'
```

- **in**: 값이 지정한 후보값 중 하나와 일치하면 `true`, 아니면 `false`

형식

```
in(값, 후보값1, 후보값2, ..., 후보값N)
```

예시

```
in(status, 200, 201, 204) == false
```

작성 시 주의사항

- 권장사항
  - 간단한 변수명은 `${}` 없이 사용 (예: cpu, memory)
  - 특수문자 포함 시 반드시 `${}` 사용
  - 복잡한 연산은 괄호로 명확하게 구분
  - 함수명은 정확히 입력 (대소문자 구분)

- 피해야 할 사항
  - 불완전한 표현식 작성 (예: `cpu >` , `memory &&`)
  - 괄호 미스매칭 (예: `(cpu > 80 && cpu > 70)`)
  - 연속된 연산자 사용 (예: `cpu >> 80` , `cpu >< 80`)

복합 로그 이벤트 설정

홈 > 프로젝트 선택 > **경고 알림** > **이벤트 설정** > **복합 로그** 탭

수집한 로그 데이터를 조건에 맞춰 필터링하려 경고 알림을 설정할 수 있습니다. 로그 이벤트를 사용하려면 로그 모니터링이 활성화되어야 합니다. **로그 모니터링 활성화**에 대한 자세한 내용은 (각 제품 로그 문서 링크)를 참고하세요.

- **복합 로그 이벤트**: 설정한 **데이터 조회 범위** 내 로그를 기준으로, 로그 검색 조건을 충족하는 로그 수가 임계 개수 이상일 경우 이벤트가 발생합니다.

> **권한**
>
> 복합 메트릭스 이벤트를 설정하려면 **알림 설정** 권한이 필요합니다. 알림 설정 권한은 **홈 화면 > 프로젝트 선택 > 관리 > 통합 멤버 관리**에서 권한을 부여할 멤버의 아이콘을 클릭해 설정할 수 있습니다.

기본 옵션

- **검색**: 이벤트 이름 또는 이벤트 수신 태그를 선택해 원하는 이벤트 검색
- **JSON 다운로드/업로드**: 이벤트 규칙은 JSON 편집기를 통해 바로 수정하거나, 파일로 다운로드하여 수정 후 다시 업로드 가능
- **알림 메시지 설정**: 수신할 메시지의 상세 항목 선택 및 항목 별 내용 편집
- **+ 이벤트 추가**: 새로운 로그 이벤트 추가

표 | 복합 로그 이벤트 목록 구성

항목	설명
이벤트 이름	알림 제목으로 사용될 이벤트 이름
템플릿	복합 로그 템플릿
카테고리	로그 구분 명칭(로그 폴더명)
규칙	이벤트 발생하는 조건

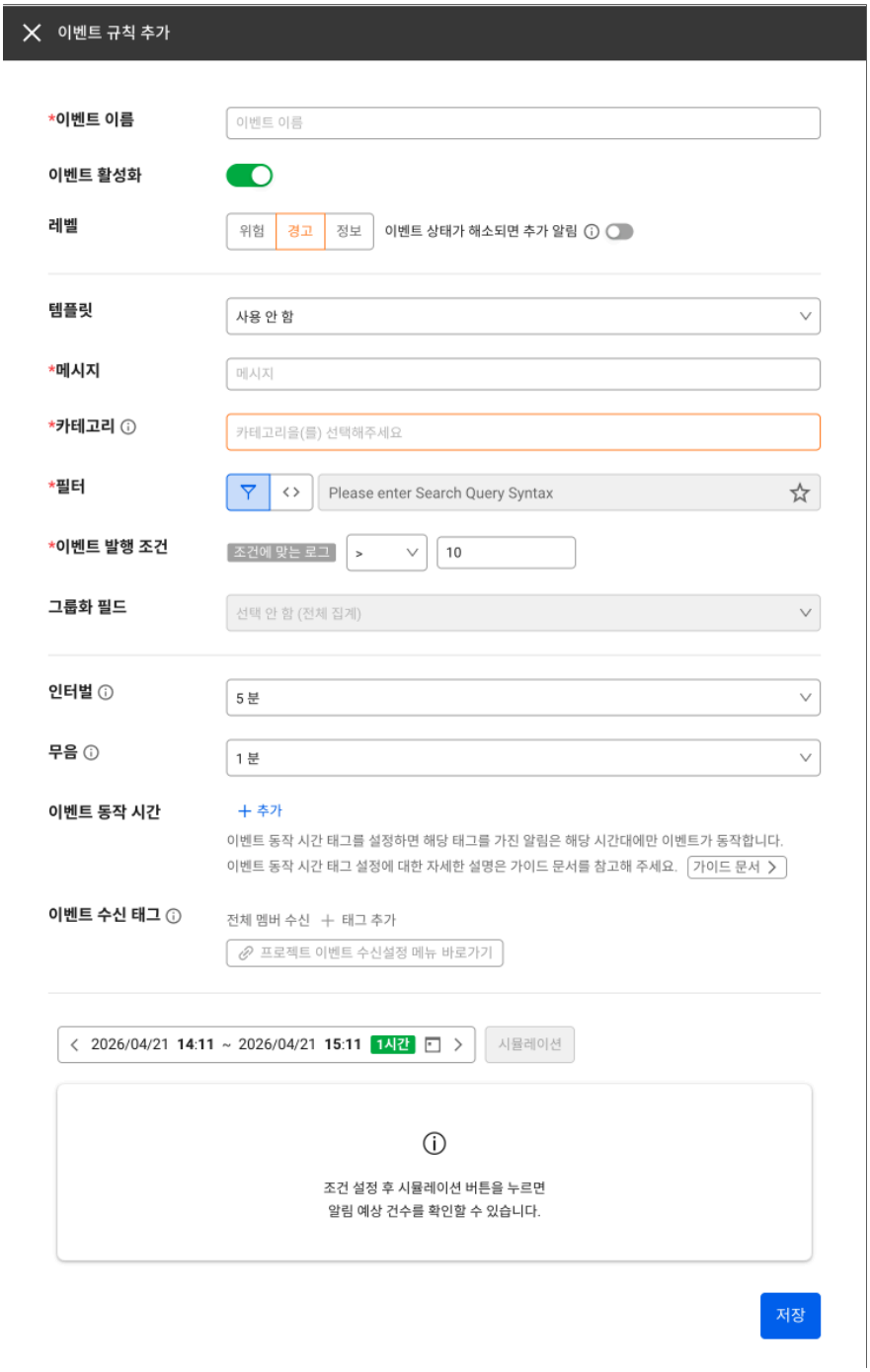
항목	설명
이벤트 상태가 해소되면 추가 알림	Critical과 Warning 레벨의 이벤트가 해소되면 정상(RECOVERED) 상태의 알림 수신 여부 - 활성화: 진행 중 → 정상 상태로 전환될 때 RECOVERED 알림을 보내며, 이벤트 기록에서 진행 중/해소 상태를 구분해 볼 수 있음 - 비활성화: 임계값을 초과할 때 마다 단발성 이벤트 발생
이벤트 발생 일시 중지	이벤트 알림 발생 후, 같은 이벤트 발생을 일정 시간 동안 멈춤 - 선택 가능 시간: 사용 안 함, 5분, 10분, 15분, 20분, 30분, 1시간, 2시간, 3시간, 6시간, 12시간, 1일 - 해소된 알림이 활성화된 경우, 정상(RECOVERED) 상태 알림을 받은 후 선택한 시간 동안 같은 이벤트가 발생해도 알림이 발생하지 않음
인터벌	로그 데이터를 조회할 시간 범위
이벤트 수신 태그	해당 이벤트 알림을 받을 사용자 또는 그룹(수신 태그)
	이벤트 수정 및 삭제
	이벤트의 활성화 여부 - 활성화: 조건을 충족하면 이벤트 발생 - 비활성화: 이벤트가 동작하지 않으며, 조건을 검사하지 않음

이벤트 설정

복합 로그 이벤트는 설정한 **데이터 조회 범위** 내 로그를 기준으로, 로그 검색 조건을 충족하는 로그 수가 임계 개수 이상일 경우 경고 알림을 설정할 수 있는 기능입니다.

복합 로그 이벤트 추가

1. **경고 알림** > **이벤트 설정**의 **복합 로그** 탭을 클릭하세요.
2. 화면 오른쪽의 [+ 이벤트 추가] 버튼을 클릭하세요.
3. **이벤트 규칙 추가** 창에서 로그 이벤트 조건을 설정하세요. 자세한 내용은 기본 설정, 상세 설정, 추가 설정를 참고하세요.



4. **이벤트 규칙 추가** 창에서 설정이 완료되면 [저장] 버튼을 클릭하세요.

기본 설정

- **이벤트 이름**: 이벤트 제목으로 사용할 이름을 입력하세요.
- **이벤트 활성화**: 토글 버튼 활성화 시, 조건을 충족하면 이벤트 발생합니다.
- **레벨**: 이벤트 레벨(Critical, Warning, Info)을 선택하세요.

상세 설정

템플릿(프리셋), 메시지, 카테고리, **필터**, 이벤트 발생 조건 항목을 설정합니다.

템플릿(프리셋)

미리 정의된 프리셋을 선택하면 **필터, 쿼리 타입, 이벤트 발행 조건**이 자동으로 설정됩니다. 프리셋을 선택한 후 필요에 맞게 수정할 수 있습니다.

복합 로그 템플릿 유형

프리셋	쿼리 타입	기본 규칙
사용 안 함	none	`rows > 10`
2xx 상태코드 건수	count	`count > 10`
3xx 상태코드 건수	count	`count > 10`
4xx 상태코드 건수	count	`count > 10`
5xx 상태코드 건수	count	`count > 10`
정상 상태코드(2xx, 3xx) 건수	count_v2	`include_minus_exclude_count > 10`
에러 상태코드(4xx, 5xx) 건수	count_v2	`include_minus_exclude_count > 10`
에러 수신 건수	count	`count > 10`
예외 수신 건수	count	`count > 10`

프리셋	쿼리 타입	기본 규칙
초당 로그 발생건수	rps	`rps > 100`
필드 집계 (Aggregation)	aggr	`avg > 1000`
비율 (Rate)	rate	`rate > 10`
고유값 수 (Cardinality)	cardinality	`cardinality > 100`
백분위 (Percentile)	percentile	`p99 > 3000`

메시지

이벤트 발생 시 전달받을 알림 메시지를 입력하세요. 메시지에 변수를 사용하여 동적 메시지를 구성할 수 있습니다.

- 예시: 메시지 `${host}`에서 에러 `${count}`건 발생 → 알림 메시지 web-01에서 에러 15건 발생

표 | 메시지 변수

변수	설명	사용 조건
`${count}`	조건에 맞는 로그 건수	항상 사용 가능
`${groupField}`	그룹화 필드 이름	그룹화 알림만
`${groupValue}`	그룹화 필드 값	그룹화 알림만
`${필드명}`	그룹화 필드 값 (필드명으로 직접 참조)	그룹화 알림만

`${count}` 변수는 그룹화 여부와 관계없이 모든 알림에서 사용할 수 있습니다. `${groupField}`, `${groupValue}`, `${필드명}` 변수는 그룹화 필드를 설정한 경우에만 치환됩니다.

카테고리

로그 구분 명칭(로그 폴더명)을 목록에서 선택하거나 직접 입력하세요. 카테고리를 선택해야 검색 키와 검색 값을 입력할 수 있습니다.

필터

Lucene 쿼리 문법으로 로그 검색 조건을 입력합니다. 두 가지 입력 모드를 지원합니다.

- **필터 모드** : 필드와 값을 선택하여 쿼리를 구성합니다.
- **코드 모드** <>: Lucene 쿼리를 직접 입력합니다.

표 | Lucene 쿼리 예시

쿼리	설명
`level:ERROR`	level 필드가 ERROR인 로그
`status_nxx:status_4xx OR status_nxx:status_5xx`	4xx 또는 5xx 상태코드
`appender:accessLog`	accessLog appender의 로그
`response_time.n:*`	response_time.n 필드가 존재하는 로그
`host:web-01 AND level:ERROR`	web-01 호스트의 에러 로그

이벤트 발생 조건

쿼리 타입별로 설정합니다. 선택한 템플릿(프리셋) 또는 쿼리 타입에 따라 추가 설정 항목이 표시됩니다.

표 | 이벤트 발행 조건 키 의미

키	의미
`rows` / `count`	필터 조건에 맞는 로그 건수
`rps`	초당 로그 발생건수 (건수 / 인터벌 초)

키	의미
`avg`	대상 필드 값의 평균
`sum`	대상 필드 값의 합계
`min`	대상 필드 값의 최솟값
`max`	대상 필드 값의 최댓값
`rate`	분자 건수 / 분모 건수 × 100 (%)
`cardinality`	대상 필드의 고유한 값 개수
`p99` / `p95` / `p90` / `p75` / `p50`	대상 필드의 백분위 값
`include_minus_exclude_count`	포함 필터 건수 - 제외 필터 건수

- **건수 (count / none)**: 설정한 필터 조건에 맞는 로그의 건수를 카운트합니다.

표 | 건수 설정 항목

설정 항목	설명
필터	로그 검색 조건 (Lucene 쿼리)
이벤트 발행 조건	`건수` 또는 `조건에 맞는 로그` `{연산자}` `{임계값}` (예: `count > 10`)

- **초당 로그 발생건수 (rps)**: 설정한 필터 조건에 맞는 로그의 초당 발생건수(Requests Per Second)를 계산합니다. 인터벌 시간 동안의 총 건수를 인터벌 초로 나누어 산출합니다.

표 | 초당 로그 발생건수 설정 항목

설정 항목	설명
필터	로그 검색 조건 (Lucene 쿼리)
이벤트 발행 조건	`RPS` `{연산자}` `{임계값}` (예: `rps > 0.5`)

- **필드 집계 (aggr)**: 지정한 숫자 필드의 집계 값(평균, 합계, 최솟값, 최댓값)을 계산합니다. 이벤트 발행 조건에서 집계 방식을 드롭다운으로 선택할 수 있습니다.

표 | 필드 집계 설정 항목

설정 항목	설명
필터	로그 검색 조건 (Lucene 쿼리)
대상 필드	집계 대상 숫자 필드 (예: `response_time.n`)
이벤트 발행 조건	`평균` / `합계` / `최솟값` / `최댓값` `{연산자}` `{임계값}` (예: `avg > 1000`)

- **비율 (rate)**: 분자 필터와 분모 필터의 건수 비율(%)을 계산합니다. 전체 로그 대비 특정 조건의 비율을 모니터링할 때 유용합니다.

표 | 비율 설정 항목

설정 항목	설명
필터 (분자)	비율 계산의 분자가 되는 로그 조건 (예: `level:ERROR`)
분모 필터	비율 계산의 분모가 되는 로그 조건 (예: `level:*`)
이벤트 발행 조건	`비율 (%)` `{연산자}` `{임계값}` (예: `rate > 10`)

  - **예시**: 필터 `level:ERROR`, 분모 필터 `level:*`, rule `rate > 10` → 전체 로그 중 에러 비율이 10%를 초과하면 알림 발생

- **고유값 수 (cardinality)**: 지정한 필드의 고유한 값 개수를 계산합니다. 비정상적으로 많은 종류의 값이 발생하는 경우를 감지할 때 유용합니다.

표 | 고유값 수 설정 항목

설정 항목	설명
필터	로그 검색 조건 (Lucene 쿼리)
대상 필드	고유값을 계산할 필드 (예: `host`)

설정 항목	설명
이벤트 발행 조건	`고유값 수` `{연산자}` `{임계값}` (예: `cardinality > 100`)

- **백분위 (percentile)**: 지정한 숫자 필드의 백분위(Percentile) 값을 계산합니다. 응답 시간의 tail latency를 모니터링할 때 유용합니다. Percentile 선택을 변경하면 이벤트 발행 조건의 키가 자동으로 업데이트됩니다.

표 | 백분위 설정 항목

설정 항목	설명
필터	로그 검색 조건 (Lucene 쿼리)
대상 필드	백분위를 계산할 숫자 필드 (예: `response_time.n`)
Percentile	P99, P95, P90, P75, P50 중 선택
이벤트 발행 조건	`P99` / `P95` /... `{연산자}` `{임계값}` (예: `p99 > 3000`)

표 | 백분위 의미

백분위	의미
P50	중앙값. 전체 데이터의 50%가 이 값 이하
P75	전체 데이터의 75%가 이 값 이하
P90	전체 데이터의 90%가 이 값 이하
P95	전체 데이터의 95%가 이 값 이하
P99	전체 데이터의 99%가 이 값 이하

- **포함/제외 건수 (count_v2)**: 포함 필터의 건수에서 제외 필터의 건수를 뺀 순수 건수를 계산합니다.

표 | 포함/제외 건수 설정 항목

설정 항목	설명
