+++ title = "Export logs of usage insights" description = "Export logs of usage insights" keywords = ["grafana", "export", "usage-insights", "enterprise"] aliases = ["/docs/grafana/latest/enterprise/usage-insights/export-logs.md"] weight = 500 +++

Export logs of usage insights

> ***Note:*** *Available in Grafana Enterprise v7.4+.*

By exporting usage logs to Loki, you can directly query them and create dashboards of the information that matters to you most, such as dashboard errors, most active organizations, or your top-10 most-used queries.

Usage insights logs

Usage insights logs are JSON objects that represent certain user activities, such as:

- A user opens a dashboard.
- A query is sent to a data source.

Scope

A log is created every time a user opens a dashboard or when a query is sent to a data source in the dashboard view. A query that is performed via Explore does not generate a log.

Format

Logs of usage insights contain the following fields, where the fields followed by * are always available, and the others depend on the logged event:

Field name	Type	Description
`eventName`*	string	Type of the event, which can be either `data-request` or `dashboard-view`.
`folderName`*	string	Name of the dashboard folder.
`dashboardName`*	string	Name of the dashboard where the event happened.
`dashboardId`*	number	ID of the dashboard where the event happened.
`datasourceName`	string	Name of the data source that was queried.
`datasourceType`	string	Type of the data source that was queried. For example, `prometheus`, `elasticsearch`, or `loki`.
`datasourceId`	number	ID of the data source that was queried.
`panelId`	number	ID of the panel of the query.
`panelName`	string	Name of the panel of the query.
`error`	string	Error returned by the query.
`duration`	number	Duration of the query.
`orgId`*	number	ID of the user’s organization.
`orgName`*	string	Name of the user’s organization.

`timestamp`*	string	The date and time that the request was made, in Coordinated Universal Time (UTC) in RFC3339 format.
`tokenId`*	number	ID of the user’s authentication token.
`username`*	string	Name of the Grafana user that made the request.
`userId`*	number	ID of the Grafana user that made the request.

Configuration

To export your logs, enable the usage insights feature and [configure]({{< relref "../../administration/configuration.md" >}}) an export location in the configuration file:

```
[usage_insights.export]
# Enable the usage insights export feature
enabled = true
# Storage type
storage = loki
```

The options for storage type are `loki` and `logger` (added in Grafana Enterprise 8.2).

If the storage type is set to `loki` you'll need to also configure Grafana to export to a Loki ingestion server. To do this, you'll need Loki installed. Refer to Install Loki for instructions on how to install Loki.

```
[usage_insights.export.storage.loki]
# Set the communication protocol to use with Loki (can be grpc or http)
type = grpc
# Set the address for writing logs to Loki (format must be host:port)
url = localhost:9095
# Defaults to true. If true, it establishes a secure connection to Loki
tls = true
```

Using `logger` will print usage insights to your [Grafana server log]({{< relref "../../administration/configuration.md#log" >}}). There is no option for configuring the `logger` storage type.

Visualize Loki usage insights in Grafana

If you export logs into Loki, you can build Grafana dashboards to understand your Grafana instance usage.

1. Add Loki as a data source. Refer to Grafana fundamentals tutorial.
2. Import one of the following dashboards:
   - Usage insights
   - Usage insights datasource details
3. Play with usage insights to understand them:
   - In Explore, you can use the query `{datasource="gdev-loki",kind="usage_insights"}` to retrieve all logs related to your `gdev-loki` data source.
   - In a dashboard, you can build a table panel with the query `topk(10, sum by (error) (count_over_time({kind="usage_insights", datasource="gdev-prometheus"} |`

`json | error != "" [$__interval])))` to display the 10 most common errors your users see using the `gdev-prometheus` data source.

- In a dashboard, you can build a graph panel with the queries `sum by(host) (count_over_time({kind="usage_insights"} | json | eventName="data-request" | error != "" [$__interval]))` and `sum by(host) (count_over_time({kind="usage_insights"} | json | eventName="data-request" | error = "" [$__interval]))` to show the evolution of the data request count over time. Using `by (host)` allows you to have more information for each Grafana server you have if you have set up Grafana for [high availability](<{{< relref "../../administration/set-up-for-high-availability.md" >}}>).