

+++ title = "Logs in Explore" description = "Logs in Explore" keywords = ["explore", "logs",] weight = 15 +++

Logs in Explore

Along with metrics, Explore allows you to investigate your logs in the following data sources:

- [Elasticsearch]({{< relref "../datasources/elasticsearch.md" >}})
- [InfluxDB]({{< relref "../datasources/influxdb/_index.md" >}})
- [Loki]({{< relref "../datasources/loki.md" >}})

During an infrastructure monitoring and incident response, you can dig deeper into the metrics and logs to find the cause. Explore also allows you to correlate metrics and logs by viewing them side-by-side. This creates a new debugging workflow:

1. Receive an alert.
2. Drill down and examine metrics.
3. Drill down again and search logs related to the metric and time interval (and in the future, distributed traces).

Logs visualization

Results of log queries are shown as histograms in the graph and individual logs are explained in the following sections.

If the data source supports a full range log volume histogram, the graph with log distribution for all entered log queries is shown automatically. This feature is currently supported by Elasticsearch and Loki data sources.

If the data source does not support loading full range log volume histogram, the logs model computes a time series based on the log row counts bucketed by an automatically calculated time interval, and the first log row's timestamp then anchors the start of the histogram from the result. The end of the time series is anchored to the time picker's **To** range.

Log level

For logs where a level label is specified, we use the value of the label to determine the log level and update color accordingly. If the log doesn't have a level label specified, we try to find out if its content matches any of the supported expressions (see below for more information). The log level is always determined by the first match. In case Grafana is not able to determine a log level, it will be visualized with an unknown log level.

Tip: If you use Loki data source and the "level" is in your log-line, use parsers (JSON, logfmt, regex,..) to extract the level information into a level label that is used to determine log level. This will allow the histogram to show the various log levels in separate bars.

Supported log levels and mapping of log level abbreviation and expressions:

Supported expressions	Log level	Color
emerg	critical	purple
fatal	critical	purple
alert	critical	purple
crit	critical	purple
critical	critical	purple

err	error	red
eror	error	red
error	error	red
warn	warning	yellow
warning	warning	yellow
info	info	green
information	info	green
notice	info	green
debug	debug	blue
debug	debug	blue
trace	trace	light blue
*	unknown	grey

Logs navigation

Logs navigation next to the log lines can be used to request more logs. You can do this by clicking on Older logs button on the bottom of navigation. This is especially useful when you hit the line limit and you want to see more logs. Each request that is run from the navigation is then displayed in the navigation as separate page. Every page is showing from and to timestamp of the incoming log lines. You can see previous results by clicking on the page. Explore is caching last five requests run from the logs navigation, so you are not re-running the same queries when clicking on the pages.



Navigate logs in Explore

Visualization options

You can customize how logs are displayed and select which columns are shown.

Time

Shows or hides the time column. This is the timestamp associated with the log line as reported from the data source.

Unique labels

Shows or hides the unique labels column that includes only non-common labels. All common labels are displayed above.

Wrap lines

Set this to True if you want the display to use line wrapping. If set to False, it will result in horizontal scrolling.

Prettify JSON

Set this to `true` to pretty print all JSON logs. This setting does not affect logs in any format other than JSON.

Deduping

Log data can be very repetitive and Explore can help by hiding duplicate log lines. There are a few different deduplication algorithms that you can use:

- **Exact** - Exact matches are done on the whole line except for date fields.
- **Numbers** - Matches on the line after stripping out numbers such as durations, IP addresses, and so on.
- **Signature** - The most aggressive deduping, this strips all letters and numbers and matches on the remaining whitespace and punctuation.

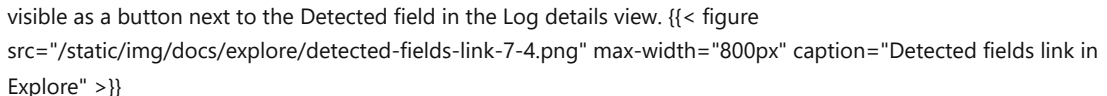
Flip results order

You can change the order of received logs from the default descending order (newest first) to ascending order (oldest first).

Labels and detected fields

Each log row has an extendable area with its labels and detected fields, for more robust interaction. For all labels we have added the ability to filter for (positive filter) and filter out (negative filter) selected labels. Each field or label also has a stats icon to display ad-hoc statistics in relation to all displayed logs.

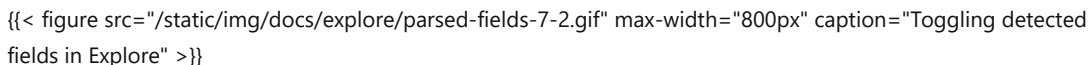
Derived fields links

By using Derived fields, you can turn any part of a log message into an internal or external link. The created link is visible as a button next to the Detected field in the Log details view. 

Toggle detected fields

Note: Available in Grafana 7.2 and later versions.

If your logs are structured in `json` or `logfmt`, then you can show or hide detected fields. Expand a log line and then click the eye icon to show or hide fields.



Loki-specific features

As mentioned, one of the log integrations is for the new open source log aggregation system from Grafana Labs - [Loki](#). Loki is designed to be very cost effective, as it does not index the contents of the logs, but rather a set of labels for each log stream. The logs from Loki are queried in a similar way to querying with label selectors in Prometheus. It uses labels to group log streams which can be made to match up with your Prometheus labels. For more information about Grafana Loki, refer to [Grafana Loki](#) or the Grafana Labs hosted variant: [Grafana Cloud Logs](#).

For more information, refer to [Loki's data source documentation]([../datasources/loki.md](#)) on how to query for log data.

Switch from metrics to logs

If you switch from a Prometheus query to a logs query (you can do a split first to have your metrics and logs side by side) then it will keep the labels from your query that exist in the logs and use those to query the log streams. For example, the following Prometheus query:

```
grafana_alerting_active_alerts{job="grafana"}
```

after switching to the Logs data source, the query changes to:

```
{job="grafana"}
```

This will return a chunk of logs in the selected time range that can be grepped/text searched.

Live tailing

Use the Live tailing feature to see real-time logs on supported data sources.

Click the **Live** button in the Explore toolbar to switch to Live tail view.

While in Live tail view new logs will come from the bottom of the screen and will have fading contrasting background so you can keep track of what is new. Click the **Pause** button or scroll the logs view to pause the Live tailing and explore previous logs without interruption. Click **Resume** button to resume the Live tailing or click **Stop** button to exit Live tailing and go back to standard Explore view.

{{< figure src="/static/img/docs/v64/explore_live_tailing.gif" class="docs-image--no-shadow" caption="Explore Live tailing in action" >}}