필터 (포함)	포함할 로그 조건 (예: `status_nxx:status_4xx OR status_nxx:status_5xx`)
필터 제외	제외할 로그 조건 (예: `status_nxx:status_4xx`)
분모 필터	비율 계산용 분모 (선택사항)
이벤트 발행 조건	`포함-제외 건수` `{연산자}` `{임계값}` (예: `include_minus_exclude_count > 10`)

- **예시**: 필터 `status_nxx:status_4xx OR status_nxx:status_5xx`, 필터 제외 `status_nxx:status_4xx` → 4xx+5xx에서 4xx를 제외한 순수 5xx 건수가 10을 초과하면 알림 발생

그룹화 필드(Group By)

그룹화 필드를 설정하면 해당 필드의 값별로 독립적으로 알림 조건을 판정합니다. 드롭다운에서 그룹화할 필드를 선택합니다. 드롭다운을 클릭하면 현재 카테고리에서 사용 가능한 필드 목록이 자동완성됩니다.

- **예시:** 필터 `level:ERROR`, 그룹화 필드 `host`, rule `rows > 3`

표 | 그룹화 설정 비교

항목	그룹화 없음	그룹화 사용
판정 단위	전체 로그를 하나로 집계	그룹 값별 독립 집계
알림 발생	1건	조건 충족 그룹 수만큼
알림 제목	`이벤트 제목`	`[그룹 값] 이벤트 제목`
Stateful	전체 기준 상태 전이	그룹별 독립 상태 전이

- `[web-01] 에러 알림` — web-01에서 에러 8건 → 알림 발생
- `[web-02] 에러 알림` — web-02에서 에러 5건 → 알림 발생
- `[api-01] 에러 알림` — api-01에서 에러 2건 → 조건 미충족, 알림 미발생

> **이벤트 상태가 해소되면 추가 알림**을 활성화한 경우, 그룹별로 독립적으로 ABNORMAL → RECOVERED 상태가 전이됩니다. 예를 들어 `web-01` 에서 에러가 해소되면 `[web-01] RECOVERED` 알림이 발생하며, 다른 호스트의 상태에는 영향을 미치지 않습니다.

추가 설정

- **인터벌**: 선택한 시간(`unset` , `1분` , `5분` , `10분` , `30분` , `1시간` , `12시간`) 간격으로 알림 조건을 확인합니다.
- **무음**: 이벤트 발생 후, 선택한 시간(`unset` , `1분` , `5분` , `10분` , `30분` , `1시간` , `12시간`) 동안 같은 이벤트가 발생되지 않습니다.
- **이벤트 동작 시간**: 이벤트가 특정 시간대(근무/비근무/점검 시간 등)에 동작하도록 태그를 설정하세요. 태그를 설정하지 않으면 이벤트가 활성화된 상태에서 24시간 상시 동작합니다.
    1. **+ 추가**를 클릭하세요.
    2. **이벤트 동작 시간**에서 **+새로운 태그 생성**을 클릭하세요.
    3. **이벤트 동작 시간 태그 생성**에서 **태그 이름, 요일, 시간, 색상**을 선택하고 [적용] 버튼을 클릭하세요.
    4. 생성한 태그는 태그 목록에서 확인할 수 있으며, 체크 박스를 선택하면 적용됩니다.
    5. 수정 또는 삭제할 태그의 ✎ 아이콘을 클릭해 **이벤트 동작 시간 태그 수정**에서 태그를 수정하거나 태그 삭제할 수 있습니다.

> 태그 삭제 시 해당 태그가 적용된 모든 사용자의 태그가 제거됩니다. 단, 이벤트 규칙에서 사용 중인 태그는 삭제할 수 없습니다.

- **이벤트 수신 태그**: 이벤트에 수신 태그를 설정하면, 해당 태그를 가진 프로젝트 멤버에게 알림을 전송합니다. 수신자 태그를 설정하지 않으면 프로젝트 내 모든 사용자에게 알림이 전송됩니다.
    1. **+ 태그 추가** 또는 **+**를 클릭하세요.
    2. **이벤트 수신 태그** 창 아래의 **+ 새 태그 생성**을 클릭하세요.
    3. **태그 생성** 창에서 태그 이름을 입력하고, 색상을 선택한 후, [태그 생성] 버튼을 클릭해 태그를 생성하세요.
    4. 생성된 태그는 태그 목록의 ✎ 아이콘을 클릭해 수정 또는 삭제하세요.
    5. **이벤트 수신 태그** 창의 **태그 목록**에서 원하는 태그를 선택하면 적용됩니다.

> 이벤트 설정 시 이벤트 수신 태그를 선택하여 해당 태그를 가진 프로젝트 멤버와 3rd-party 플러그인에 알림을 전송할 수 있습니다. **경고 알림 > 이벤트 수신 설정**에서 프로젝트 멤버와 3rd-party 플러그인에 각각 태그를 지정할 수 있습니다.

시뮬레이션

알림을 저장하기 전에 과거 데이터로 시뮬레이션하여 알림 발생 예상 건수를 확인할 수 있습니다.

1. 알림 설정을 완료한 후 하단의 시간 범위를 선택하세요.
2. [시뮬레이션] 버튼을 클릭하세요.
3. 차트에 시간대별 데이터와 알림 발생 횟수가 표시됩니다.

   > 시뮬레이션 시간 범위를 조절하여 다양한 시간대의 알림 발생 패턴을 확인할 수 있습니다. 최근 1~2분의 데이터는 인덱스 빌드 지연으로 조회되지 않을 수 있습니다.

4. **이벤트 규칙 추가** 창에서 설정이 완료되면 [저장] 버튼을 클릭하세요.

로그 탐색기에서 알림 생성

로그 탐색기에서 직접 알림을 생성할 수 있습니다.

1. **로그** > **로그 탐색** 메뉴로 이동하세요.
2. 카테고리와 검색 쿼리를 설정하세요.
3. 화면 오른쪽 위의 [이벤트 추가] 버튼을 클릭하세요.
4. Drawer에서 알림 설정 폼이 열리며, 현재 카테고리와 검색 쿼리가 자동으로 채워집니다.
5. 추가 설정을 완료하고 [저장] 버튼을 클릭하세요.
6. 저장 완료 후 [이동] 버튼을 클릭하면 알림 목록으로 이동합니다.

이벤트 수정/삭제

1. **경고 알림** > **이벤트 설정**의 **복합 로그** 탭으로 이동하세요.
2. 로그 이벤트 목록에서 수정 또는 삭제하려는 이벤트의 아이콘을 클릭하세요.
3. **이벤트 설정** 화면에서 옵션을 수정하고, [저장] 버튼을 클릭하세요.

   a. 선택한 이벤트를 삭제하려면 **이벤트 설정** 화면의 오른쪽 위의 [삭제] 버튼을 클릭하세요.

이벤트 공유

메트릭스 이벤트 설정을 JSON 파일로 저장해 다른 사용자와 설정을 공유하거나 다른 사용자의 설정을 가져올 수 있습니다.

- JSON 파일명: event-rules-`YYYYMMDD`.json

내보내기

1. 화면 오른쪽 위에 [**JSON**] 버튼을 클릭하세요.
2. JSON 편집 창이 나타나면 [내보내기] 버튼을 클릭하세요.
   - 이벤트를 검색 후 내보내기 기능을 이용하면 검색한 목록만 JSON 파일로 다운받을 수 있습니다.
3. JSON 파일이 다운로드되면 공유할 다른 사용자에게 전달하세요.

가져오기

1. 화면 오른쪽 위에 [ ] 버튼을 선택하세요.
2. **내보내기** 기능을 통해 다운로드한 JSON 파일을 선택하세요.
3. JSON 편집 창이 나타나면 [목록에 추가하기] 또는 [덮어쓰기] 버튼을 선택하세요.

> 같은 종류의 제품 간에 이용할 것을 권장합니다. 다른 제품의 프로젝트로부터 이벤트 설정을 가져올 수는 있지만 정상 작동하지 않습니다.

Json 형식으로 수정하기

1. 화면 오른쪽 위에 [**JSON**] 버튼을 클릭하세요.
2. 편집 창이 나타나면 JSON 형식에 맞춰 내용을 수정하세요.
3. 수정을 완료하면 화면 아래 [저장] 버튼을 선택하세요.

> 수정한 내용이 JSON 형식에 맞지 않으면 화면 아래에 에러 메시지가 표시되며, 저장할 수 없습니다. 표시되는 에러 메시지는 형식에 따라 다를 수 있습니다.
>
> ```
> Expected ',' or '}' after property value in JSON at position 1964 (line 80 column 42)
> ```

이벤트 수신 설정

사용자별 이벤트 수신 설정

> **사용자별 이벤트 수신 설정** 의 다른 멤버의 정보는 **멤버 관리** 권한을 가진 멤버만 확인할 수 있습니다. 그 외 멤버에게는 필수 정보가 마스킹 처리됩니다.
>
> - 이벤트 알림의 일괄 수신설정 및 접근 설정을 위한 모바일 기기 관리는 **계정 정보**에서 할 수 있습니다.
> - 멤버 권한에 대한 자세한 설명은 다음 문서를 참고하세요.

수신 수단 설정

이메일, **SMS**, **WhatsApp**, **모바일** 알림을 제공합니다.

사용자별 이벤트 수신 설정 섹션에서 원하는 알림 수신 수단의 체크 박스를 선택하면 경고 알림을 받을 수 있습니다. 알림 수신 수단의 체크 박스를 해제하면 경고 알림을 보내지 않습니다.

> 프로젝트 최고 관리자를 제외한 모든 사용자는 자신의 수신 설정만 변경할 수 있습니다.

이메일 알림

이메일 알림은 회원 가입 시 입력한 이메일 주소로 알림을 보냅니다.

SMS 알림

SMS로 알림을 수신할 수 있습니다.

> SMS 알림 기능은 유료 프로젝트에 한정됩니다.

1. 화면 오른쪽 위의 **자신의 프로필 아이콘** > **계정 관리**로 이동하세요.
2. **계정 정보**의 **사용자 전화번호** 섹션에서 [일반 휴대전화] 버튼을 클릭하세요.
3. **전화번호**에 인증번호를 수신을 전화번호를 입력하세요.
    - SMS를 알림으로 수신할 수 있는 전화번호는 **한국의 휴대전화 번호만 등록**할 수 있습니다.
4. [저장] 버튼을 클릭하세요.

> **등록한 전화번호 변경 및 삭제**
>
> - 전화번호를 변경할 경우, [변경] 버튼을 클릭하세요. SMS 인증이 필요합니다.
> - 전화번호를 삭제할 경우, [삭제] 버튼을 클릭하세요. 전화번호 삭제 시 SMS로 수신받는 알림은 수신할 수 없습니다.

WhatsApp 알림

WhatsApp을 통해 알림을 수신할 수 있습니다.

> WhatsApp 알림 기능은 유료 프로젝트에 한정됩니다.

1. 화면 오른쪽 위의 **자신의 프로필 아이콘** > **계정 관리**로 이동하세요.
2. **계정 정보**의 **사용자 전화번호** 섹션에서 [WhatsApp] 버튼을 클릭하세요.
3. 인증번호를 받을 **전화번호**를 입력하세요.
4. [인증번호 전송] 버튼을 클릭하세요.
5. WhatsApp 애플리케이션으로 전송된 인증번호 6자리를 입력하세요.
    - 10분 내로 인증번호를 받지 못했다면, [인증번호 재전송] 버튼을 클릭하세요.
6. [인증하기] 버튼을 클릭하세요.

모바일 알림

모바일 알림 수신이 필요한 사용자는 모바일 기기에 와탭 애플리케이션을 설치한 후 로그인해야 합니다.

- 와탭 애플리케이션 다운로드 - **안드로이드**
- 와탭 애플리케이션 다운로드 - **iOS**

반복 알림(에스컬레이션) 설정

경고 알림 발생 시간으로부터 알림 발생 상황이 해소되지 않을 경우 최초 알림 발생 시각으로부터의 알림 반복 간격을 설정할 수 있습니다. 위험(Critical) 레벨의 알림만 적용됩니다.

예를 들어, 경고 알림 발생 시간으로부터 0분(즉시), 1시간 후, 1일 후에 경고 알림을 반복하려면 `0,1H,1D` 를 **반복 알림(에스컬레이션)** 컬럼 항목에 입력하세요.

1. **경고 알림** > **이벤트 수신 설정**의 **사용자별 이벤트 수신 설정** 섹션으로 이동하세요.
2. **반복 알림(에스컬레이션)** 항목에서 설정하세요.
   - **M**: 분, **H**: 시간, **D**: 일, 단위를 생략하면 분 단위로 시간을 설정합니다.
   - 숫자 또는 숫자+단위(**M**, **H**, **D**)로 입력하세요. 입력이 올바르지 않으면 메시지가 표시됩니다.
3. [저장] 버튼을 클릭하세요. 저장하지 않으면 설정되지 않습니다.

이벤트 수신 태그

이벤트 설정 시 이벤트 수신 태그를 선택하여 해당 태그를 가진 프로젝트 멤버와 3rd-party 플러그인에 알림을 전송할 수 있습니다. 이벤트 수신 태그를 설정하지 않으면 전체 멤버에게 경고 알림이 전송됩니다.

이벤트 수신 태그 추가/수정/삭제하기

> 이벤트에 적용 중인 **이벤트 수신 태그** 항목은 삭제할 수 없습니다.

1. **경고 알림** > **이벤트 수신 설정**으로 이동하세요.
2. **사용자별 이벤트 수신 설정** 섹션의 사용자 목록에서 **+ 태그 추가** 또는 **+** 를 클릭하세요.
3. **이벤트 수신 태그** 창에서 **+ 새 태그 생성**을 클릭하세요.
4. **태그 생성** 창에서 태그 이름을 입력하고, 색상을 선택한 후, [태그 생성] 버튼을 클릭해 태그를 생성하세요.
5. 생성된 태그는 태그 목록의 아이콘을 클릭해 수정 또는 삭제하세요.
6. **이벤트 수신 태그** 창의 **태그 목록**에서 원하는 태그를 선택하면 적용됩니다.

이벤트 수신 태그 해제하기

1. **경고 알림** > **이벤트 수신 설정**의 **사용자별 이벤트 수신 설정** 섹션으로 이동하세요.
2. 이벤트 수신 태그를 해제할 사용자의 **이벤트 수신 태그** 항목의 **+** 를 클릭하세요.
3. **이벤트 수신 태그** 창에서 해제할 태그의 X 를 클릭하면 해당 태그가 해제됩니다.

수신 태그 미설정 알림 받기

이벤트 설정 시 이벤트 수신 태그를 선택하여 해당 태그를 가진 프로젝트 멤버와 3rd-party 플러그인에 알림을 전송할 수 있습니다.

- 이벤트 수신 태그가 설정되지 않은 경고 알림을 받으려면 **수신 태그 미설정 알림 받기** 옵션을 선택하세요.
- 이벤트 수신 태그가 설정된 경고 알림만 받고 싶다면 선택을 해제하세요.

> ⓘ 모든 경고 알림을 받지 않으려면 해당 옵션을 해제하고 선택한 **이벤트 수신 태그**가 없어야 합니다.

이벤트 수신 모드 선택하기

사용자별 이벤트 수신 설정 섹션에서 멤버 또는 3rd 파티 채널별로 이벤트 알림 수신 범위를 다음 네 가지 모드 중에서 선택할 수 있습니다. 수신 모드는 각 멤버 또는 채널 행의 **이벤트 수신 태그** 열에서 라디오 버튼으로 선택하세요.

수신 모드	설명
모든 알림 받기	이벤트 수신 태그와 무관하게 모든 경고 알림 수신
선택한 태그만 받기	선택한 이벤트 수신 태그를 가진 경고 알림만 수신. 태그를 최소 한 개 이상 선택
선택한 태그만 제외하기	선택한 이벤트 수신 태그를 가진 경고 알림만 제외하고 나머지 전부 수신. 제외 태그를 최소 한 개 이상 선택
모든 알림 받지 않기	어떤 이벤트 알림도 받지 않음. 선택해 둔 태그는 삭제되지 않고 그대로 보존

- 이벤트 수신 태그가 지정되지 않은 경고 알림까지 받으려면 태그 선택 영역에서 **미분류** 가상 태그를 선택하세요. **미분류** 태그는 일반 태그와 동일하게 선택하거나 해제할 수 있으며, 태그 자체를 수정하거나 삭제할 수는 없습니다.

> ⓘ **모든 알림 받지 않기** 모드에서는 해당 멤버 또는 채널이 이벤트 알림을 전혀 받지 않습니다. 이때 선택해 둔 태그는 삭제되지 않고 보존되므로, 다시 **선택한 태그만 받기** 모드로 전환하면 이전에 선택한 태그 설정을 그대로 사용할 수 있습니다.

선택한 태그만 받기

1. **사용자별 이벤트 수신 설정** 섹션에서 설정할 멤버 또는 3rd 파티 채널 행의 **이벤트 수신 태그** 열을 클릭하세요.
2. **선택한 태그만 받기** 라디오 버튼을 선택하세요.
3. 태그 선택 영역에서 알림을 받을 태그를 하나 이상 선택하세요.

> ⓘ **선택한 태그만 받기** 모드에서 태그를 하나도 선택하지 않았을 때 나타나는 완전 차단 경고는 **선택한 태그만 제외하기** 모드에는 표시되지 않습니다. 제외 태그가 없는 상태는 알림을 전부 차단하는 상태가 아니라 전부 수신하는 상태이기 때문입니다.

선택한 태그만 제외하기

경고 알림을 대부분 받되 특정 분류만 걸러 내고 싶을 때 사용합니다. 받을 태그를 하나씩 고르는 **선택한 태그만 받기**와 방향이 반대입니다.

1. **사용자별 이벤트 수신 설정** 섹션에서 설정할 멤버 또는 3rd 파티 채널 행의 **이벤트 수신 태그** 열을 클릭하세요.
2. **선택한 태그만 제외하기** 라디오 버튼을 선택하세요.
3. 아래 태그 선택 영역에서 알림을 받지 않을 태그를 하나 이상 선택하세요.
    - 이벤트에 붙은 태그와 제외 태그가 하나라도 겹치면 해당 경고 알림은 전송되지 않습니다. 겹치는 태그가 없는 알림은 모두 수신합니다.
    - 제외 태그는 최소 한 개 이상 선택해야 합니다. 제외 태그가 하나도 없는 상태는 **모든 알림 받기**와 결과가 같아지므로, 마지막 태그는 제거할 수 없고 제외 태그가 비어 있는 동안 **최소한 하나의 태그를 선택해 주세요.** 안내가 표시됩니다.
    - 제외 태그 목록은 **선택한 태그만 받기** 모드의 수신 태그 목록과 별도로 저장됩니다. 두 모드를 번갈아 선택해도 각 모드에서 고른 태그가 그대로 남습니다.
    - **미분류** 가상 태그도 제외 대상으로 선택할 수 있으며, 태그 한 개로 계산됩니다.

알림 수신 언어 선택

사용자의 알림 수신 언어를 선택할 수 있습니다.

1. **경고 알림** > **이벤트 수신 설정**으로 이동하세요.
2. **사용자별 이벤트 수신 설정** 섹션의 사용자 목록에서 언어를 설정할 사용자의 가장 왼쪽에 ▶ 버튼을 클릭하세요.
3. **알림 수신 언어**를 선택하세요.

요일 및 시간별 알림 설정

사용자의 요일별, 시간별 알림 수신 여부를 선택할 수 있습니다.

1. **경고 알림** > **이벤트 수신 설정**으로 이동하세요.
2. **사용자별 이벤트 수신 설정** 섹션의 사용자 목록에서 요일 및 시간별 알림을 설정할 사용자의 가장 왼쪽에 ▶ 버튼을 클릭하세요.
3. 경고 알림 수신을 원하는 요일을 선택하거나 시간을 입력하세요.
    - 알림 수신 수단별로 설정할 수 있습니다.

3rd 파티 플러그인

외부 애플리케이션을 통해 경고 알림을 받을 수 있습니다.

- 지원 애플리케이션: Slack, Telegram, Teams, Teams Workflows, Jandi, AlertNow, Webhook, Webhook JSON, Opsgenie, PagerDuty, LINE, ilert

> ⓘ 와탭랩스의 지원 범위에 포함하지 않는 사내 메신저는 표준 Webhook, Webhook json을 통해 연동할 수 있습니다.

표 | 3rd 파티 플러그인 목록 구성

항목	설명
플러그인 이름	플러그인의 이름

항목	설명
인증 키	인증 키
인증 값	인증값
수신 레벨	경고, 전체, 위험으로 구분
반복 알림(에스컬레이션)	알림 발생 시간으로부터 알림 발생 상황이 해소되지 않았을 경우, 최초 알림 발생 시각으로부터의 알림 재발행 주기 - 위험(Critical) 단계의 알림에만 적용되는 기능
Smart 메시지(AI)	Telegram 3rd 파티 플러그인 추가 시 AI 알림 기능 제공 - 지원 언어: 한국어, 영어, 일본어
이벤트 수신 태그	프로젝트의 멤버 중 특정 멤버 또는 팀을 대상으로 알림 수신 여부
삭제	3rd 파티 플러그인 삭제

3rd 파티 플러그인 추가

1. **경고 알림** > **이벤트 수신 설정**으로 이동하세요.
2. **3rd 파티 플러그인** 섹션의 [추가하기] 버튼을 클릭하세요.
3. **서드 파티 플러그인 추가** 창에서 원하는 서비스 탭으로 선택하세요.
4. 선택한 서비스의 안내(3rd 파티 플러그인 알림 추가 방법)에 따라 설정을 진행하세요.
5. [수신 테스트] 버튼을 클릭해 설정이 잘 되었는지 확인하세요.
6. 모든 과정을 완료하면 [추가하기] 또는 [등록] 버튼을 클릭하세요.

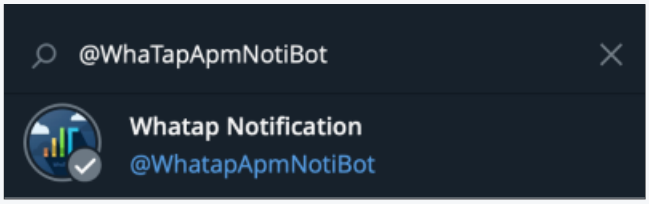
3rd 파티 플러그인 알림 추가 방법

Slack

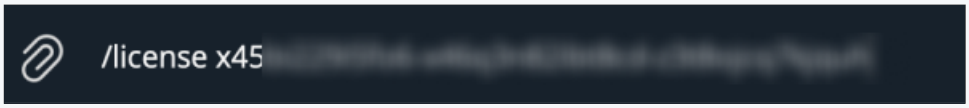
> **서드 파티 플러그인 추가**의 **Slack** 탭 화면의 [**슬랙에 추가하기**] 버튼을 클릭하면 Slack의 팀 및 채널에 **WhaTapNotiBot**을 등록할 수 있습니다.

Telegram

1. **Telegram**에서 **@WhaTapApmNotiBot**을 검색하세요.



2. Bot을 선택하고 [시작] 버튼을 클릭하세요.
3. 채팅창에 `/license` 명령어와 알림을 받을 프로젝트의 액세스 키를 채팅창에 입력하세요.

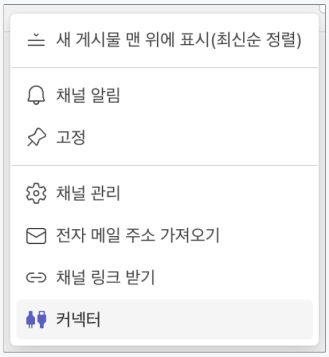


사용할 수 있는 명령어 목록

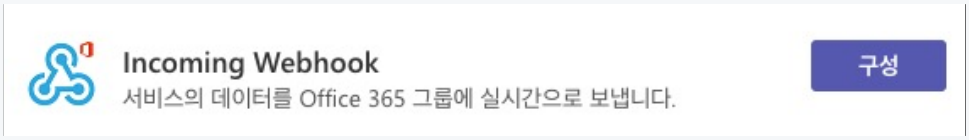
- `/help` : 명령어 목록 보기
- `/license` : 알림 받을 프로젝트의 액세스 키 입력
- `/remove` : 알림을 받지 않을 프로젝트 코드 입력
- `/list` : 현재 알림을 받고 있는 프로젝트 코드 보기

Teams

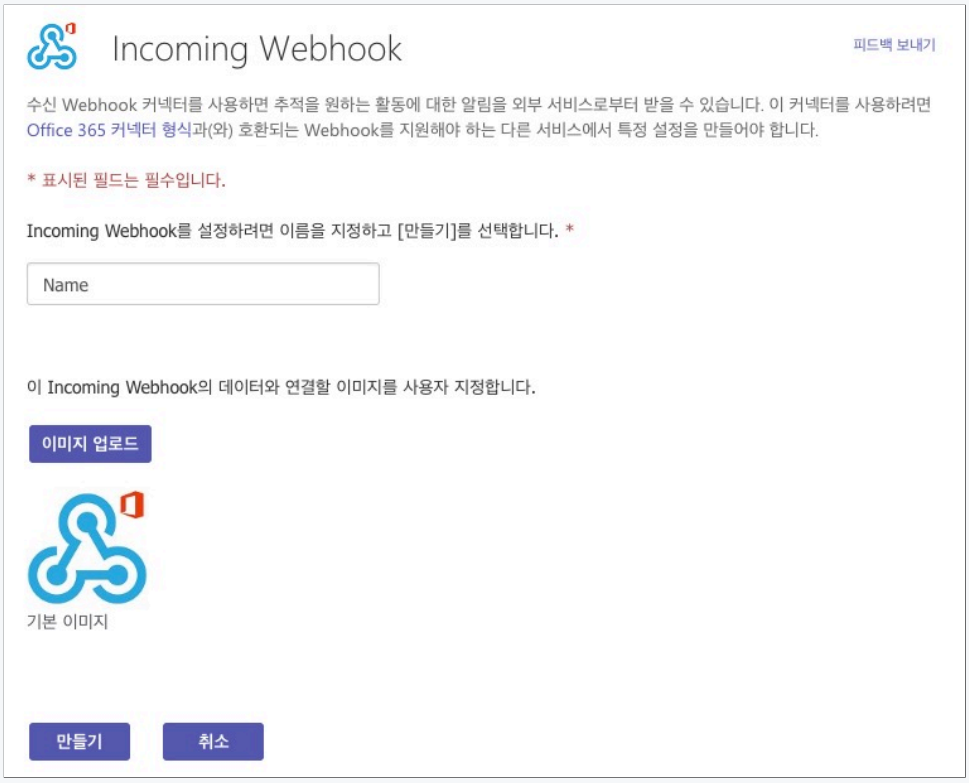
1. 알림을 받을 채널의 **커넥터**로 이동하세요.



2. **Webhook**을 추가하고 [구성] 버튼을 클릭하세요.



3. Webhook 이름을 입력하고 [만들기] 버튼을 클릭하세요.



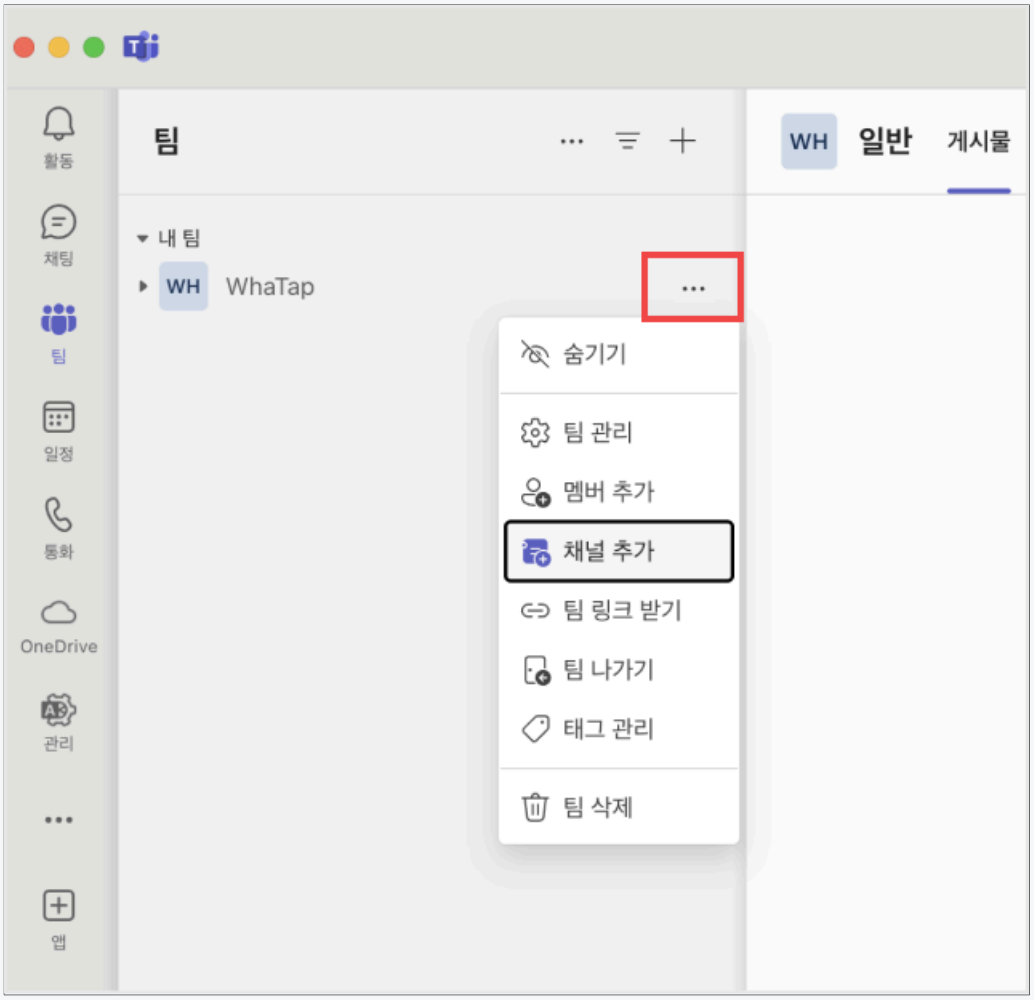
4. 생성된 Webhhok URL을 복사하세요.

5. 와탭 모니터링 서비스의 **서드 파티 플러그인 추가** 창으로 돌아와 **Teams** 탭 하단의 **Channel Name**과 **Webhook URL** 항목을 입력하세요.
6. [**등록**]버튼을 클릭하세요.

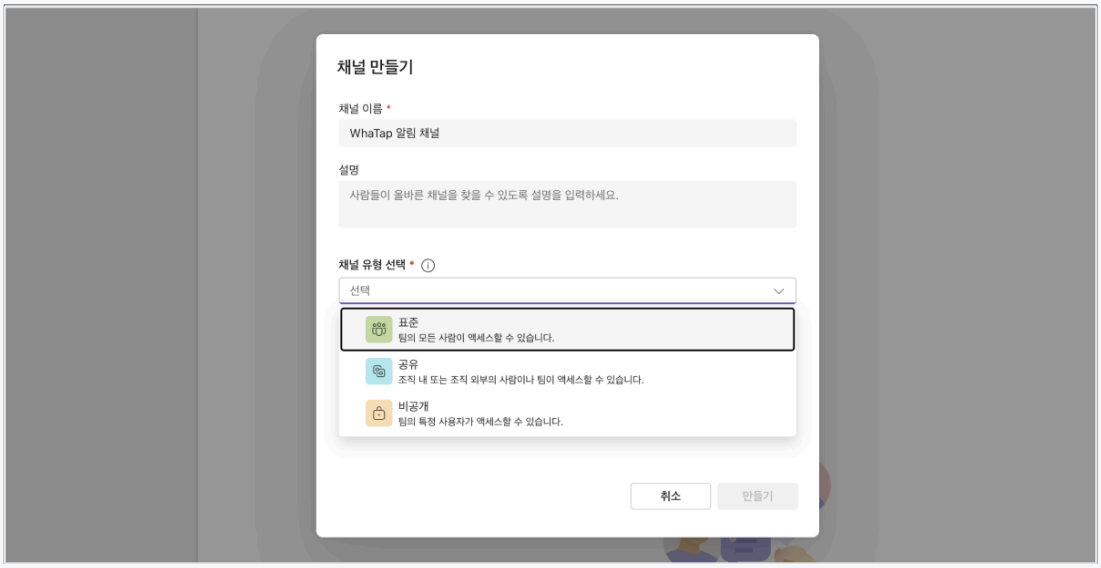
Teams Workflows

워크플로를 사용하려면 **Microsoft Teams**에 액세스할 수 있는 계정과 비즈니스 플랜 구독이 필요합니다.

1. **팀** 메뉴의 더보기(...) > **채널 추가**를 클릭하세요.

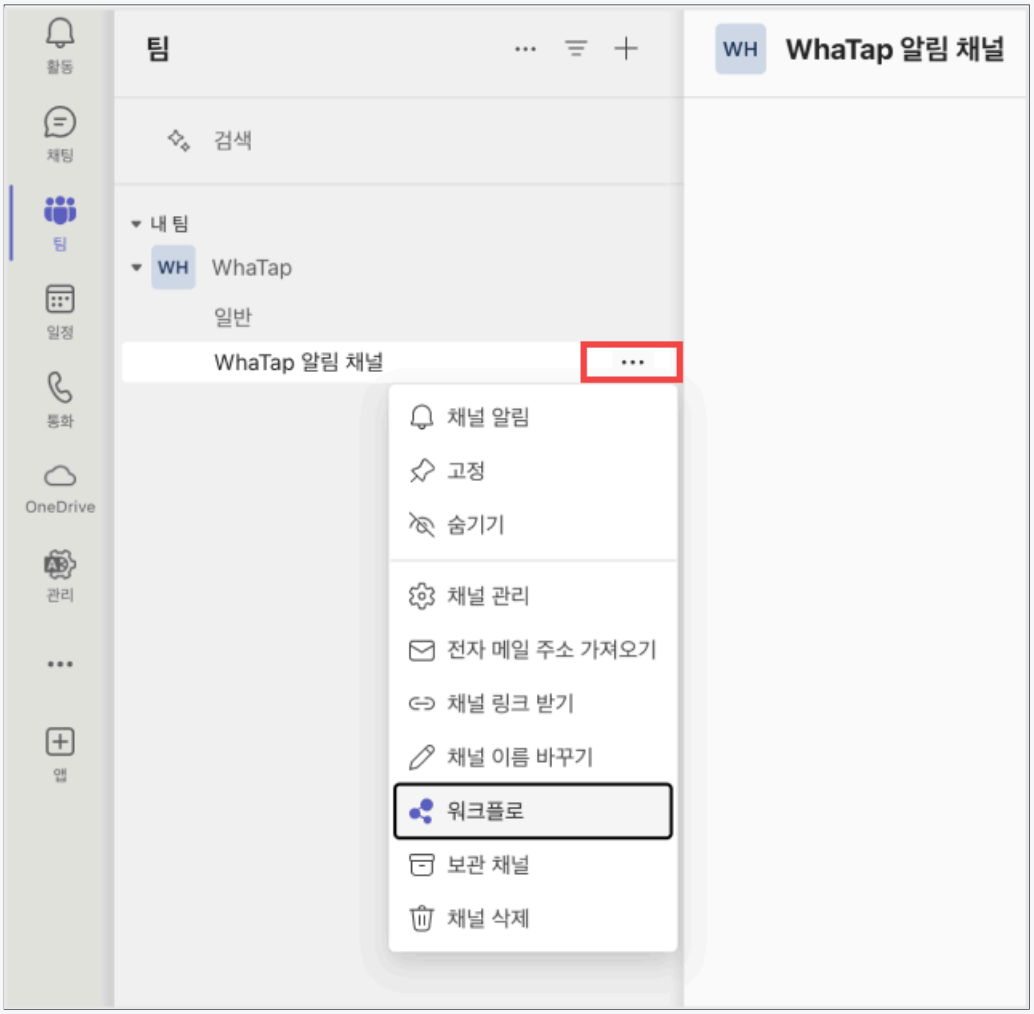


2. **채널 만들기** 창에서 **채널 이름**과 **설명**을 입력하고, **채널 유형 선택** 항목은 **표준**으로 선택한 후, [만들기] 버튼을 클릭하세요.



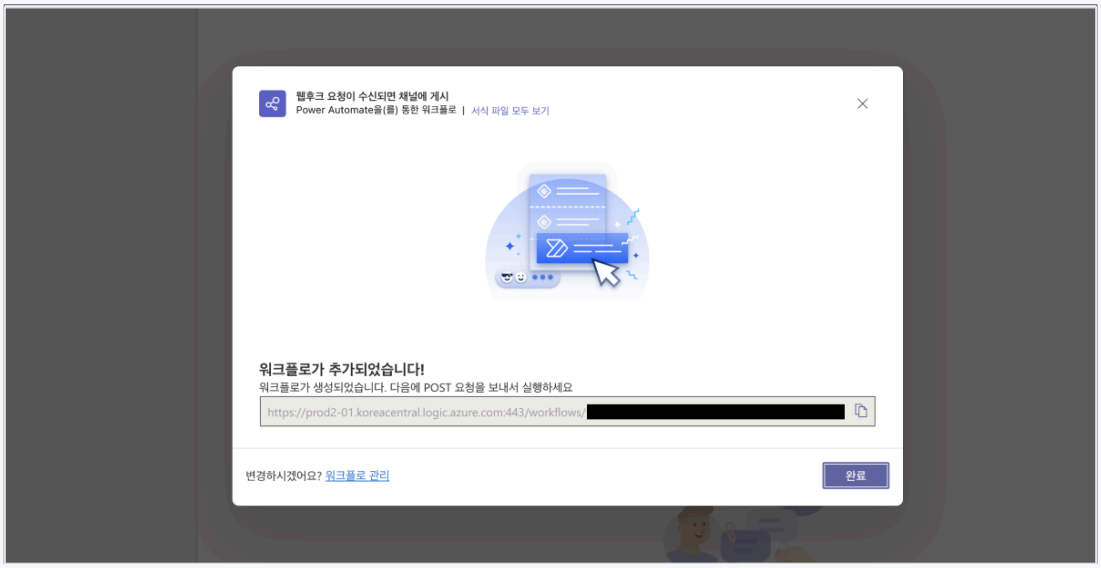
> 채널 유형을 **공유**로 선택하면 워크플로를 사용할 수 없으며, **비공개**를 선택하면 워크플로를 추가할 수 있지만 알림을 수신할 수 없습니다.

3. 생성한 채널의 더보기(...) > **워크플로**를 클릭하세요.



4. **워크플로** 대화상자의 **워크플로**의 목록에서 **웹후크 요청이 수신되면 채널에 게시**를 선택 후, [추가] 버튼을 클릭하세요.

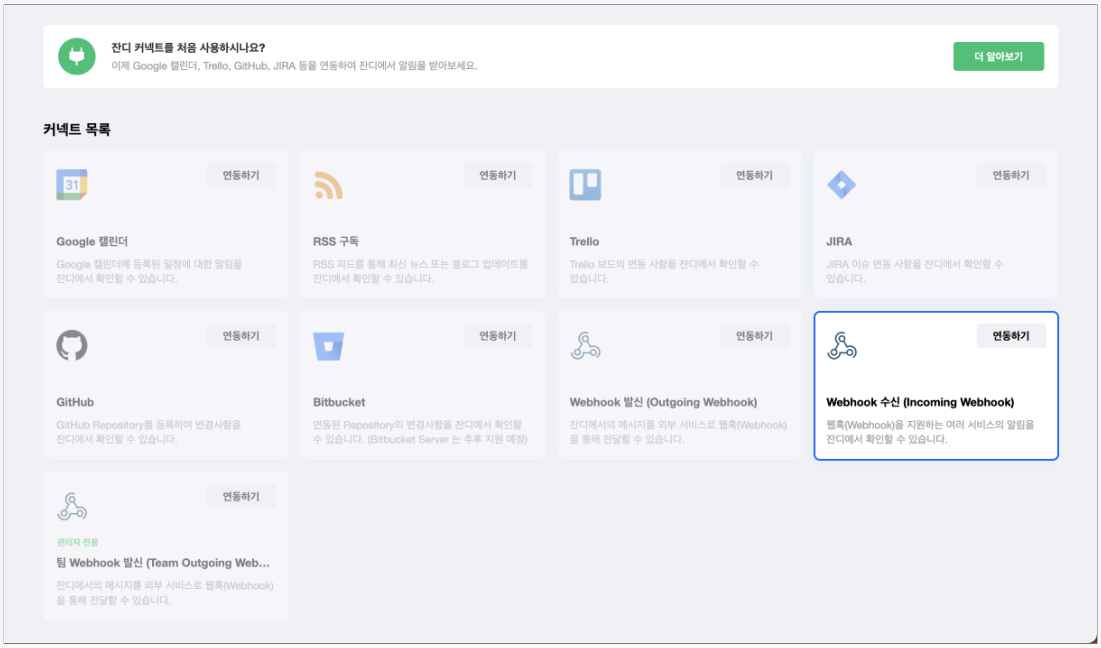
5. **이름**과 **연결** 상태를 확인한 후, [다음] 버튼을 클릭하세요.
6. **세부 정보**를 확인한 후, [워크플로 추가] 버튼을 클릭하세요.
7. 워크플로가 추가되었습니다. 생성된 주소를 복사하세요.



8. 와탭 모니터링 서비스의 **서드 파티 플러그인 추가** 창으로 돌아와 **Teams Workflows** 탭 안내 하단의 **Channel Name**과 **Webhook URL** 항목을 입력하세요.
9. [**등록**] 버튼을 선택하세요.

잔디(Jandi)

1. 알림을 받을 토픽 또는 채팅에서 잔디 커넥트 설정으로 이동하세요.
2. **커넥트 목록**에서 **Webhook 수신 (Incoming Webhook)** 항목에서 [연동하기] 버튼을 클릭하세요.

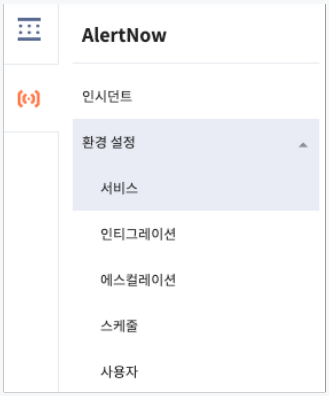


3. [Webhook 수신 추가] 버튼을 클릭하세요.
4. **Webhook URL 등록** 항목에 Webhook URL을 복사하세요.

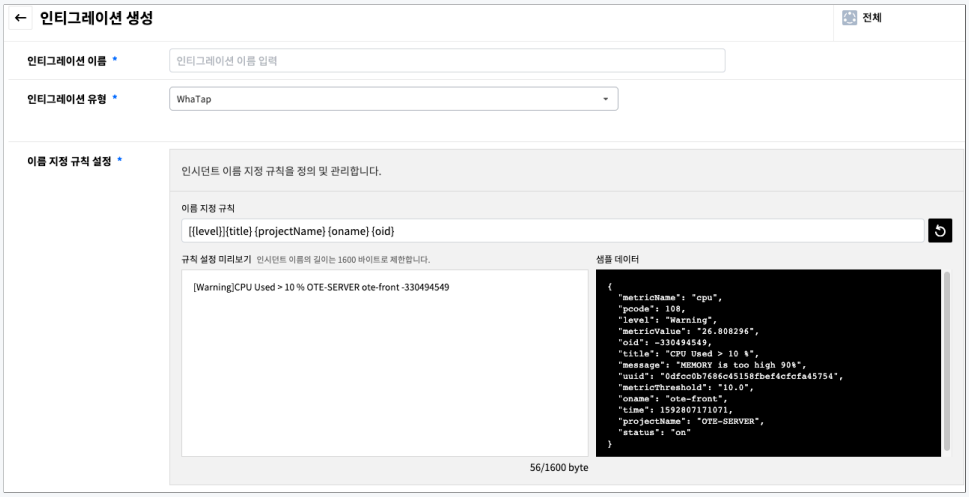
5. 와탭 모니터링 서비스의 **서드 파티 플러그인 추가** 창으로 돌아와 **Jandi** 탭 안내 하단의 **Channel Name**과 **Webhook URL** 항목을 입력하세요.
6. [**등록**] 버튼을 클릭하세요.

AlertNow

1. **AlerNow** 서비스를 접속한 후, **환경 설정** > **Integration** 메뉴로 이동하세요.



2. **인티그레이션** 화면에서 [인티그레이션 생성] 버튼을 클릭하세요.
3. Integration 유형 중 **WhaTap**을 선택하세요.
4. **인티그레이션 생성** 화면에서 이름 및 이름 지정 규칙을 설정하세요.



5. [확인] 버튼을 클릭하면 Integration이 생성됩니다.
6. 생성된 인티그레이션 목록을 확인하고 생성된 Integration 항목을 선택하세요.

7. 상세 화면에서 URL 오른쪽의 복사 아이콘을 클릭해 URL을 복사하세요.

8. 와탭 모니터링 서비스의 **서드 파티 플러그인 추가** 창으로 돌아와 **AlertNow** 탭 안내 하단의 **Channel Name**과 **Webhook URL** 항목을 입력하세요.
9. [**등록**] 버튼을 클릭하세요.

Webhook

1. 알림을 받을 Webhook을 등록하세요. 등록한 URL에 POST Method로 알림의 `title` , `text` 가 전송됩니다.

```
{
    "title":"Alert Title",
    "text": "Sample warning message"
}
```

1. **Webhook Name**과 **Webhook URL** 항목을 입력하세요.
2. [**등록**] 버튼을 클릭하세요.

Webhook JSON

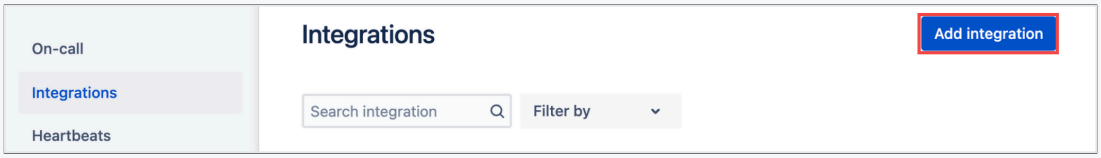
1. 알림을 받을 Webhook을 등록하세요. 등록한 URL에 POST Method로 알림 데이터가 JSON 형태로 전송됩니다. JSON 내에 포함 시킬 수 있는 대체 변수에 대해선 **사용 가능 변수** 부분을 참고하세요.
   - `${something}` 은 와탭에서 제공하는 대체될 값입니다. 지원하는 전체 목록과 의미는 사용 가능 변수를 참고하세요.
   - 이벤트 상태(`status`)는 이벤트 발생(`on`), 이벤트 해소(`off`)로 구분할 수 있습니다.
   - `level` 은 `None` , `Info` , `Warning` , `Critical` 로 구분합니다.

```
{
  "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}"
}
```

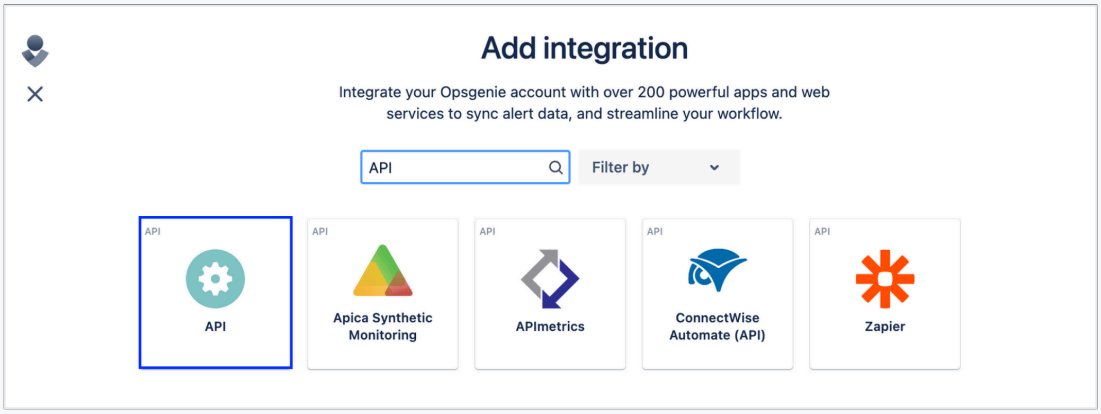
2. **Webhook Name**과 **Webhook URL** 항목을 입력하세요.
3. Header 값을 추가하려면 **사용자 Header 값 추가**의 **+ 추가**를 클릭해 key, value을 추가하세요.
4. [**등록**] 버튼을 클릭하세요.

Opsgenie

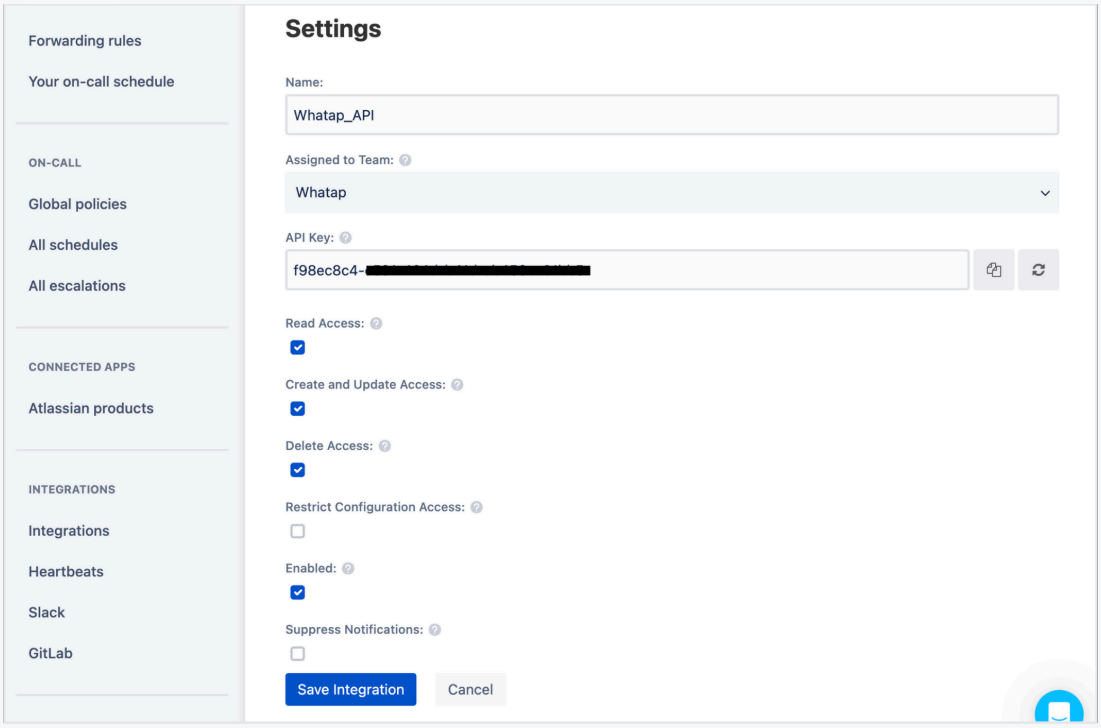
1. **Teams** 메뉴에서 알림을 수신할 팀을 선택하세요.
2. **Integration** 메뉴로 이동해 [Add Integration] 버튼을 클릭하세요.



3. Integration 목록에서 'API' 검색 후 선택하세요.



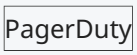
4. API 명과 Team을 할당하고 각 권한을 설정한 후, [Save Integration] 버튼을 클릭하세요.



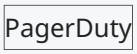
5. 와탭 모니터링 서비스의 **서드 파티 플러그인 추가** 창으로 돌아와, **Opsgenie** 탭 안내 하단의 **Name**과 **API Key** 항목을 입력하세요.
6. [**등록**] 버튼을 선택하세요.

PagerDuty

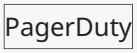
1. PagerDuty 계정에 등록하고 로그인하세요.
2. PagerDuty에서 **Services** > **Service Directory** 메뉴에서 [+ New Service] 버튼을 클릭하세요.

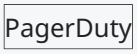
3. Service 정보(**Name**)를 입력한 후 [Next] 버튼을 클릭하세요.

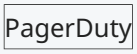
4. Escalation Policy를 설정 후 [Next] 버튼을 클릭하세요.

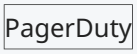
5. **Alert Grouping**과 **Transient Alerts** 항목을 설정한 후 [Next] 버튼을 클릭하세요.

6. **Integrations**에서 **Event API V2**를 선택한 후 [Create Service] 버튼을 클릭하세요.

7. 생성한 Service 화면에서 **Integration Name**과 **Integration Key** 값을 복사하세요.

8. 와탭 모니터링 서비스의 **서드 파티 플러그인 추가** 화면으로 돌아와 **Integration Name**과 **Integration Key** 항목을 입력하세요.
9. [등록] 버튼을 클릭하세요.

> 메시지를 수신하지 못했거나 플러그인이 등록되지 않았다면 **Integration Name** 또는 **Integration Key**값이 올바른지 확인하세요.

LINE

1. LINE 앱을 설치합니다.
2. 아래 QR 코드를 스캔하여 **WhaTapLabs** 공식 채널을 친구로 추가합니다.

3. 다음 명령어를 사용하여 프로젝트를 라인과 연동하고 알림을 설정하세요.

명령어

```
/help # 사용 가능한 명령어 안내
/list # 연동된 프로젝트 목록 조회
/license [액세스키] # 새 프로젝트 연동 추가
/remove [프로젝트코드] # 프로젝트 연동 해제
/language [ko|ja|en] # 알림 언어 설정 (한국어/일본어/영어)
```

- 프로젝트 액세스 키는 **관리 > 프로젝트 관리**에서 확인하실 수 있습니다.
- 라인 내에서 `/language` 명령어를 사용하는 것은 라인 내 채팅방에만 적용됩니다. 다른 3rd 플러그인이나 프로젝트의 알림 수신 언어가 변경되는 것은 아닙니다.

사용 예시

```
/license XXXXXXXXXX # 프로젝트 연동
/language ko # 한국어로 언어 설정
```

```
/list # 연동된 프로젝트 확인
```

- 프로젝트 액세스 키: WhaTap 프로젝트 내 **관리 > 프로젝트 관리**에서 확인할 수 있음
- 언어 설정: `/language` 명령어는 LINE 채팅방 내에서만 적용되며, 다른 알림 채널의 언어는 변경되지 않음

ilert

1. ilert 콘솔에서 **Alert sources** 화면의 오른쪽 위의 [+ Create alert source]를 클릭하세요.
2. **Integration type 단계**의 하단에 있는 **Filter integrations 검색창**에 `WhaTap` 을 입력해 검색합니다.
3. 검색 결과에서 **WhaTap 타일**을 선택한 후 [Next] 버튼을 클릭합니다.
4. 알림 소스 이름을 입력하고 팀을 지정한 후 [Next] 버튼을 클릭합니다.
   - **에스컬레이션 정책은** 새 정책을 생성하거나 기존 정책을 선택할 수 있습니다.
5. 알림 그룹화 방식을 선택한 후 [Continue setup] 버튼을 클릭합니다.
   - 우선 선택한 **Do not group alerts는** 이후 변경할 수 있습니다.
6. 다음 화면에서 **알림 템플릿, 우선순위 등 추가 설정**을 할 수 있습니다. 우선 [Finish setup] 버튼을 클릭해 설정을 완료합니다.
7. 설정이 완료되면 완료 화면에서 **API 키** 또는 **Webhook URL**이 생성됩니다.
8. **Webhook JSON** 탭에서 생성된 **Webhook URL**을 복사해 아래의 **Webhook URL 입력란**에 붙여넣습니다.
9. [등록] 버튼을 클릭하세요.

대량 알림 발생 방지

알림이 대량으로 발생하면 지정한 시간 동안 알림이 일시적으로 중지됩니다. 예를 들어, 5분 동안 20회의 이벤트가 발생하면 5분

동안 경고 알림을 중지합니다. 설정한 **정지 시간** 시간이 지나면 대량 알림 발생 방지 기능은 해제됩니다.

1. **경고 알림** > **이벤트 수신 설정**의 **대량 알림 발생 방지** 섹션으로 이동하세요.
2. 토글 버튼을 클릭해 **활성화** 또는 **비활성화** 할 수 있습니다.
   - **활성화**: 대량 알림 일시적 중지
   - **비활성화**: 대량 알림 재개
3. **탐지 시간, 탐지 횟수, 정지 시간**을 설정하세요.
   - **탐지 시간** 동안 **탐지 횟수** 이상의 이벤트가 발생하면 **정지 시간** 동안 경고 알림을 중지합니다.

> ⓘ 대량 알림으로 차단되는 경우 차단 안내 박스가 표시됩니다. 대량 알림 차단 기능을 해제하려면 박스의 [대량 알림 발생 중지] 버튼을 클릭하세요.

이벤트 수신 포맷

공용 알림 템플릿

공통 템플릿은 대부분의 경우 적용 가능합니다. 다양한 환경에서 동일한 포멧으로 알림을 받을 수 있습니다.

- Product Type: 애플리케이션, 데이터베이스, 쿠버네티스
- Event Type: 애플리케이션 알림, 데이터베이스 알림, 메트릭스 알림
- Event Channel: sms, mobile, 3rd party plugin, plugin

이벤트 제목

Event title format

```
[Level][Platform][ProjectName][ApplicationName][EventTitle]
```

Event title example

```
[Info][JAVA][애플리케이션 프로젝트][TC-0-1-8081][CRITICAL_HIGH_MEMORY]
```

> ⓘ ApplicationName을 설정하지 않은 경우 이벤트 제목에서 생략합니다.
>
> - Platform은 다음 중 한 가지로 표시합니다.
>   - 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

이벤트 메시지

이벤트 메시지에 포함할 수 있는 정보입니다. `optional` 이 `false` 인 경우 항상 메시지에 포함합니다. `optional` 이 `true` 인 경우 해당 데이터를 확인할 수 있으면 표시합니다.

Event message example

```
Project Name : 애플리케이션 프로젝트
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
```

표 | 이벤트 메시지 구성 요소

En	Ko	지원되는 알림 타입	설명
Project Name	프로젝트 이름	전체	-
Project Code	프로젝트 Code	전체	-
Agent Name	에이전트 이름	전체(optional)	oname
Message	이벤트 메시지	전체	-
Alert Type	이벤트 종류	전체	아래의 AlertType 종류 표 참고
Event Time	이벤트 발생 시간	전체	2022-04-13 10:40:49 +0900에서 +0900는 GMT를 의미합니다.
Event Off Time	이벤트 해제 시간	전체(optional)	2022-04-13 10:40:49 +0900에서 +0900는 GMT를 의미합니다.
Metric Name	메트릭스 이름	전체(optional)	이벤트 조건 판단에서 사용하는 메트릭스의 이름
Metric Value	메트릭스 값	전체(optional)	메트릭스 값이 메트릭스 임계치를 넘으면 이벤트 발생 조건이 만족한 경우입니다.
Metric Threshold	메트릭스 임계치	전체(optional)	메트릭스 값이 메트릭스 임계치를 넘으면 이벤트 발생 조건이 만족한 경우입니다.
Stateful	해소된 이벤트 알림	전체(optional)	해소된 이벤트 알림 기능 사용 중이면 true, 아니면 false
Event Rule	이벤트 발생 조건	메트릭스 알림	-

En	Ko	지원되는 알림 타입	설명
Target Filter	이벤트 대상 선택	메트릭스 알림	특정 대상에서 수집된 메트릭스에 대해서만 이벤트 조건을 확인합니다.
Repeat Count	이벤트 반복 횟수	메트릭스 알림	이벤트 조건이 이벤트 반복 시간동안 이벤트 반복 횟수만큼 만족해야 이벤트가 발생됩니다.
Repeat Duration	이벤트 반복 시간	메트릭스 알림	이벤트 조건이 이벤트 반복 시간동안 이벤트 반복 횟수만큼 만족해야 이벤트가 발생됩니다.
Receiver	수신자	메트릭스 알림	-
Query	MXQL 쿼리	복합 메트릭스 알림	-
Rule	이벤트 발생 조건	복합 메트릭스 알림	-
Query Period	쿼리 기간	복합 메트릭스 알림	-
Query Interval	쿼리 간격	복합 메트릭스 알림	-
Silent Time	무음 시간	복합 메트릭스 알림	-
Query	URL	Exception 알림	Exception을 발생시킨 요청의 URL
TXID	트랜잭션 ID	Exception 알림	-
Class	에러 클래스 이름	Exception 알림	-

En	Ko	지원되는 알림 타입	설명
Log Message	로그 메시지	서버 - 파일 로그 알림	-
Log File	로그 파일 경로	서버 - 파일 로그 알림	-
IP	IP	서버 알림 전체	-
CPU	CPU	서버 알림 전체	이벤트 발생 당시의 Snapshot
CPU_load1	CPU_load1	서버 알림 전체	이벤트 발생 당시의 Snapshot
CPU_loadPerCore	CPU_loadPerCore	서버 알림 전체	이벤트 발생 당시의 Snapshot
Memory	Memory	서버 알림 전체	이벤트 발생 당시의 Snapshot
Swap	Swap	서버 알림 전체	이벤트 발생 당시의 Snapshot
Disk Name | Used Percent | Free Size | IO Percent	디스크 퍼포먼스	서버 알림 전체	이벤트 발생 당시의 Snapshot
Name | Bps | Pps	트래픽 퍼포먼스	서버 알림 전체	이벤트 발생 당시의 Snapshot
Message | Time | Name	처리내역 메시지	서버 알림 전체	-

ⓘ 해당 이벤트에서 제공할 수 있는 최대한 많은 정보를 보여줍니다.

- AlertType은 다음 중 한 가지로 표시합니다.

AlertType	설명
APPLICATION_CPU	애플리케이션 CPU 알림
APPLICATION_MEMORY	애플리케이션 MEMORY 알림

AlertType	설명
APPLICATION_DISK	애플리케이션 DISK 알림
APPLICATION_ACTIVE_TRANSACTION	애플리케이션 액티브 트랜잭션 알림
APPLICATION_ERROR_TRANSACTION	애플리케이션 에러 트랜잭션 알림
APPLICATION_SLOW_TRANSACTION	애플리케이션 트랜잭션 응답시간 알림
METRICS	메트릭스 알림
COMPOSITE_METRICS	복합 메트릭스 알림
ANOMALY	이상치 탐지 알림
LOG_REALTIME	로그 실시간 알림
COMPOSITE_LOG	복합 로그 알림
SERVER_REBOOT	서버 - 재시작 알림
SERVER_NO_DATA	서버 - 미수신 알림
SERVER_PORT	서버 - 포트 알림
SERVER_NETWORK_IOPS	서버 - 네트워크 IOPS 알림
SERVER_NETWORK_BPS	서버 - 네트워크 BPS 알림
SERVER_DISK_IO	서버 - 디스크 I/O 알림
SERVER_DISK_QUOTA	서버 - 디스크 사용량 알림
SERVER_DISK_INODE	서버 - inode 알림

AlertType	설명
SERVER_CPU	서버 - CPU 알림
SERVER_MEMORY	서버 - 메모리 알림
SERVER_CPU_STEAL	서버 - steal 알림
SERVER_MEMORY_SWAP	서버 - 스왑 알림
SERVER_LOG_FILE	서버 - 로그 파일 알림
SERVER_WINDOW_EVENT	서버 - 윈도우 이벤트 알림
SERVER_OFF	서버 - 알림 OFF 알림
SERVER_ACKNOWLEDGE	서버 - 처리내역 알림
SERVER_PROCESS_COUNT	서버 - 프로세스 수 알림
SERVER_PROCESS_CPU	서버 - 프로세스 CPU 알림
SERVER_PROCESS_MEMORY	서버 - 프로세스 메모리 알림
SERVER_PROCESS_OFF	서버 - 프로세스 알림 OFF 알림
AGENT_ACTIVE	에이전트 활성화 알림
AGENT_INACTIVE	에이전트 비활성화 알림
AGENT_REACTIVATED	에이전트 재활성화 알림
URL	URL 알림
TOO_MANY_EVENT	너무 많은 이벤트 발생 알림

AlertType	설명
CLOUD_WATCH	Cloud Watch 알림
EXCEPTION	Exception 알림

이벤트 기록 및 모니터링

홈 > 프로젝트 선택 > **경고 알림** > **이벤트 기록**

이벤트 기록은 프로젝트에서 발생한 여러 이벤트를 조회하고 상세 정보를 분석할 수 있습니다. Elasticsearch DSL 쿼리를 지원하며, 원하는 조건에 맞는 이벤트를 정확하고 빠르게 찾을 수 있습니다.

이벤트 기록 조회하기

검색 결과에서 **이벤트 제목을 클릭**하면 해당 이벤트가 발생한 시간대의 상세 데이터를 분석할 수 있는 화면으로 이동합니다. 이를 통해 이벤트 발생 전후 상황을 종합적으로 파악할 수 있습니다.

1. **경고 알림** > **이벤트 기록**으로 이동하세요.
2. **이벤트 기록** 화면의 **시간 선택**에서 이벤트 기록을 조회할 기간를 설정하세요.
3. 필터 입력 창에 쿼리을 입력하고 [돋보기] 버튼을 클릭하세요.
    - 쿼리를 입력하지 않으면 전체 이벤트가 조회됩니다.

컬럼 설정

원하는 컬럼을 표시하거나 숨길 수 있습니다.

1. **이벤트 기록** 화면의 왼쪽 위 [컬럼 설정] 버튼을 클릭하세요.
2. **컬럼 설정** 창에서 표시하거나 숨길 컬럼을 선택하세요.
    - **표시 항목**에서 컬럼의 순서를 변경할 수 있습니다.
    - **컬럼명 검색**창에 원하는 컬럼 항목을 검색할 수 있습니다.

컬럼 항목	설명
이벤트 이름	이벤트의 제목
메시지	이벤트의 메시지 또는 스냅샷 정보

컬럼 항목	설명
이벤트 상태	이벤트의 현재 상태
에이전트 명	이벤트가 발생한 대상 에이전트의 이름(Oname) - 특정 에이전트에 대해 발생한 이벤트가 아닌 경우 빈칸으로 표시됨
에이전트 그룹	이벤트가 발생한 대상 에이전트의 종류(OkindName) - 특정 에이전트에 대해 발생한 이벤트가 아닌 경우 빈칸으로 표시됨
에이전트 노드	이벤트가 발생한 대상 에이전트의 서버(OnodeName) - 특정 에이전트에 대해 발생한 이벤트가 아닌 경우 빈칸으로 표시됨
이벤트 발생 시각	이벤트가 발생한 시간
이벤트 해소 시각	이벤트가 해소된 시간 상태가 없는 이벤트의 경우 `-`로 표시됨

3. 설정을 완료한 후, [확인] 버튼을 클릭해 컬럼 설정을 반영합니다.

CSV 다운로드

조회 결과를 CSV 파일 형식으로 다운로드할 수 있습니다.

1. **경고 알림** > **이벤트 기록**에서 [CSV] 버튼을 클릭하세요.
2. 현재 필터 조건에 맞는 이벤트 목록이 CSV 파일로 저장됩니다.

진행 중인 이벤트만 보기

현재 진행 중인 상태의 이벤트만 조회할 수 있습니다.

1. **경고 알림** > **이벤트 기록**에서 [진행 중인 이벤트만 보기] 버튼을 클릭하세요.
2. 필터 쿼리에 다음 조건이 자동으로 추가되어 검색됩니다.

```
Stateful: true and Status : "ON"
```

이벤트 검색

- **정확한 검색**: 원하는 조건을 정확히 지정하여 필요한 이벤트만 조회
- **복합 조건**: 여러 필드를 동시에 검색하여 복잡한 조건도 한 번에 처리
- **유연한 패턴**: 와일드카드, 부분 검색 등 다양한 검색 방식 지원
- **빠른 성능**: 인덱싱된 데이터를 활용한 고속 검색

검색 가능한 필드

필드와 값에 대해 대소문자를 구분하지 않습니다.

- **필드명 태그**: 필드가 지원되는 이벤트 타입
- **필드 타입**: 필드의 데이터의 타입
  - `#`: Number
  - `T`: String
  - : Boolean
  - : Date

이벤트 정보 필드

필드명	타입	설명	예시 값
Title	String	이벤트 제목	"Database Connection Error"
OffTitle	String	이벤트 해소 제목	"RECOVERED: Database Connection Error"
Message	String	이벤트 메시지	"Connection timeout occurred"
OffMessage	String	이벤트 해소 메시지	"RECOVERED: Connection timeout occurred"
Level	String	현재 이벤트 레벨	`Critical`, `Warning`, `Info`

필드명	타입	설명	예시 값
OriginLevel	String	원본 이벤트 레벨	`Critical`, `Warning`, `Info`
Status	String	이벤트 상태	`ON`, `OFF`, `CANCEL`, `ACKNOWLEDGE`, `MAINTENANCE`, `DISABLED`
ActCount	Number	처리내역 개수	2
MetricName	String	지표 이름	"memory"
MetricValue	String	지표 값	"85.5"
OffValue	String	해소 값	"72.8"
MetricThreshold	String	임계치	"80"
alertType	String	이벤트 종류	"BASIC", "METRICS", "TRANSACTION" 등
alertId	String	이벤트 규칙의 고유식별자	"zf3uojer0fv4v7"

이벤트 타입별 지원 필드

> ⓘ 특정 이벤트 타입에서만 지원되는 필드입니다. 해당 이벤트 타입이 아닌 경우 값이 없을 수 있습니다.

필드명	타입	이벤트 종류	설명	예시 값
eventRule	String	기본, 메트릭스	이벤트 발생 규칙	"memory ≥ 80"
field	String	실시간 로그	로그 검색 키	"content"

필드명	타입	이벤트 종류	설명	예시 값
keyword	String	실시간 로그	로그 검색 값	"Error"
logCategory	String	실시간 로그	로그 카테고리	"AppLog"
logContent	String	실시간 로그	로그 내용	"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."

대상 필드

필드명	타입	설명	예시 값
Oid	Number	에이전트 고유식별자	-98765432
Oname	String	에이전트 이름	"web-server-01"
Okind	Number	에이전트 종류 고유식별자	867318026
OkindName	String	에이전트 종류명	"web-server"
Onode	Number	에이전트 서버 고유식별자	334634079
OnodeName	String	에이전트 서버명	"production-node-1"

이벤트 고유값 필드

필드명	타입	설명	예시 값
Id	Number	이벤트의 고유식별자	5768121

필드명	타입	설명	예시 값
UUID	String	이벤트 고유식별자	"550e8400-e29b-41d4-a716"

상태/플래그 필드

필드명	타입	설명	예시 값
Stateful	Boolean	상태 기반 이벤트 여부	true, false
Disabled	Boolean	종료된 이벤트 여부	true, false
Escalation	Boolean	에스컬레이션 여부	true, false
SystemEvent	Boolean	시스템 이벤트 여부	true, false

검색 유형별 문법

> ⓘ 검색 기본 문법은 WhaTap log search query 문법 문서를 참고하시기 바랍니다.

1. 키워드 검색

Database Connection 문자열이 포함된 이벤트

```
title: "Database Connection"
```

2. 여러 값 검색

여러 옵션 중 하나라도 포함되는 이벤트를 조회합니다.

```
title: Database Connection
```

현재 이벤트 레벨이 Info이거나 Warning인 이벤트

```
level: info warning
```

3. 패턴 검색

와일드카드를 사용하여 패턴을 검색할 수 있습니다.

이름 패턴으로 특정 에이전트에서 발생된 이벤트 찾기

```
oname: web-*-prod
```

제목이 Connection으로 끝나는 이벤트 찾기

```
title: *Connection
```

4. 복합 조건 검색

여러 조건을 조합하여 정확히 검색할 수 있습니다.

Critical 레벨이면서 제목에 Database가 포함된 이벤트

```
level: critical and title: Database
```

특정 에이전트의 Warning 또는 Critical 이벤트

```
oname: web-server-01 and level: Warning Critical
```

5. OR 조건 검색

여러 조건 중 하나라도 만족하는 이벤트를 조회합니다.

제목 또는 메시지에 특정 키워드가 포함된 이벤트

```
title: Connection or message: Connection
```

6. 제외 조건

특정 조건을 제외해 검색할 수 있습니다.

Info 레벨을 제외한 모든 이벤트

```
not level: info
```

실시간 알림

실시간 알림 기능은 **이벤트 설정**을 통해 설정한 이벤트가 발생할 때 팝업 형태 표시됩니다. 이벤트 발생 즉시 알림을 받아 문제를 빠르게 인지하고 대응할 수 있습니다. 이미 발생했거나 놓친 알림 내역을 확인하여 실시간 대응이 어려웠던 상황을 파악하고 해결할 수 있습니다.

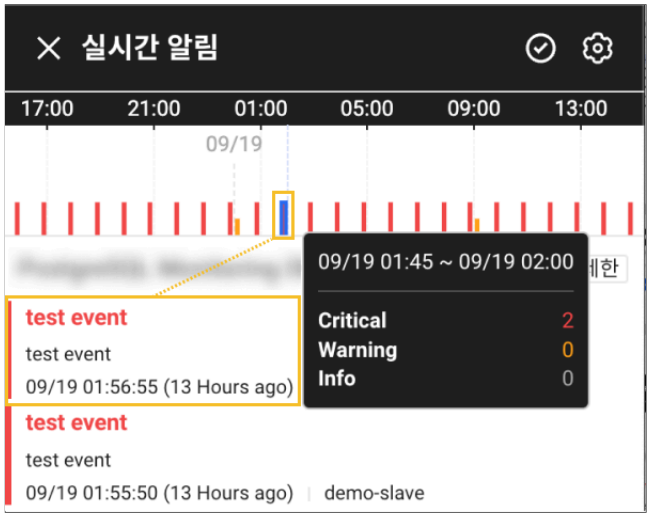
실시간 알림 확인하기

실시간 알림은 화면 오른쪽 위 팝업 형태로 표시됩니다. 알림이 발생한 시간, 제목, 메시지를 확인할 수 있습니다.

- 개별 알림을 닫으려면 메시지 오른쪽 상단의 ✕ 버튼을 클릭하세요.
- 화면에 표시된 모든 팝업 알림을 닫으려면 **전체 알림 닫기**을 클릭하세요.

알림 내역 확인하기

1. 프로젝트를 선택해 진입한 화면의 가장 위에 🔔 아이콘을 클릭하세요.
2. **실시간 알림** 창이 나타납니다.
    - 현재 프로젝트의 최근 24시간 동안의 알림을 최대 100개까지 확인할 수 있습니다.
    - 개별 이벤트 항목을 선택하면 관련 메뉴로 이동합니다.
    - **실시간 알림** 창의 시간별 차트에서 원하는 시간대를 선택하면 해당 이벤트 항목으로 이동할 수 있습니다.



- 전체 알림 내역을 확인하려면 **전체 알림 기록** 버튼을 선택하세요. **경고 알림** > **이벤트 기록** 메뉴로 이동합니다.

> **실시간 알림** 목록이 비어 있을 경우, [전체 알림 기록] 버튼이 표시됩니다. **수신 태그 미설정 알림 받기** 옵션을 해제하세요.

이벤트 일괄 해소하기

실시간 알림 목록에서 진행 중인 모든 이벤트를 일괄 해소합니다. 이벤트 해소 작업 완료까지 약 30초에서 1분 정도 소요됩니다.

- **이벤트 일괄 처리** 기능을 사용하면 현재 진행 중인 이벤트에 대한 해소 알림이 다량 발생할 수 있습니다.
- **이벤트 일괄 처리** 기능을 사용한 후에도 이벤트 규칙이 활성화되어 있고, 데이터가 임계치를 초과했다면 이벤트는 다시 발생할 수 있습니다.
- 이벤트 해소 알림은 **경고 알림** > **이벤트 설정** 메뉴에서 이벤트 수정을 통해 비활성화할 수 있습니다. **이벤트 상태가 해소되면 추가 알림** 또는 **해소된 알림** 옵션을 비활성화하세요.

사용 방법:

1. 프로젝트를 선택해 진입한 화면 상단의 아이콘을 클릭하세요.
2. **실시간 알림** 창이 나타나면 오른쪽 상단의 아이콘을 클릭하세요.

3. `실시간 알림 목록에서 진행중인 이벤트를 일괄 해소합니다.` 메시지 창이 나타나면 [해소] 버튼을 클릭하세요.

실시간 알림 설정하기

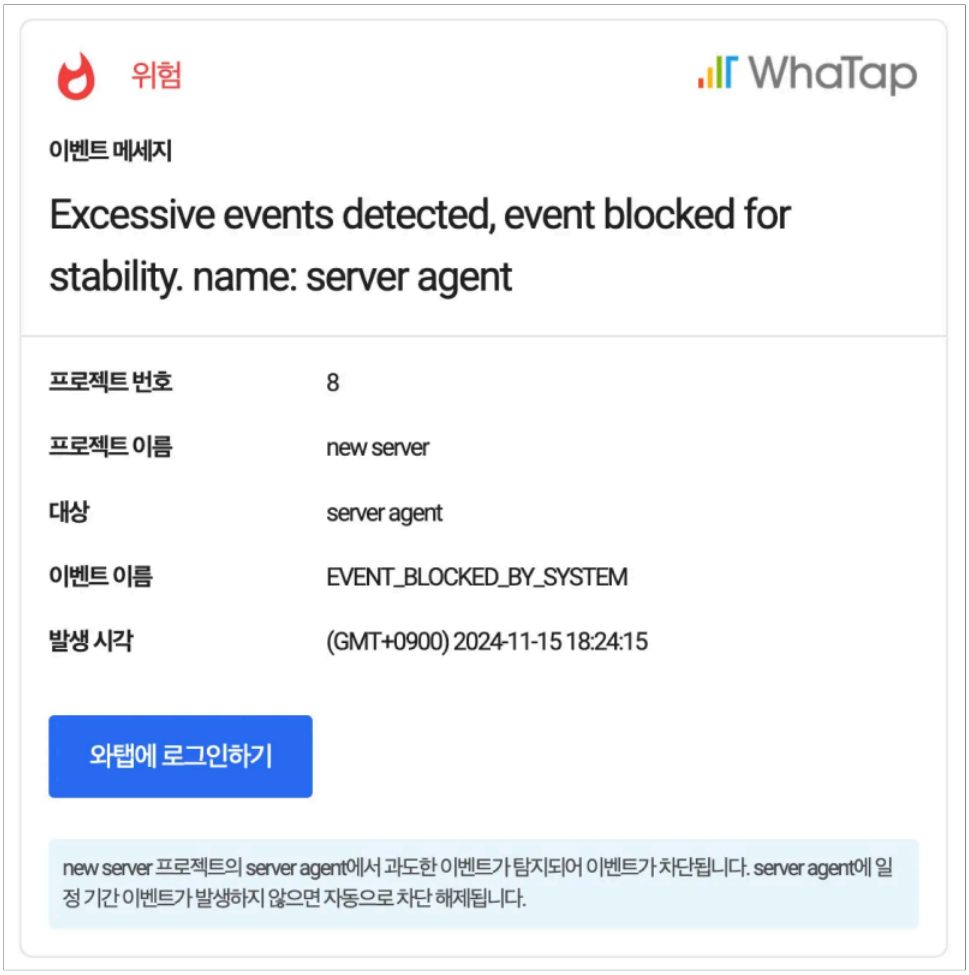
1. 프로젝트를 선택해 진입한 화면 상단의 아이콘을 클릭하세요.
2. **실시간 알림** 창이 나타나면 오른쪽 상단의 아이콘을 클릭하세요.
3. 실시간 알림 설정 창에서 각 항목을 설정하세요.
   - **팝업 알림**: **실시간 알림** 창이 닫힌 상태에서도 알림을 팝업 형태로 제공합니다.
   - **해소된 이벤트 보기**: **실시간 알림** 목록에서 해소된 이벤트를 포함해 확인할 수 있습니다.
   - **알림음 미리 듣기**: 브라우저에서 발생하는 알림음을 미리 들을 수 있습니다. Critical 또는 Warning 옆의 버튼을 클릭하세요. 알림음이 들리지 않으면 브라우저 설정을 확인하시기 바랍니다.

시스템 이벤트

와탭 모니터링은 플랫폼 전반에서 자동으로 탐지 및 생성되는 공용 이벤트를 제공합니다. 이 문서는 시스템에 의해 발생하는 이벤트의 유형 및 발생 조건, 대응 방법에 대해 안내합니다.

EVENT_BLOCKED_BY_SYSTEM

프로젝트에서 짧은 시간에 과도한 이벤트가 발생하면, 와탭 시스템이 이를 탐지하여 이벤트를 차단하고 다음 메시지를 사용자에게 전달합니다. 이벤트가 차단되었다면 사용자의 이벤트 설정을 점검하세요.



차단 기준

이벤트를 차단하는 기준은 다음과 같습니다.

1. **에이전트 단위 차단**

   한 개의 에이전트에서 **1분** 내에 **10개**의 이벤트가 발생하면 해당 에이전트에서 발생하는 이벤트를 차단합니다.

2. **다중 에이전트 차단 시 전체 차단**

   1의 경우로 이벤트가 차단된 에이전트가 **1분** 내에 **3개** 이상이 되는 경우 프로젝트 전체에서 발생하는 이벤트를 차단합니다.

3. **프로젝트 단위 차단**

   한 개의 프로젝트에서 **1분** 내에 **100개**의 이벤트가 발생하면 프로젝트 전체에서 발생하는 이벤트를 차단합니다.

해제 방법

시스템에 의해 이벤트가 차단된 경우, 1분 동안 대상 에이전트 또는 프로젝트에서 이벤트가 발생하지 않아야 합니다.

- **경고 알림** > **이벤트 설정** 메뉴에서 설정한 이벤트를 모두 비활성화하세요. **이벤트 설정을 점검하여 과도한 이벤트가 발생하지 않도록 임계치를 조정해야 합니다**.
- 이벤트를 설정하기 전 **시뮬레이션** 기능을 활용해 이벤트 발생 횟수를 예측해볼 수 있습니다.